



Por que falha tanto o software?

A indústria informática sempre esteve no topo da inovação, mas o desenvolvimento de *software* empresarial continua a ter uma taxa de sucesso incompreensivelmente baixa

Texto João Ramos

O **S RELATÓRIOS CHAOS** da consultora Standish Group têm revelado alguns dados pouco animadores e que deixam os diretores dos departamentos de informática (CIO – *chief information officer*) com os cabelos em pé e os engenheiros de *software* e programadores acalorados.

O ano 2009 foi mesmo o pior da última década na qualidade de *software*. Apenas 32% dos projetos de *software*, segundo esta consultora, foram concluídos com sucesso (dentro do prazo, sem ultrapassar o orçamento e de acordo com as funcionalidades solicitadas).

O pior é que 24% dos projetos contratados falharam totalmente e nunca chegaram a ver “a luz do dia”. E há uma maioria (44%) que foi entregue com algum tipo de problema: surgiu fora do prazo, terminou acima do orçamento ou não respondia a todos os requisitos exigidos.

Um preocupante défice da qualidade média do *software* que, segundo a Standish Group, atinge quase todos os sectores de atividade, designadamente a banca e a administração pública.

O ERRO HUMANO

Afinal, como é possível que dois terços dos projetos de *software* tenham acidentes de percurso, quando as ferramentas de progra-

mação e os computadores atingiram o maior grau de sofisticação de sempre?

Gonçalo Quadros, presidente executivo da Critical Software, não tem dúvidas de que o fator mais importante é “o erro humano”. “Este tipo de erro que é tão mais importante quanto mais complexo o projeto for”, acrescenta o fundador da Critical Software, tecnológica que se rege pelos mais elevados padrões de qualidade.

SER AUTODIDATA NÃO CHEGA

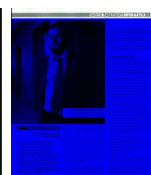
Na opinião de Fernando Miguel Carvalho, professor-adjunto no curso de Engenharia Informática do Instituto Superior de Engenharia de Lisboa (ISEL) a taxa de insucesso no desenvolvimento de *software* em Portugal deverá ser semelhante ao que a Standish Group estima existir a nível mundial. “O erro humano resulta, em grande parte, do facto de as equipas de desenvolvimento não utilizarem (ou utilizarem de uma forma inadequada) as novas metodologias e processos de desenvolvimento inovadores que agilizam e melhoram a qualidade da produção de *software*.” Ou seja, os programadores não utilizam essas técnicas de engenharia de *software* porque lhes “faltam competências para as utilizar”, defende o docente do ISEL.

Esta lacuna fez com que Fernando Carvalho e o Centro de Cálculo do ISEL tivessem decidido criar a pós-graduação de Desenvolvimento de Software (Prompt) que já teve os primeiros alunos no primeiro semestre



de 2011. “Os paradigmas da programação evoluíram para novos conceitos que já não estão ao alcance de qualquer autodidata”, refere o docente do ISEL.

Gonçalo Quadros concorda que “a qualificação das pessoas e das equipas é uma condição essencial” para que haja projetos informáticos de qualidade, mas também diz que “não é suficiente”. “À medida que aumenta a dimensão e complexidade do projeto – o número de pessoas envolvidas e a duração –, é importante também instituir práticas e procedimentos que garantam que os desafios são endereçados de forma adequada através de um sistema de gestão da qualidade”, defende o fundador da tecnológica de Coimbra. E sustenta que o projeto não pode estar dependente de uma única



Margarida Gonçalves e Pedro Henriques (Strongstep): "É preciso escolher a metodologia certa"

→ QUALIDADE É PREOCUPAÇÃO

36% das empresas portuguesas admitem os defeitos

Um estudo desenvolvido pela Strongstep (entre Fevereiro e Abril de 2011) revela que há um nível crescente de consciência para as questões da qualidade entre os setores chave da atividade económica portuguesa (finanças, telecomunicações, saúde, TI energia e administração pública). 73% dizem ter introduzido uma cultura de qualidade e 82% têm um staff dedicado ao assunto. Mesmo assim, 36% dizem ter encontrado defeitos no desenvolvimento depois da realização de testes internos. O estudo revela ainda que as metodologias CMMI/TSP e Agile são as que dominam em Portugal.

peessoa. "O projeto não pode depender do estado de espírito de alguém em particular." Por isso, é que a Critical Software, que tem clientes nos exigentes mercados do aeroespacial e defesa, instituiu um sistema de gestão da qualidade que é certificado pelas normas mais exigentes a nível mundial por um conjunto de siglas (indecifráveis para o comum dos cidadãos), como por exemplo: CMMI nível 5, EN 9100 e TickIT (além da mais comum ISO 9001). E os seus programadores estão na crista da onda da tecnologia de programação (usam PMP, Six Sigma, Scrum e Istqb).

