
Prefácio

Um de nossos doces favoritos aqui na Dinamarca é o Ga-Jol, cujos fortes vapores de licorice são um complemento perfeito para nosso clima úmido e, geralmente, frio. Parte do charme do Ga-Jol, para nós dinamarqueses, são os dizeres sábios impressos em cada caixa. Comprei dois pacotes dessa iguaria essa manhã e nela veio este antigo ditado dinamarquês: *Ærlighed i små ting er ikke nogen lille ting*.

“Honestidade em pequenas coisas não é uma coisa pequena”. Era um bom presságio para com o que eu já desejava dizer aqui. Pequenas coisas são importantes. Este é um livro sobre preocupações modestas cujos valores estão longe de ser pequenos.

Deus está nos detalhes, disse o arquiteto Ludwig Mies van der Rohe. Essa citação retoma argumentos com contemporâneos sobre o papel da arquitetura no desenvolvimento de software, especialmente no mundo Agile. Bob e eu, às vezes, acabávamos engajados entusiasmadamente debatendo sobre este assunto. E sim, Mies van der Rohe atentava para os utilitários e as formas imemoriais de construção que fundamentam uma ótima arquitetura. Por outro lado, ele também selecionava pessoalmente cada maçaneta para cada casa que ele projetava. Por quê? Por que pequenas coisas são importantes.

Em nosso “debate” sobre Desenvolvimento dirigido a testes (TDD, sigla em inglês), Bob e eu descobrimos que concordamos que a arquitetura do software possuir um lugar importante no desenvolvimento, embora provavelmente tenhamos perspectivas diferentes do significado exato disso. Essas diferenças são relativamente irrelevantes contudo, pois podemos admitir que profissionais responsáveis dedicam algum tempo para pensar e planejar o início de um projeto. As noções de desenvolvimento dirigido apenas por testes e por códigos do final da década de 1990 já não existem mais. Mesmo assim, a atenção aos detalhes é um fundamento de profissionalismo ainda mais crítico do que qualquer visão maior. Primeiro, é por meio da prática em pequenos trabalhos que profissionais adquirem proficiência e confiança para se aventurar nos maiores. Segundo, a menor parte de uma construção desleixada, a porta que não fecha direito ou o azulejo levemente torto do chão, ou mesmo uma mesa desarrumada, retiram completamente o charme do todo. É sobre isso que se trata o código limpo.

Ainda assim, a arquitetura é apenas uma metáfora para o desenvolvimento de software, especialmente para a parte que entrega o produto inicial no mesmo sentido que um arquiteto entrega uma construção imaculada. Nessa época do Scrum e do Agile, o foco está em colocar o