

Unidade 1

Introdução a Metodologias Ágeis



Abertura

Olá, amigo(a) discente! Seja bem-vindo(a)!

Diante da existência de uma grande demanda por desenvolvimento de software, diversas empresas vêm adotando os métodos ágeis como uma alternativa aos métodos tradicionais de desenvolvimento de software. Em face de um histórico processo de extrapolação de custos e prazos, onde o desenvolvimento se desdobra lentamente entre análise de requisitos, implementação, testes, integração e manutenção, o *Agile* surgiu como uma resposta à produtividade.

O modelo ágil tem como premissa ciclos iterativos e incrementais, que possibilita flexibilidade e adaptação, esses ciclos são voltados para a melhoria contínua de equipes e processos, tendo como foco entrega de valor para os clientes em um curto espaço de tempo. Nesta unidade vamos compreender e discutir os principais conceitos que corroboram para o sucesso das metodologias ágeis.

Na aula 01, vamos fazer um comparativo entre as metodologias tradicionais e as metodologias ágeis, além de entender como elas estão organizadas e suas premissas.

Na aula 02, dando continuidade ao nosso processo de conhecimento das metodologias ágeis, vamos conhecer o tão falado manifesto ágil, e como ele impacta o desenvolvimento de softwares.

Objetivos

- Ter uma visão geral das diferenças entre as metodologias tradicionais e metodologias ágeis;
- Entender como as metodologias ágeis funcionam e por que elas fazem tanto sucesso atualmente nas fábricas de software.

Conteúdo Programático

Aula 01 – Conhecendo as metodologias ágeis e suas diferenças em relação às metodologias tradicionais

Aula 02 – Manifesto ágil



Quer **assistir às videoaulas** em seu celular? Basta apontar a câmera para os **QR Codes** distribuídos neste conteúdo.

Caso necessário, instale um aplicativo de leitura QR Code no celular e efetue o login na sua conta Gmail.



Videoaula Apresentação da Disciplina

Utilize o QR Code para assistir!

Assista!



Videoaula Minicurrículo

Utilize o QR Code para assistir!

Assista!



Introdução Metodologia Tradicional e Metodologias Ágeis

Olá, estudante, como vai? Bom vamos começar nossos estudos fazendo um comparativo entre as metodologias tradicionais e as metodologias ágeis. Você deve estar se perguntando para que eu tenho que estudar isso? Às vezes visualizamos o desenvolvimento de software somente pela etapa de codificação, esquecendo de todas as outras etapas da cadeia do desenvolvimento de software.

Hoje em dia para qualquer aplicação que queremos desenvolver precisamos de um projeto. Em meio a esse processo, devemos ficar atentos a algumas especificações, como os prazos, custos, qualidade e satisfação do cliente, visto que o não cumprimento destes requisitos podem gerar problemas mais à frente. Deste modo, dar a importância devida ao gerenciamento de projetos é fundamental para o sucesso do negócio, nesse sentido, o uso de uma boa metodologia é essencial. Agora você deve estar se perguntando, qual metodologia devo usar? Atualmente temos duas metodologias que são bem conhecidas no mercado de software: as metodologias tradicional e ágil.



Videoaula 1

Utilize o QR Code para assistir!

Assista o vídeo para entender melhor o assunto.



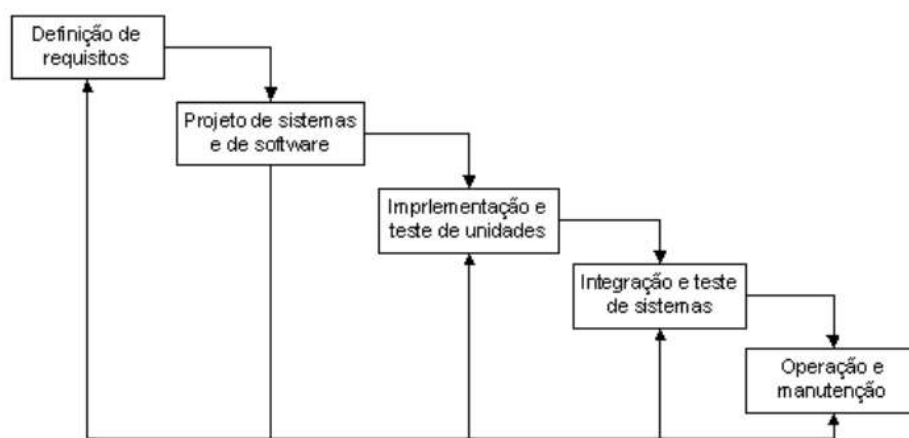
Metodologia Tradicional

A metodologia tradicional tem fases bem definidas sendo elas: planejamento do projeto, estimativas de prazo e custo, desenvolvimento (execução) e entrega do produto quando tudo estiver pronto. Essa seria a estrutura de um software de vendas, utilizando as etapas da metodologia tradicional:

