

Resenha – Managing Technical Debt (Steve McConnell)

O white paper *Managing Technical Debt*, escrito por Steve McConnell, é quase um manual de bom senso para quem trabalha com software no mundo real aquele onde prazos apertam, bugs aparecem e o código precisa “sair pra ontem”. Em vez de tratar o tema de forma acadêmica, McConnell fala direto: a dívida técnica é o preço que pagamos por atalhos no desenvolvimento.

A metáfora é simples e poderosa. Assim como uma dívida financeira, a dívida técnica pode ser útil quando bem administrada ajuda a entregar mais rápido, ganhar mercado, economizar tempo mas, se ignorada, cresce com juros e acaba sufocando o projeto.

McConnell divide a dívida em duas categorias principais. A não intencional, que nasce de descuidos, más práticas ou falta de experiência, e a intencional, quando a equipe decide conscientemente sacrificar um pouco de qualidade para cumprir um objetivo estratégico. Dentro dessa segunda, ele ainda diferencia entre dívida de curto prazo, que deve ser paga logo, e dívida de longo prazo, planejada e assumida com propósito.

O autor mostra que o perigo não está em dever, mas em não saber o quanto se deve. É aí que o time começa a “pagar juros” sem perceber gastando mais tempo mantendo o sistema do que criando novas funcionalidades. Ele sugere práticas simples para evitar isso, como manter uma lista visível das dívidas, registrar o esforço necessário para quitá-las e reservar tempo no cronograma para pagar o que foi adiado.

Outro ponto forte do texto é a comunicação. McConnell percebe que o vocabulário técnico muitas vezes não conecta com o pessoal de negócio, e propõe usar a própria metáfora financeira para criar empatia: dizer que “40% do orçamento está sendo gasto com juros de dívida técnica” é muito mais eficaz do que reclamar de “código legado difícil de dar manutenção”.

Ele também chama atenção para um erro comum: tratar decisões técnicas como se só houvesse duas opções “fazer rápido” ou “fazer certo”. Às vezes existe um meio-termo, o “rápido, mas limpo”, que entrega valor sem comprometer o futuro.

No fim das contas, McConnell passa uma mensagem madura: dívida técnica é inevitável, mas não é um inimigo. O problema não é dever, e sim perder o controle da dívida. Saber quando assumir, como registrar e quando pagar é o que diferencia equipes maduras de times que vivem apagando incêndios.

Leve, direto e cheio de exemplos práticos, *Managing Technical Debt* é leitura essencial para quem quer construir software sustentável aquele que continua evoluindo sem colapsar sob o peso das próprias pressas.