

Resenha – “Documenting Architecture Decisions” (Michael Nygard, Cognitect, 2011)

No artigo *Documenting Architecture Decisions*, Michael Nygard discute um dos maiores desafios no desenvolvimento de software moderno: **como registrar e justificar decisões arquiteturais sem criar burocracia**. Ele parte da observação de que, em projetos ágeis, as decisões de arquitetura não são tomadas todas de uma vez — elas evoluem conforme o sistema cresce e novas informações aparecem. Isso cria um problema comum: a perda de memória técnica. Com o tempo, ninguém mais lembra *por que* certas escolhas foram feitas, o que leva a retrabalho e decisões repetidas ou contraditórias.

Para resolver isso, Nygard propõe uma abordagem simples e pragmática: os **ADRs (Architecture Decision Records)**. Em vez de longos documentos, cada ADR é um pequeno registro — geralmente uma página — que descreve **uma única decisão arquitetural**, com seu **contexto, a escolha feita e as consequências**. O autor sugere manter esses registros dentro do próprio repositório do código, numerados sequencialmente, de modo que possam evoluir junto com o projeto. Quando uma decisão é alterada, o ADR antigo não é apagado, mas marcado como “substituído”.

O ponto mais forte do texto é sua **visão prática e realista** sobre documentação. Nygard não defende o abandono da documentação, mas sim que ela deve ter **valor e propósito**. Em vez de documentos extensos e esquecidos, ele propõe algo leve, direto e histórico — uma narrativa das decisões que moldam o sistema ao longo do tempo. Essa ideia ajuda times ágeis a manter clareza arquitetural sem abrir mão da flexibilidade.

Além de ser simples, o formato dos ADRs incentiva a reflexão: antes de tomar uma decisão, o time precisa articular claramente o contexto e as consequências, o que melhora a qualidade das escolhas. Por outro lado, Nygard reconhece que há limitações — como a dificuldade de envolver stakeholders não técnicos, já que os ADRs ficam normalmente no Git. Ainda assim, o benefício para a equipe técnica é evidente: **um registro vivo e versionado da evolução da arquitetura**.

Em síntese, o artigo é uma leitura fundamental para arquitetos e desenvolvedores que desejam equilibrar agilidade com responsabilidade técnica. Nygard mostra que documentar não é sobre burocracia, mas sobre preservar conhecimento coletivo. O conceito de ADR se tornou, desde então, uma prática amplamente adotada em projetos de software modernos, justamente por unir **simplicidade, rastreabilidade e clareza**.