

OPINIÃO



Filipe Nuno
Carlos
LÍDER DA WINTRUST,
E PRESIDENTE
DA ASSOCIAÇÃO
PORTUGUESA
DE TESTES
DE SOFTWARE

OS TESTADORES PRECISAM DE PROGRAMAR?



Quantas vezes não se ouviram profissionais a dizer que optaram pela carreira de testes porque não gostavam de programar? Será que esta frase pode ocorrer nos dias de hoje?

Começando por um enquadramento histórico, lembro-me perfeitamente que, há 15 anos, bastava ter um profissional paciente e com espírito de “detetive” para poder detetar defeitos. Estamos a falar de um tempo em que as aplicações evoluíram de um modelo monolítico para o modelo distribuído.

O nível de falha numa arquitetura distribuída aumentava devido à incorporação de peças de diversos fabricantes. Quem não se lembra dos “drivers ODBC”

que não faziam o que era suposto e a aplicação assumia pressupostos não cumpridos?

Quem não se lembra de querer integrar uns relatórios realizados em Crystal Reports e, sabe-se lá porquê, o documento não produzia os resultados que se pretendia? Neste contexto, só o facto de haver alguém que testasse as funcionalidades, nem que fosse de forma aleatória, já entregava valor ao projeto.

Entretanto a maturidade do desenvolvimento foi aumentando, e já não se descobriam defeitos de forma tão fácil. Tínhamos que ir à procura deles.

O passo seguinte passou por conceber (desenhar) os testes de uma forma sistemática, aplicando técnicas como a Análise de Valor Fronteira, Tabelas de Decisão ou Transição de Estados. Surgem os primeiros cursos de testes e nascem esquemas de certificação como o ISTQB em 2002.

Neste aumento de exigência, os profissionais com pouca base tecnológica / matemática começam a claudicar. O seu grau de abstração precisa de ser desenvolvido para poderem ler uma especificação de requisitos, aplicar as técnicas referidas e depois conceber os testes que melhor possam diminuir o risco. Passamos de profissionais que podiam apenas aplicar o lado direito do cérebro (lado criativo) para profissionais que precisam de aplicar o lado esquerdo do cérebro (lado racional).

Entretanto, a complexidade dos sistemas continua a aumentar e o teste manual de per-si começa a ser demasiado dispendioso para se ter a cobertura de teste desejada. Para além do teste ser mais complexo devido às funcionalidades serem mais abrangentes, a mesma aplicação pode ser accedida de diversos dispositivos.

Como se isto não bastasse, assiste-se ao movimento DevOps, ou seja, o tempo que medeia entre a alteração aplicacional e a sua entrada em produção deve medir-se em minutos e não em dias.

Por estes factos, a automação de testes constitui uma tendência. E agora voltamos à pergunta inicial. É preciso ser programador para abraçar esta nova tendência? A resposta é “não, não é preciso, mas se for, dá jeito”. E o mais curioso é que os programadores viam a atividade de teste como algo enfadonho e agora veem como algo interessante.

Em suma, aquilo que se deve procurar numa equipa de teste, de forma a poder incorporar os níveis de automação que se exigem sem excluir os profissionais existentes, é ter uma equipa de programadores a criar uma boa base (framework de automação), que seja fácil de usar pelos outros testadores (na prática a “programação” são comandos iguais aos usados no teste manual).

Esta equipa de não programadores deverá ter profissionais com base matemática (lado esquerdo do cérebro) e profissionais sem base matemática (lado direito do cérebro). Estes últimos deverão ser em menor número, pois o seu papel é mais determinante na vertente comportamental do que na vertente sistemática. Para a vertente sistemática, um matemático ou engenheiro é mais eficiente. Em oposição, na vertente comportamental, os não-matemáticos são melhores a gerirem um conflito, a apoiarem os testes de aceitação junto dos utilizadores finais ou a efetuarem testes exploratórios. ●

A AUTOMAÇÃO DE TESTES CONSTITUI UMA TENDÊNCIA. É PRECISO SER PROGRAMADOR PARA ABRAÇAR A NOVA TENDÊNCIA? A RESPOSTA É “NÃO, NÃO É PRECISO, MAS SE FOR, DÁ JEITO”

