



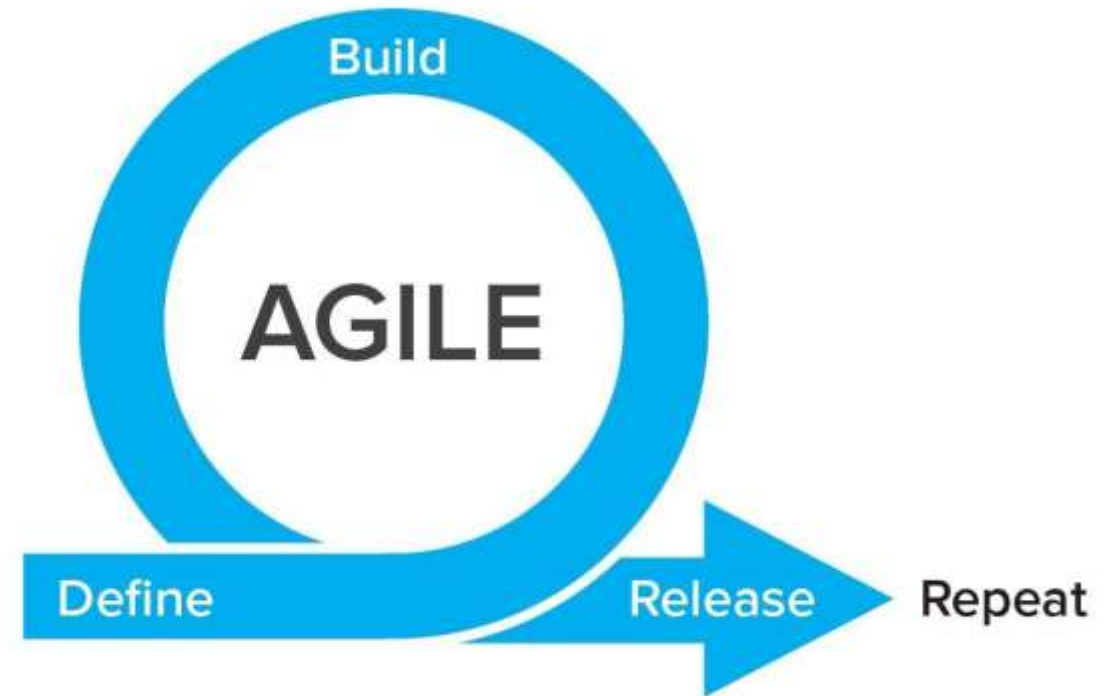
FILOSOFIA DO PROJETO GNU

- Software livre significa que os usuários do software têm liberdade. (A questão não é sobre o preço.) Nós desenvolvemos o sistema operacional GNU de modo que os usuários possam ter liberdade no uso do computador.
- Especificamente, software livre significa que os usuários têm as quatro liberdades essenciais:
 - (0) para executar o programa;
 - (1) para estudar e mudar o código-fonte do programa;
 - (2) para redistribuir cópias exatas e
 - (3) para distribuir versões modificadas.



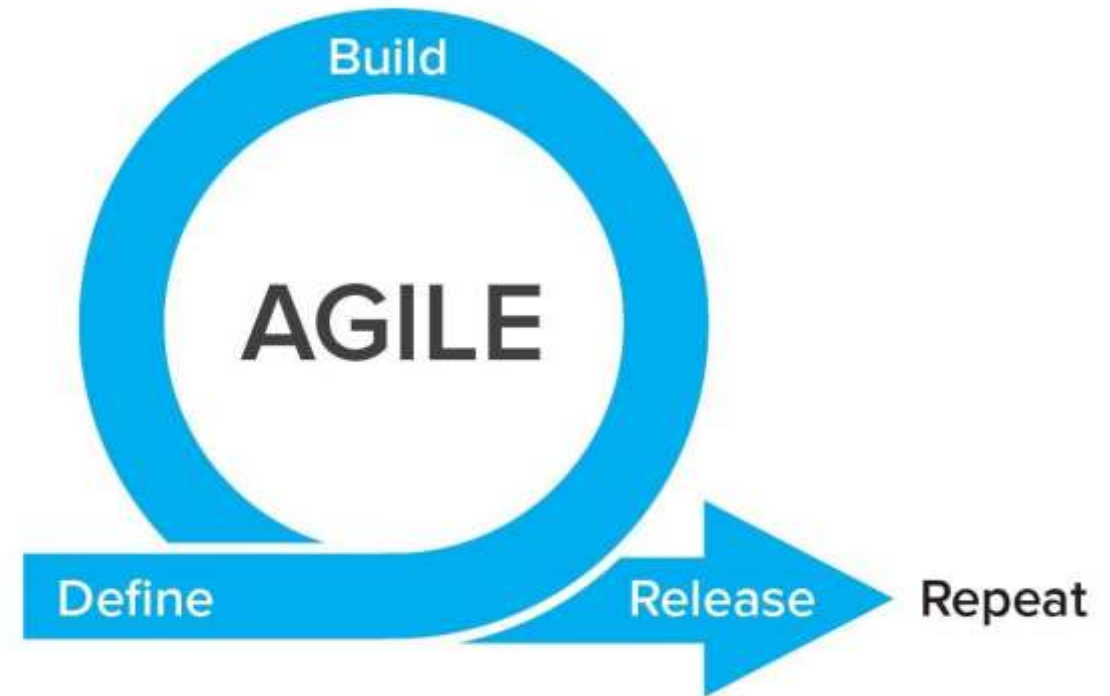
O QUE É SER ÁGIL?