- Planejamento do software (pensar em como o software ficará depois de pronto);
- Planejamento das atividades (análise, programação, design, etc.);
- Definição de prazos e custos;
- Execução (mão na massa);
- Testes;
- Implantação.

A metodologia tradicional segue um modelo sequencial, ou seja, uma etapa deve ser executada após a outra, deste modo, uma tarefa não pode ser iniciada enquanto a anterior não for finalizada. Outro ponto que se espera nessa metodologia, é que seja executado exatamente o que foi planejado, focando sempre no resultado final. Para o projeto ser considerado um sucesso, não basta apenas seguir as ordens definidas previamente, é importante entregar no prazo, dentro do orçamento e com a qualidade esperada. No contexto da metodologia tradicional um produto só gera valor quando o projeto estiver 100% finalizado. A metodologia tradicional mais conhecida é o modelo Cascata (*waterfall*). Na Figura 1. Pode-se visualizar o fluxo do modelo Cascata.

Figura 1. Modelo Cascata



Fonte: SOMMERVILLE, 2004.

A principal dificuldade dentro da metodologia tradicional é que ela não tem muita flexibilidade em relação às mudanças. Diante de um mundo tão dinâmico, onde as coisas mudam rapidamente, esse é um problema que gera grande preocupação para os gerentes de projetos.

Metodologia Ágil

Apesar da denotação, a palavra ágil não significa agilidade, mas sim o poder de quebrar o projeto em partes menores. Ao contrário da metodologia tradicional que prega uma entrega só no final do projeto, na metodologia ágil as entregas são constantes, e acontecem até que o projeto todo seja finalizado.

As preocupações com custos, qualidade e prazos são as mesmas da metodologia tradicional, porém você consegue controlar e gerenciar as mudanças com maior facilidade, visto que, conforme as entregas acontecem, elas já podem ser testadas pelo usuário. Deste modo, bug ou outros tipos de problemas podem ser facilmente corrigidos. Na metodologia ágil o foco principal é a entrega de valor ao cliente, por isso, é priorizado a entrega de funcionalidade ao invés de documentações robustas, por exemplo. Deve ficar claro ao leitor que isso não quer dizer que não haverá documentação, mas que o foco maior ficará na entrega de valor. Na metodologia ágil existem vários contextos da metodologia tradicional, mas vistos de maneiras

diferentes, como por exemplo: o planejamento da metodologia ágil é de forma iterativa e incremental, enquanto, na metodologia tradicional planeja-se com muita antecedência como será cada fase do projeto. Na Figura 2. Pode-se visualizar uma reunião de planejamento na metodologia ágil.

Figura 2. Exemplo de reunião de planejamento na Metodologia Ágil



Fonte: Annelise Gripp

Metodologia Tradicional x Metodologias Ágeis

As metodologias tradicionais se amparam em etapas mais rígidas e controladas, enquanto as metodologias ágeis se baseiam na flexibilidade e adaptabilidade das estratégias. Antes de continuarmos com as nossas comparações, é importante deixar claro que não existe certo ou errado, cada organização adota a metodologia que faz mais sentido para a sua realidade.

Uma das maiores diferenças entre as duas metodologias estão nas questões de rigidez e flexibilidade. As metodologias tradicionais como as citadas acima são mais rígidas, ou seja, existem fases predefinidas que devem ser cumpridas de maneira sequencial até que o projeto seja concluído. Já nas metodologias ágeis o funcionamento é diferente, em vez de seguir um modelo predefinido, com uma única entrega no final do projeto, há várias entregas o que facilitam a gestão de mudanças que pode ocorrer durante a execução do projeto. As entregas constantes aumentam as chances de sucesso do produto, visto que elas servem como um termômetro do que o cliente realmente espera, fazendo com que as próximas etapas do projeto sejam focadas na necessidade do usuário.

