Artigo

Seção 6, aula 39

O custo efetivo dos bugs no desenvolvimento de software

**Quanto custa a correção de bugs no desenvolvimento de sistemas? Depende de quão tarde você irá descobri-los!**

 A máxima que diz que a prevenção é melhor do que a cura não vale só na medicina! A frase do erudito holandês DESIDERIUS ERASMUS é bem oportuna também na indústria de desenvolvimento de sistemas.

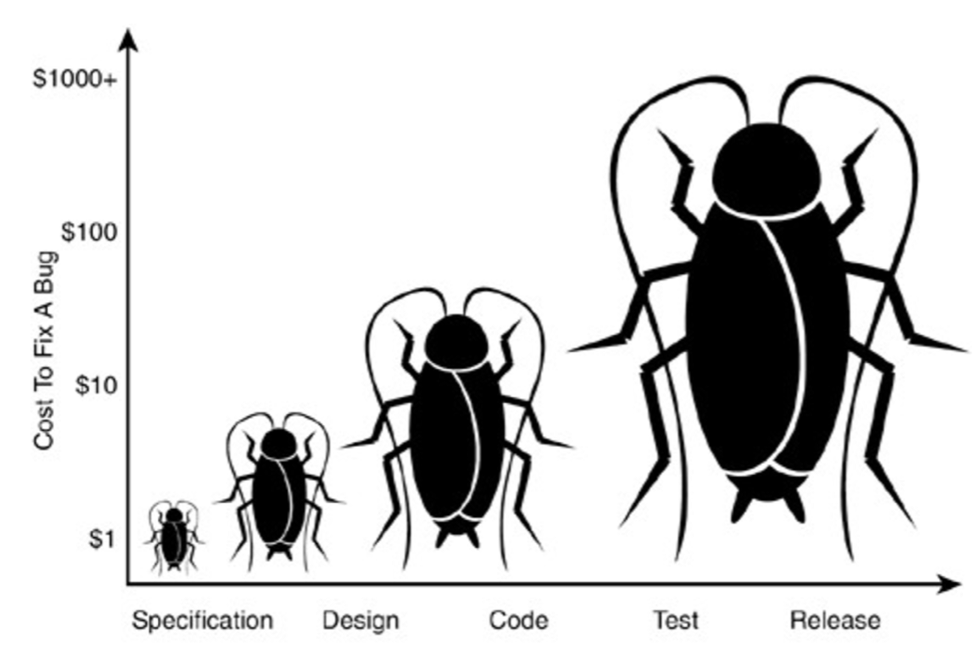
Veremos que o custo de corrigir um bug é muito maior do que criar mecanismos para evitá-los.

Entenda como bug tudo que é um defeito ou problema, desde a concepção até o pleno uso de um sistema. Programadores por aí também o chamam de pau, probleminha, usuário, negocinho, e por aí vai (risos).

É provável que esteja se lembrando das frases clássicas de um desenvolvedor quando você encontra um problema e ouve: “no meu computador funciona”, “você está fazendo alguma coisa errada” ou “aperta F5”...

Por mais engraçado que isso pareça, situações como estas ocultam problemas que se não forem resolvidos rapidamente sairão muito caro, não só financeiramente, como também para a imagem de sua empresa e de seu time de desenvolvimento.

Na medicina, curar é muito mais caro e traumático do que prevenir. Em tecnologia não é diferente: quanto mais cedo um bug for encontrado mais fácil e menos impactante é sua correção.



O problema é que a evidência tem mostrado que defeitos em softwares é um fator inerente na indústria de tecnologia. Um artigo da Coralogix relata que 75% do tempo dos desenvolvedores estão voltados em debugar código para descobrir e consertar problemas.

A estimativa é de que a cada 1000 linhas de código criadas, 70 novos bugs são criados e 15 serão descobertos pelo seu cliente. E se isso de fato acontecer, cada bug corrigido em sistemas que estão em produção custará 30 vezes mais (tempo e dinheiro) para serem corrigidos do que se tivessem sido descobertos na fase de desenvolvimento.