- 1. Se v.quis dizer “ágil”, com acento, ela deriva do Latim AGILITAS, “rapidez, mobilidade”, de AGERE, “mover-se, deslocar-se”.
- 2. Esta vem do Anglo-Francês POVOIR, “ser capaz”, do Latim POTERE, “poder, obter, conseguir”.
- 3. Que se comporta ou trabalha de maneira eficaz e rápida; diligente, expedito e trabalhador.



AGILIDADE

- Ser ágil é sinônimo de adaptabilidade. Muitos acham que agilidade é a mesma coisa ser rápido, mas estão enganados. A agilidade é uma característica importante para se acomodar com o ambiente de incertezas em que está inserido.



MANIFESTO ÁGIL

- Em 2001, um grupo de 17 desenvolvedores reconhecidos se juntou em Utah, nos EUA, para discutir maneiras de desenvolvimento mais leves com base em suas experiências.
- Eles assinaram um documento chamado “Manifesto para o Desenvolvimento Ágil de Software”.



OS 4 PILARES DO MANIFESTO ÁGIL

- **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



OS 12 PRINCÍPIOS DO MANIFESTO ÁGIL

- 1. Nossa maior prioridade é satisfazer o cliente através 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 à 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. 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;
- 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;

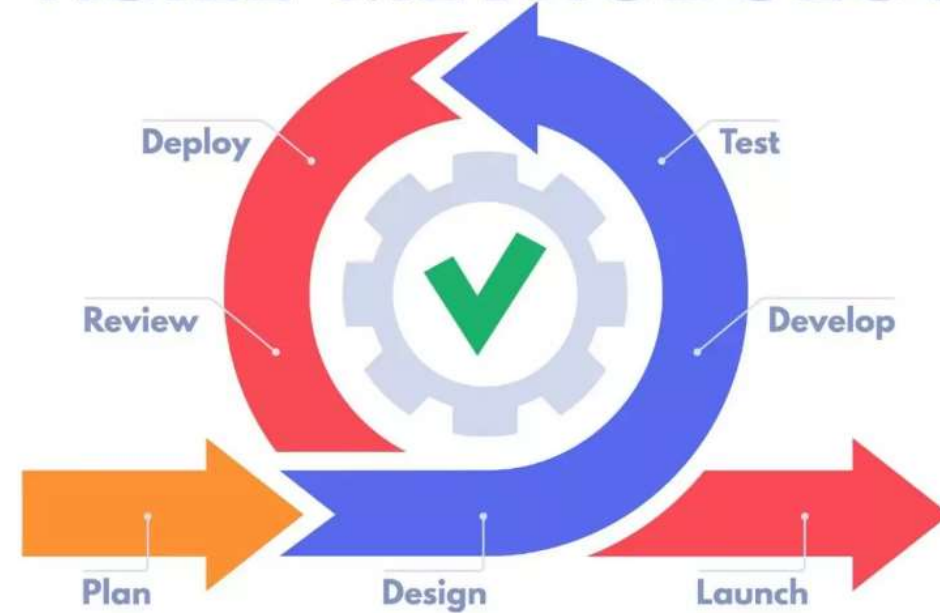
AUTORES



MÉTODOS ÁGEIS

- XP – Xtreme Programing;
- DSDM – Driven Software Development Model;
- FDD – Feature Driven Development;
- LEAN – Lean Software Development;
- SCRUM;

