***Metodologia Ágeis***

**Definição:** As metodologias ágeis são conjuntos de técnicas e práticas de gestão de projetos que visam acelerar o desenvolvimento e aumentar a eficiência e produtividade. Elas enfatizam a flexibilidade, a adaptação às mudanças e a colaboração, em vez de um planejamento rígido e detalhado. Existem vários exemplos de metodologias ágeis, cada uma com suas características e aplicações específicas.

Três exemplos de metodologias ágeis

**Scrum:** O Scrum é um framework ágil focado no gerenciamento de projetos complexos, permitindo que equipes entreguem valor de forma iterativa e incremental. Ele opera em ciclos fixos chamados Sprints, geralmente de 1 a 4 semanas, ao final dos quais um Incremento funcional e potencialmente utilizável do produto é entregue. O processo é guiado por três papéis principais: o Product Owner, que representa o cliente e gerencia o Product Backlog; o Time de Desenvolvimento, um grupo auto-organizável e multifuncional que constrói o produto; e o Scrum Master, que atua como líder servidor, garantindo que o time siga os princípios do Scrum e removendo impedimentos. O trabalho dentro de um Sprint é estruturado por eventos chave: o Sprint Planning define o que será feito e como; o Daily Scrum é uma reunião diária rápida para sincronização do time; a Sprint Review serve para inspecionar o Incremento e obter feedback; e a Sprint Retrospective foca na melhoria contínua do processo da equipe.

**Extreme Programming (XP):** O Extreme Programming (XP) é uma metodologia ágil que se concentra intensamente nas práticas de engenharia de software para produzir código de alta qualidade e responder rapidamente às mudanças nos requisitos. Baseado em valores como Comunicação, Simplicidade, Feedback, Coragem e Respeito, o XP prescreve um conjunto de práticas técnicas interligadas. Entre as mais conhecidas estão o Desenvolvimento Orientado a Testes (TDD), onde testes automatizados são escritos antes do código funcional; a Programação em Par, com dois desenvolvedores colaborando em um único computador para melhorar a qualidade e compartilhar conhecimento; a Refatoração contínua para manter o código limpo e fácil de modificar sem alterar seu comportamento; e a Integração Contínua (CI), onde o código é integrado e testado automaticamente várias vezes ao dia. Outras práticas importantes incluem Entregas Pequenas e frequentes para obter feedback rápido, um Cliente Presente trabalhando junto à equipe para esclarecer dúvidas, o Jogo do Planejamento para definir o trabalho colaborativamente, manter um Design Simples, aderir a Padrões de Codificação, praticar a Propriedade Coletiva do Código e trabalhar em um Ritmo Sustentável para evitar esgotamento. O XP busca a excelência técnica como meio para alcançar a agilidade e a satisfação do cliente.

Feito por: Rafael Costa de Lima, Vinícius Vicentini, Pietro Rodrigues. 1ºC, Etec MCM.