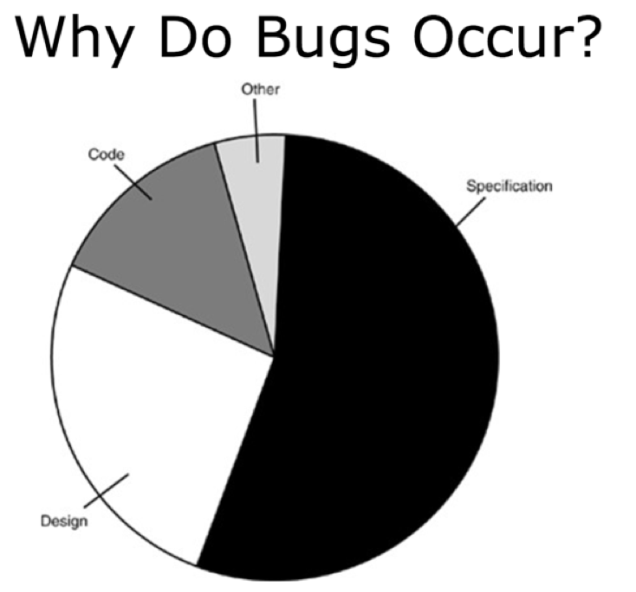
Porém, alguns podem custar bem mais:

* (2007) - ARQUIVOS DO PROJETO FBI VIRTUAL: Custo de US$170 milhões.
* (2004) - ERRO EM SOFTWARE DO REINO UNIDO DESTRÓI REGISTROS FISCAIS: Custo de no mínimo £85 milhões.
* (2004) - PROJETO DE SOFTWARE DA FORD E ORACLE DESABA: Custo US$400 milhões.
* (1962) - FALHA EM FOGUETE POR FALTA DE UM HÍFEN: Custo de US$135 milhões.

Aqui na Interaktiv discutimos frequentemente sobre testes automatizados e seus benefícios e o quão importante eles são para a causa. Como focamos no desenvolvimento web e mobile estão sempre no nosso radar:

* Teste de aplicativos Web
* Testes de aplicativos móveis
* Testes de performance
* Testes de serviços (transações e regras de negócio)
* Testes de código (BDD/TDD)

Entendemos que o problema é "mais embaixo" e não são apenas os desenvolvedores os únicos causadores de bugs e defeitos em sistemas. Na verdade, o desenvolvimento representa apenas uma parte dos problemas, como vemos no gráfico abaixo:



Há um consenso de todas as partes envolvidas no processo (incluindo o cliente) de que não há nada mais nocivo para um projeto do que uma especificação mal feita. A falta de organização no levantamento de requisitos, subdimensionamento das demandas e detalhamento superficial é uma bomba relógio que vai explodir mais cedo ou mais tarde.

Portanto, não há uma solução de varinha mágica. É necessário fazer parte da cultura da empresa processos que previnam defeitos, e isso começa do começo. É necessário traduzir os requisitos do cliente de forma clara para que a arquitetura da solução, a revisão do código e os testes tenham base concreta para minimizar os problemas que certamente serão descobertos em outras fases do projeto.

Crie uma base de dados de problemas e discuta com o seu time as lições aprendidas. Escolha uma metodologia ágil para gerenciar o fluxo do projeto e motive toda a equipe a minimizar os erros. Porém, quando os acharem, incentive-os a não jogá-los debaixo do tapete, mas resolvê-los o quanto antes. Isso certamente economizará algumas noites de sono de todos.

Encontre mecanismos de medição para analisar a quantidade de bugs que entram e saem. Analise as recorrências e reúna a equipe para discutir como evitá-las. Os resultados não virão de um dia para o outro, mas quem definirá a velocidade dessa transição será a capacidade que você e sua equipe têm de aprender com os erros, ou melhor, com os bugs.

**André Cripa Rodrigues**

(Sócio fundador da Interaktiv)