Em relação à hierarquia, as metodologias tradicionais focam em uma figura central. Normalmente, esse papel fica com o líder do projeto a qual responde por todo o processo e

delega as atividades. Já nas metodologias ágeis a estrutura é dividida em equipes multidisciplinares que possuem autonomia para a tomada de certas decisões.

Com relação ao controle orçamentário também há uma distinção entre as metodologias, sendo que na tradicional o orçamento é definido a longo prazo e na fase de planejamento, percorrendo todas as fases do projeto sem que haja uma grande mudança durante o seu desenvolvimento. Na metodologia ágil esse conceito é um pouco diferente, visto que é possível administrar o custo do projeto a partir do desenvolvimento de cada fase, deste modo é possível identificar oportunidades para reduzir custos ou até mesmo realizar aportes que possam trazer resultados mais satisfatórios para o negócio.



Videoaula 2

Utilize o QR Code para assistir!

Assista o vídeo para entender melhor o assunto.



Indicação de Vídeo:

Projetos Ágeis e Tradicionais: qual é a diferença?

https://www.youtube.com/watch?v=5oFzY1_pXI. Acesso em: 11 mar. 2022.

Conhecendo as Metodologias Ágeis mais Usadas no Mercado

Hoje há inúmeras metodologias ágeis disponíveis no mercado, sendo que cada uma tem suas particularidades. Nessa seção descreveremos de forma introdutória algumas dessas metodologias.

Scrum

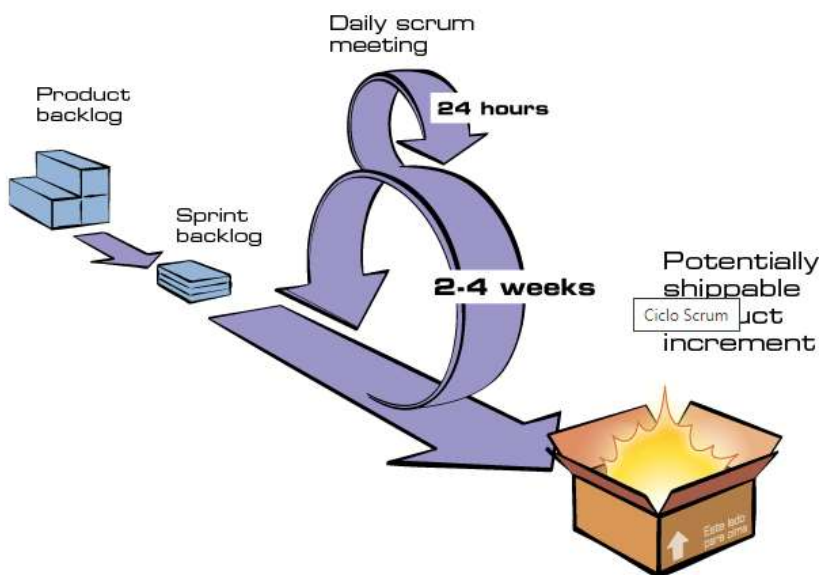
Atualmente, o *Scrum* é a metodologia ágil mais usada no mercado. No *Scrum*, os projetos são divididos em ciclos que duram normalmente de duas a quatro semanas, esses ciclos são chamados de *Sprints*. O *Sprint* representa um *Time Box* dentro do qual um conjunto de atividades deve ser executado, ou seja, os sprints podem ser definidos como cada uma das fases de um projeto, estipuladas em espaços definidos de tempo. As funcionalidades a serem implementadas são colocadas em uma lista de pendências conhecida como *Product Backlog*.

No início de cada *Sprint* é realizado uma *Sprint Planning*, ou seja, uma reunião de planejamento na qual o *Product Owner (PO)* prioriza os itens do *Product Backlog* e o time de desenvolvimento seleciona as atividades que eles jugam serem capazes de implementar durante aquela *Sprint* que se inicia.

O time de desenvolvimento realiza diariamente reuniões rápidas de atualização que visam dar transparência no andamento do projeto para todo o time, essas reuniões são denominadas *Daily Scrum*. É também nestas reuniões que cada membro do time expõe o que foi concluído no dia anterior, aquilo que está sendo feito no momento e se há algo que vem impedindo o seu progresso.