Pedro Castro Henriques, fundador e diretor da Strongstep, um *spin off* da Faculdade de Engenharia da Universidade do Porto especializado na engenharia de *software*, ad-

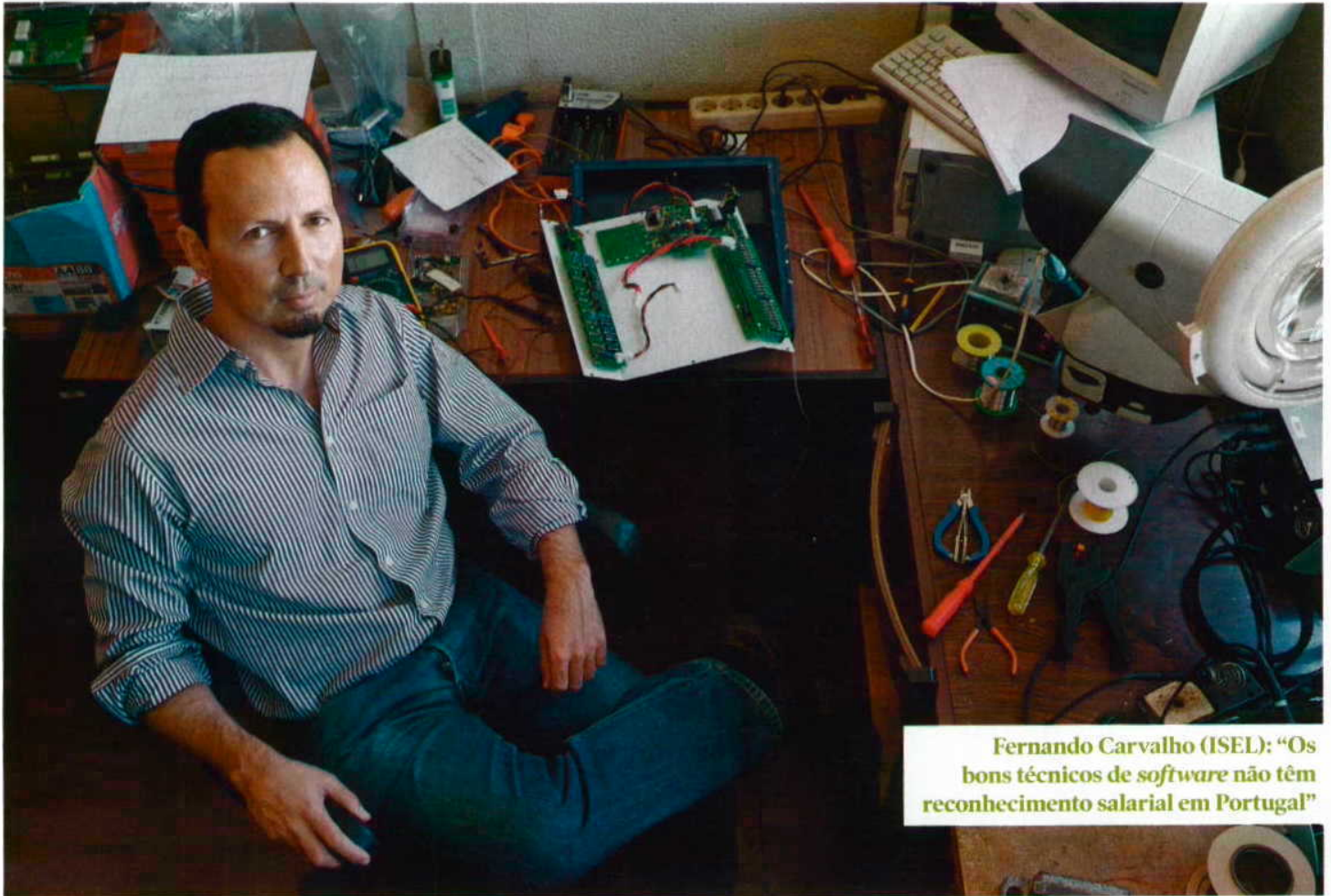
mite que os CIO e empresários não têm hoje uma tarefa fácil quando chega o momento de escolher as metodologias de programação. "Tudo depende do tipo de empresa, sector e dimensão", considera. Aconselha por isso que se devem "utilizar as munições certas para os seus objetivos, não seguir modas, mas, sim, as metodologias (as tais que têm nomes indecifráveis) que tragam mais benefício para o seu caso concreto".

