

RESUMO DA AULA

01 Introdução aos Testes Automatizados de Software

Nesta seção, vamos explorar o conceito de testes automatizados de software e entender como eles são aplicados no desenvolvimento para garantir a qualidade do código e prevenir falhas inesperadas. O que são os testes automatizados? Quando falamos em testar algo, significa garantir que determinada funcionalidade esteja funcionando corretamente. No desenvolvimento de software, esse conceito é fundamental para validar que o sistema se comporta conforme o esperado. Imagine um técnico de informática consertando um computador. Após realizar o conserto, ele liga o computador para verificar se o problema foi realmente resolvido. Esse processo de verificação é um teste. Da mesma forma, ao desenvolvermos ou corrigirmos uma funcionalidade em um sistema, precisamos testá-la. Esses testes podem ser manuais ou automatizados. No caso de testes manuais, o desenvolvedor interage com o sistema — por exemplo, ao corrigir um botão que parou de funcionar em uma página. Ele abre o site, clica no botão e verifica manualmente se o problema foi solucionado. Embora seja uma abordagem válida, ela se torna inviável em sistemas complexos e com várias funcionalidades interdependentes. Testes automatizados: o que são? Testes automatizados são uma forma de automatizar a verificação das funcionalidades do sistema por meio de código. Em vez de testar manualmente, criamos scripts que testam automaticamente se cada parte do sistema está funcionando corretamente, incluindo casos de sucesso e cenários de falha. Por exemplo, quando fazemos uma alteração no código, podemos configurar uma série de testes automatizados para verificar se o que foi modificado não quebrou outras partes do sistema. Esses testes são escritos uma vez e podem ser executados quantas vezes forem necessárias, sempre que uma mudança é feita no código. Dessa forma, os testes automatizados garantem que os cenários críticos estão cobertos, fornecendo maior confiança no desenvolvimento. Por que não apenas testar manualmente? Testes manuais funcionam bem para aplicações menores, mas se tornam inviáveis em sistemas grandes e complexos. Imagine um portal de banco, que possui inúmeras funcionalidades:

transferência de dinheiro, pagamento de boletos, consulta de saldo, investimentos, entre outros. Cada uma dessas funcionalidades depende de diversos trechos de código e serviços interconectados. Testar manualmente cada uma delas a cada atualização se tornaria uma tarefa demorada, cara e propensa a falhas humanas. O problema dos testes manuais em sistemas grandes: Suponha o seguinte cenário: Um bug reportado no formulário de transferência bancária. Você localiza e corrige o problema no código. Em seguida, testa manualmente o formulário para verificar se a transferência funciona. Aparentemente, a correção está correta e a transferência realizada com sucesso. Você então publica a correção no ambiente de produção. No entanto, meia hora depois, usuários começam a relatar que a funcionalidade de pagamento de boletos parou de funcionar. Após investigar, você descobre que a alteração feita no formulário de transferências afetou outras funcionalidades do sistema que compartilham o mesmo código. Isso conhecido como regressão — uma falha que surge em um ponto não relacionado ao código alterado inicialmente. Por que os testes automatizados são importantes? Os testes automatizados resolvem esse problema, pois são capazes de identificar rapidamente se uma alteração impactou outras partes do sistema. Se houvesse uma bateria de testes cobrindo a funcionalidade de pagamento de boletos, ela teria apontado o problema antes de a correção ser aplicada em produção. Testar manualmente todo o sistema a cada modificação não é uma abordagem escalável. Em sistemas grandes, testar a aplicação dessa forma seria praticamente impossível e consumiria um tempo enorme de desenvolvimento. Mas não é possível testar manualmente esses pontos de falha? Imprevisibilidade de impacto: Quando você corrige uma funcionalidade, é difícil prever quais outras partes do sistema podem ser afetadas, especialmente em sistemas grandes e complexos. Impossível manter todas as informações de dependências na memória. Baixa produtividade: Mesmo que você soubesse exatamente quais partes do sistema precisam ser testadas, sair "caçando" todos os pontos e testando manualmente não é produtivo. Em projetos grandes, isso poderia levar dias ou semanas. Falha humana: Confiar apenas na memória e no teste manual é arriscado. A chance de esquecer um caso de uso ou não perceber uma falha é alta. Por isso, os testes automatizados são essenciais. Eles não substituem o teste manual, mas complementam, trazendo benefícios como: Maior confiança no código: Com uma

bateria de testes automatizados, sabemos que cada funcionalidade crítica está coberta. Redução de bugs em produção: Problemas são detectados antes de o código ser publicado. Prevenção de regressões: Garantia de que novas alterações não quebrem funcionalidades existentes. Melhor compreensão do sistema: Ao escrever testes, compreendemos melhor o fluxo do software. Paz de espírito: Podemos focar no desenvolvimento sem temer que alterações quebrem o sistema. Resumo Testes manuais são limitados e propensos a erros humanos. Testes automatizados verificam se cada parte do sistema se comporta conforme o esperado, mesmo após alterações. Eles garantem cobertura para cenários de sucesso e falha, evitando regressões e aumentando a confiança no código. Em sistemas grandes, os testes automatizados são a única forma viável de garantir a qualidade do software.