Ao final do *Sprint* ocorre o *Sprint Review*, neste evento o time apresenta as funcionalidades que foram implementadas, cada membro do time analisa os recursos recém desenvolvidos. É nesse momento que ocorre o *feedback*, além de ser uma oportunidade de integração com outras equipes envolvidas no projeto. Finalmente, faz-se uma *Sprint Retrospective* e o time parte para o planejamento do próximo *Sprint*. Deste modo, reinicia-se um próximo ciclo. Na Figura 3 é possível visualizar uma ilustração do ciclo do *Scrum*.

Figura 3. Ciclo do Scrum



Fonte: disponível em: <https://bit.ly/35NuG3v>. Acesso em: 11 mar. 2022.

Lean

O Lean é mais que uma metodologia, é uma filosofia de gestão que foi introduzida nos anos 50 pelo setor automobilístico e tem como principal objetivo um desenvolvimento enxuto e eficiente. Essa metodologia é usada para lidar com os problemas de forma

sistêmica e otimizar vários pontos em um negócio, tudo o que não é estritamente necessário para realizar determinado trabalho é visto como excesso, portanto, pode ser eliminado. Esse processo é feito de maneira fluída dentro da organização, identificando gargalos e cortando todos os desperdícios para aumentar os resultados e entregar um maior valor aos clientes.

No método Lean, o time conta com um fluxo de trabalho claro e compreensível, com entregas rápidas e identificação precisa de problemas. As principais características desta metodologia são:

- Eliminar o desperdício;
- Aprimoramento contínuo;
- Otimização dos processos;
- Entregar o mais rápido possível;
- Empoderar o time;
- Qualidade e maior valor ao cliente.

Kanban

O método Kanban nasceu em uma fábrica automobilística no Japão e se popularizou por times ágeis de desenvolvimento de software graças a sua flexibilidade, velocidade e eficiência. O Kanban é um método visual para gerenciar e conduzir o trabalho, ele organiza as tarefas por status, em quadros, com as etapas do projeto distribuídas em colunas (a fazer, fazendo e pronto), deste modo ele cria um controle visual para os times.

As premissas do método Kanban são aplicáveis a praticamente qualquer tipo de projeto, seja ele software ou não. O Kanban é utilizado em associação ao conceito de *Just in time* [\[1\]](#), que garante o uso apenas dos recursos necessários para as tarefas que estão em desenvolvimento. Sendo assim, é possível eliminar excedentes ou sobras e diminuir custos.

[\[1\] Just in Time](#), significa “momento certo”, é um sistema com objetivo de produzir uma quantidade exata de um produto, de acordo com a demanda, de forma rápida e sem a necessidade da formação de estoques.

Extreme Programming (XP)

O Extreme Programming mais conhecido como XP é uma metodologia ágil voltada ao desenvolvimento de softwares que visa obter um produto com a maior qualidade possível. O XP é uma metodologia leve que pode ser facilmente adotada por diferentes níveis de desenvolvedores e para qualquer tamanho de time.

O XP pode ser utilizado de maneira complementar ao Scrum, uma vez que ele acaba focando mais em processo de engenharia e desenvolvimento do software. O XP possui um conjunto de princípios e valores. Os valores no XP servem como critério que norteiam as pessoas envolvidas no desenvolvimento do software, são eles:

- **Comunicação:** deve ser clara, frequente e adequada para transferência de conhecimento, a comunicação deve ser definida como parte do processo;
- **Simplicidade:** garantir eficiência e foco apenas no que é necessário para o sucesso do projeto;
- **Feedback:** deve ser contínuo, rápido e assertivo;
- **Coragem:** para tomar atitudes corretas, comunicar problemas, fazer coisas pouco efetivas e tentar novas abordagens. Coragem não é inconsequência – seja equilibrado;
- **Respeito:** respeite o seu time, respeite o que os outros fazem, respeite o projeto e cuide dele, tenha senso de responsabilidade;



Videoaula 3

Utilize o QR Code para assistir!

Assista o vídeo para entender melhor o assunto.



Além dos valores, existe um conjunto de princípios que o time de desenvolvimento deve ter, são eles: *feedback* rápido, presumir simplicidade, abraçar mudanças, trabalho de alta qualidade, melhoria contínua, diversidade e reflexão.

Manifesto Ágil

Olá, aluno, bora para a nossa segunda aula desta unidade? Nesta aula vamos conhecer o manifesto ágil, como ele surgiu e como ele impacta a maneira de desenvolver sistemas atualmente, além disso, vamos conhecer juntos quais são os 12 princípios que o regem.