ÁGIL GANHA AO ARTESANAL

A ideia de criar aplicações de *software* depressa e bem não é de agora. Já na década de 80, quando a área de engenharia de *software* teve um grande crescimento, surgiu a moda das ferramentas CASE (*computer-aided software engineering*). Mas com o advento da internet, as CASE acabaram por ser postas de lado. Começou a falar-se em meados dos anos 90 nas ferramentas de *software* ágil como forma de os programadores fazerem ou programas em tempo útil e fugirem aos erros e aos problemas de manutenção que surgiam nas aplicações feitas nas linguagens por objetos (Java) e plataformas maioritárias no mercado (.Net da Microsoft). "Criava-se código a metro e os projetos eram longos ou... nunca mais acabam", recorda Paulo Rosado, presidente executivo da Outsystems, tecnológica portuguesa especialista em *software* ágil. "Criou um 'complicómetro' e fez-se tábua rasa do conhecimento de *software* que tinha sido criado anteriormente. Depois apareceu o *outsourcing* para a Índia que só veio complicar ainda mais. Tudo isto veio criar uma situação caótica e de crise que se repercutiu na falta de qualidade descomunal nos sistemas de informação empresariais", acrescenta Paulo Rosado.

Na prática, os programadores estavam sempre a reinventar a roda e o pesadelo de manutenção nos anos seguintes avolumava-se porque os sistemas envelheciam a uma grande rapidez. Como o negócio estava sempre a mudar, a tecnologia também tinha de se adaptar a estas mudanças.

Foi para acelerar o processo de desenvolvimento, em oposição com método artesanal, que Paulo Rosado lançou, em 2002, a Outsystems para apostar nas metodologias ágeis. "Em relação à programação tradicional (em cascata), a metodologia ágil tem a vantagem ser feita através de interações. Resolve o problema aos bocados e mantém



Fernando Carvalho (ISEL): "Os bons técnicos de software não têm reconhecimento salarial em Portugal"

ligação com o negócio (através de um *feedback* constante", explica Paulo Rosado.

Gonçalo Quadros também defende que "as metodologias ágeis são uma poderosa resposta aos desafios quando se sabe à partida que vamos ter de ajustar em contínuo as características ou funcionalidades de uma aplicação. A sua utilização integrada com práticas mais formais tem resultados muito bons!"

EVITAR O PESADELO

Gonçalo Quadros também diz ser importante "planear de forma consciente e consistente", e que "a qualidade nunca não pode ser 'injetada' no final", pois "o esforço e o custo para corrigir o que está mal aumentam drasticamente quanto mais tarde forem concretizados". O líder da Critical Software dá um exemplo que resulta da experiência de muitos anos: "O esforço que demora a remover defeitos durante o início do ciclo de vida de desenvolvimento é oito vezes superior do que se a remoção do mesmo defeito for feita perto do final do ciclo de vida." Pior ainda, segundo Gonçalo Quadros é se

os erros surgirem antes dos chamados testes de aceitação (últimos testes, feitos antes da entrega final). Nesse caso, o esforço de correção é 20 vezes maior. "O aumento do custo e esforço aumentam muito mais se o defeito é deixado passar para os sistemas em exploração", acrescenta.

Em suma, as boas práticas de qualidade não atrasam, antes antecipam, o desenvolvimento de *software*. "São um insubstituível e poderoso fator de mitigação de risco, ou seja, de previsibilidade e de capacidade de antecipação – sem os quais, os projetos rapidamente se transformam em pesadelos descontrolados", defende.

PORTUGAL DÁ CARTAS

O desenvolvimento de *software* também falha em Portugal, mas não será por falta de qualidade do ensino. "Neste momento temos universidades que ensinam Engenharia de Software de topo aos seus alunos, sendo estes muitas vezes recrutados por empresas como a Google, Microsoft, Yahoo!, etc., e hoje vêm dos países nórdicos recrutar os nossos melhores finalistas e mestres", refere

Pedro Henriques. "Este fenómeno ocorre pela escassez existente nesses países, mas também por reconhecerem que os nossos engenheiros de *software* estão no topo a nível do conhecimento e capacidade, e também pela proximidade cultural e geográfica", acrescenta o fundador da Strongstep.

Um dos problemas parece ser a falta de reconhecimento salarial que os quadros técnicos mais qualificados se defrontam nas empresas portuguesas. "Quando atingem a maturidade, a evolução da carreira e as remunerações mais atrativas são nas áreas de gestão e comercial. Muitas vezes, perde-se um excelente programador e não se ganha um bom gestor ou especialista", defende Fernando Carvalho. "As grandes multinacionais de *software* sabem reter os seus melhores talentos de engenharia de *software*", sublinha o docente do ISEL.

Gonçalo Quadros não tem dúvidas em afirmar que "existem excelentes engenheiros de *software* em Portugal», mas admite que devíamos estar melhor no indicador quantidade e que as universidades "deviam dar mais atenção curricular à qualidade". ■