AGILE METHODOLOGY

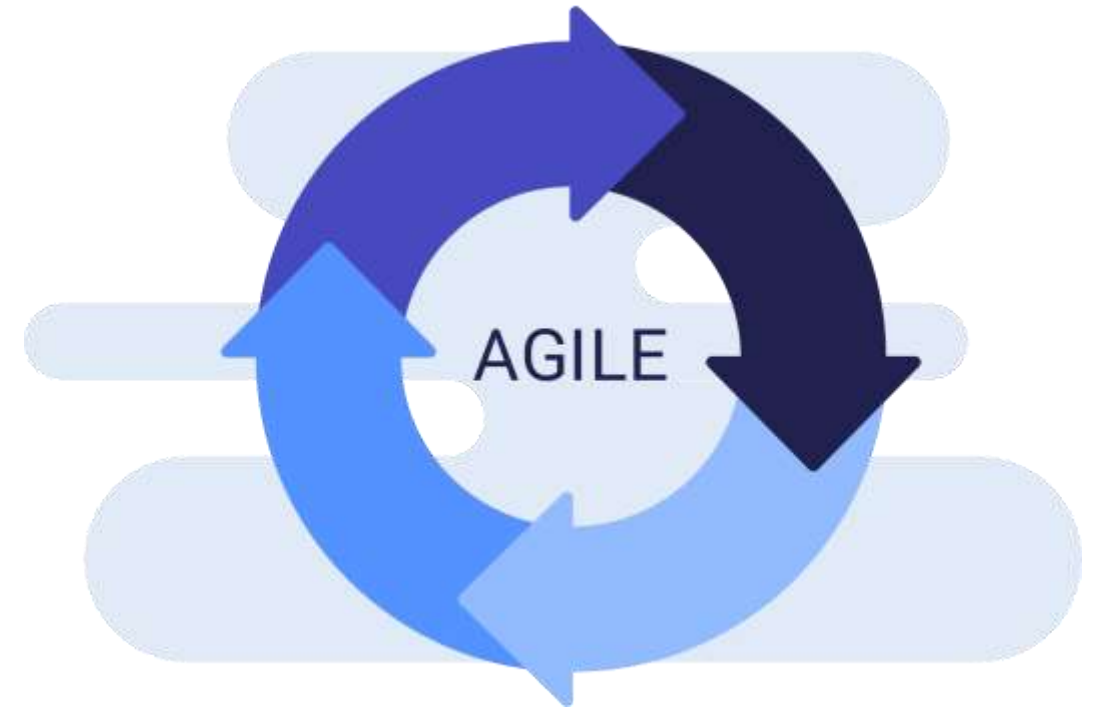


XP – XTREME PROGRAMING

- **Comunicação** frequente e adequada para a transferência de conhecimento, pois o desenvolvimento do software é um trabalho em equipe. Para isso, as melhores formas de comunicação são conversas presenciais e desenhos em um quadro branco;
- **Simplicidade** para garantir eficiência e foco apenas no que é necessário;
- **Feedback** para contar com o apoio dos clientes ao otimizar o produto e atingir os resultados;
- **Coragem** para tomar as atitudes corretas, como comunicar problemas, parar de fazer coisas pouco efetivas e tentar alguma nova abordagem, dando e aceitando feedbacks;
- **Respeito** entre todos os integrantes do time;
- **Mudança** como valor associado à flexibilidade e fator fundamental para realizar alterações nos produtos e nos requisitos, de acordo com as sugestões dos clientes.

DSDM – DYNAMIC SYSTEM DEVELOPMENT MODEL

- O método foi desenvolvido pelo consórcio DSDM em 1990, no Reino Unido, como uma extensão independente do Rapid Application Development (RAD). A intenção era usá-lo em projetos com limitação de prazos e orçamentos. Esse tipo de metodologia é caracterizado pelas práticas abaixo:
 - ciclo contínuo de melhoria;
 - constante colaboração entre cliente e equipe de desenvolvimento;
 - integração de funções entre os módulos.



FDD – FEATURE DRIVEN DEVELOPMENT

- Modelagem em objetos: construir diagramas básicos com os objetos para arquitetura do modelo do sistema;
- implementação orientada pelas características;
- uso de códigos de autoria individual;
- utilização de times para a implantação de cada característica;
- verificação da qualidade do código e do projeto;
- integração regular e predeterminada;
- manutenção de versões para gerenciar configurações;
- acompanhamento transparente do progresso do projeto.