Videoaula 1

Utilize o QR Code para assistir!

Assista o vídeo para entender melhor o assunto.



O manifesto Ágil tem sua origem em uma reunião dos líderes do *Extreme Programming* no ano de 2000, onde eles debateram a relação entre o XP e o que era denominado na época como métodos leves. Esses métodos eram o que nós conhecemos hoje como metodologias ágeis, ou seja, o Scrum, DSDM, Crystal e etc.

Tudo isso surge, pois assim como o XP esses métodos estavam indo no inverso dos métodos de desenvolvimento de softwares pesados e mais robustos cheios de burocracia e engessado como o modelo cascata.

Com esse pensamento foi que surgiu a ideia de reunir todas essas pessoas em um só lugar para debater sobre esses assuntos e como eles impactam no desenvolvimento de um software.

Esse encontro então ocorre em fevereiro de 2001 em um *resort de Ski* nas montanhas de Utah, diversas pessoas foram convidadas, porém somente 17 estavam presentes. Durante o encontro essas pessoas entraram em um consenso sobre como deveriam ser aplicados os métodos de desenvolvimento de software e esse consenso é o que se chama de manifesto ágil.

Indicação de Leitura

Assista esse link: Manifesto Ágil.

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

Assinaram este manifesto:

- Kent Beck
- Mike Beedle

- Arie van Bennekum
- Alistair Cockburn
- Ward Cunningham
- Martin Fowler
- James Grenning
- Jim Highsmith
- Andrew Hunt
- Ron Jeffries
- Jon Kern
- Brian Marick
- Robert C. Martin
- Steve Mellor
- Ken Schwaber
- Jeff Sutherland
- Dave Thomas

Entre essas pessoas se encontram os criadores do Scrum (Ken Schwaber e Jeff Sutherland) e do XP (Kent Beck e Ron Jeffries) além de nomes como Robert C. Martin autor do livro *Clean Code* (Código Limpo).

Neste manifesto se declara qual a nova visão do processo de desenvolvimento de softwares, dessa forma são valorizados dentro do manifesto:

- **Indivíduos e interações:** é entender que o desenvolvimento de software é **uma atividade humana** e que a qualidade da interação entre as pessoas pode resolver problemas crônicos de comunicação. Processos e Ferramentas são importantes, mas devem ser simples e úteis
- **Software em funcionamento:** O maior indicador de que sua equipe realmente construiu algo é software funcionando. Clientes querem o resultado e isso pode ser com software funcionando. Documentação também é importante, mas que seja somente o necessário e que agregue valor.
- **Colaboração com o cliente:** equipe de desenvolvimento e o cliente precisam sempre estar alinhados e trabalhando em conjunto e não separados, para que dessa forma possam alcançar os objetivos mais rápidos.
- **Responder a mudanças:** Desenvolver softwares e produtos é um ambiente de alta incerteza e por isso não podemos nos debruçar em planos enormes e cheio de premissas. O que deve ser feito é aprender com as informações e *feedbacks* e adaptar o plano a todo momento.

Um trecho do manifesto:

Estamos descobrindo maneiras melhores de desenvolver software, fazendo-o nós mesmos e ajudando outros a fazerem o mesmo. Por meio 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; Responder a mudanças mais que seguir um plano. Ou seja, mesmo havendo valor nos itens à direita, valorizamos mais os itens à esquerda (BECK *et al.*, 2001).



Videoaula 2

Utilize o QR Code para assistir!

Assista o vídeo para entender melhor o assunto.



Nesta visão, desenvolver um software deixa de ser apenas a ideia de construção de um produto e passa a ser a entrega de um serviço, com isto desenvolver um software é ir além do que se espera, é proporcionar às pessoas (time de desenvolvimento e cliente) uma experiência agradável e preparada para mudanças.

Figura 4. Metodologias Ágeis



Fonte: disponível em: <https://bit.ly/3hVYjll>. Acesso em: 11 mar. 2022.

Na Figura 4 podemos identificar algumas metodologias ágeis que foram incluídas do manifesto, pois seus criadores participaram da construção do mesmo e essas metodologias seguem os valores e princípios propostos.