LEAN SOFTWARE DEVELOPMENT

- O desenvolvimento de software enxuto é uma tradução dos princípios e práticas de manufatura enxuta para o domínio do desenvolvimento de software. Adaptado do Toyota Production System, está surgindo com o apoio de uma subcultura pró-lean dentro da comunidade ágil. Lean oferece uma sólida estrutura conceitual, valores e princípios, bem como boas práticas, derivadas da experiência, que dão suporte às organizações ágeis.

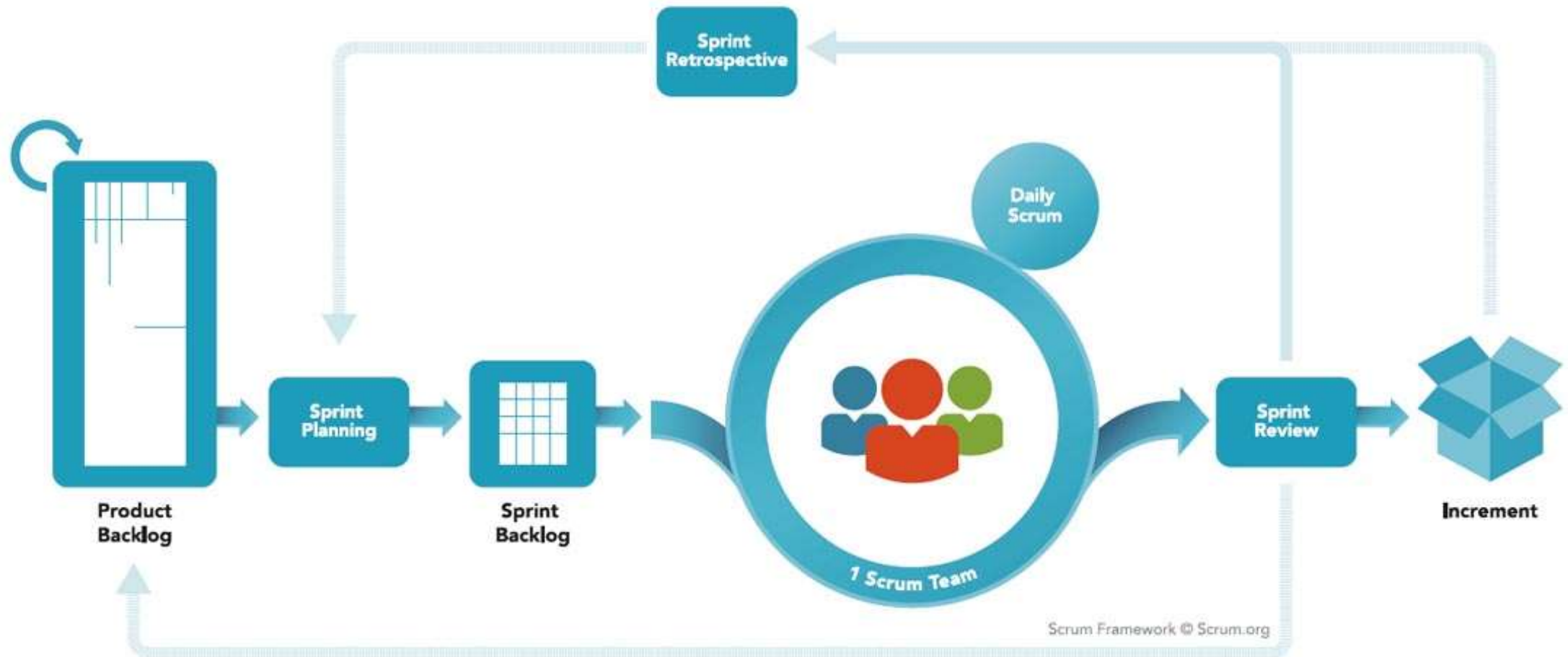


SCRUM

- Scrum é baseado no empirismo e LEAN thinking. O empirismo afirma que o conhecimento vem da experiência e da tomada de decisões com base no que é observado. O LEAN Thinking reduz o desperdício e se concentra no essencial.
- Scrum emprega uma abordagem iterativa e incremental para otimizar a previsibilidade e controlar o risco.



SCRUM FRAMEWORK



ATIVIDADE

- Realizar a leitura do Scrum Guide



BIBLIOGRAFIA

<http://agilemanifesto.org/iso/en/manifesto.html>

<https://www.scrum.org/resources/scrum-guide>