Além desses valores o manifesto também propõe 12 princípios a serem seguidos nas metodologias ágeis, são eles:

1. Nossa maior prioridade é satisfazer o cliente, por meio da entrega adiantada e contínua de software de valor.

2. Aceitar mudanças de requisitos, mesmo no fim do desenvolvimento. Processos ágeis se adequam a mudanças, para que o cliente possa tirar vantagens competitivas.
3. Entregar software funcionando com frequência, na escala de semanas até meses, com preferência aos períodos mais curtos.
4. Pessoas relacionadas a negócios e desenvolvedores devem trabalhar em conjunto e diariamente, durante todo o curso do projeto.
5. Construir projetos ao redor de indivíduos motivados. Dando a eles o ambiente e suporte necessário, e confiar que farão seu trabalho.
6. 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.
7. Software funcional é a medida primária de progresso.
8. Processos ágeis promovem um ambiente sustentável. Os patrocinadores, desenvolvedores e usuários, devem ser capazes de manter indefinidamente, passos constantes.
9. Contínua atenção à excelência técnica e bom design, aumenta a agilidade.
10. Simplicidade a arte de maximizar a quantidade de trabalho que não precisou ser feito.
11. As melhores arquiteturas, requisitos e designs emergem de times auto organizáveis.
12. Em intervalos regulares, o time reflete em como ficar mais efetivo, então, se ajustam e otimizam seu comportamento de acordo.



Videoaula 3

Utilize o QR Code para assistir!

Assista o vídeo para entender melhor o assunto.



Com a aplicação desses valores e princípios deixa de existir divisão e os então métodos leves de desenvolvimento começam a ser chamados de métodos ágeis.

Indicação de Vídeo:

Metodologia Agile #3AG - Quais são os valores e práticas associados ao primeiro pilar do manifesto? Disponível em: <https://www.youtube.com/watch?v=mj7DDQse0u0>.

Acesso em: 11 mar. 2022.

Implementar os pilares do manifesto no dia a dia da empresa é um desafio que ainda é vivenciado dentro das instituições pois envolve muito mais uma mudança cultural das pessoas do que uma mudança estrutural da instituição. Grandes empresas como Google, IBM e Dell implementam os valores e princípios ágeis dentro do seu cotidiano e com isso alcançam novos resultados todos os dias.

Nas próximas unidades vamos falar de forma mais detalhadas sobre essas e algumas outras metodologias ágeis usadas no mercado e inclusive vamos aplicar na prática esses conceitos.

Encerramento

Chegamos ao final da nossa primeira unidade, onde você teve contato com os primeiros conhecimentos de metodologia ágil. Com certeza, muito do que foi mostrado aqui você já tinha uma pequena noção. Espero que você, aluno, tenha aprendido um pouco mais sobre todo esse mundo de metodologias ágeis

No mundo dos softwares há muitas metodologias de desenvolvimento que podem ser melhores ou piores, no final tudo depende da demanda e do produto que deve ser entregue. Na aula 1 visualizamos alguns conceitos e como funciona o relacionamento entre metodologia tradicional e metodologias ágeis além de conhecer algumas metodologias ágeis.

Na aula 2, foi discutido e apresentado o manifesto ágil, um marco dentro da realidade de desenvolvimento de sistemas, conversamos sobre os seus 2 pilares, que são seus valores e princípios e também conhecemos como esse manifesto foi construído

Juntando as duas aulas teremos um embasamento básico sobre metodologias ágeis. Caro aluno, pense um pouco como os conhecimentos desta unidade podem te auxiliar em seu dia a dia. Até a próxima.

Referências

BECK, Kent *et al.* **Manifesto Ágil**. 2001. Disponível em: <https://agilemanifesto.org/iso/ptbr/manifesto.html>. Acesso em: 11 mar. 2022.

PRESSMAN, Roger S. **Engenharia de Software**. 1. ed. São Paulo: Person, 2011.

SCRUM. **O Guia Evidence-Based Management**. 2020. Disponível em: https://scrumorg-website-prod.s3.amazonaws.com/drupal/2021-01/2020-EBM-Guide-Portuguese-BR-2.0_0.pdf. Acesso em: 11 mar. 2022.

SUTHERLAND, Jeff. **A Arte de Fazer o Dobro do Trabalho na Metade do Tempo**. São Paulo: LEYA, 2014.



Quer **assistir às videoaulas** em seu celular? Basta apontar a câmera para os **QR Codes** distribuídos neste conteúdo.

Caso necessário, instale um aplicativo de leitura QR Code no celular e efetue o login na sua conta Gmail.



UNIFIL.BR