

Educação Profissional Paulista

**Técnico em
Desenvolvimento
de Sistemas**

Secretaria da
Educação



SÃO PAULO
GOVERNO DO ESTADO

Testes funcionais e não funcionais

Introdução aos testes de software e tipos de testes

Aula 2

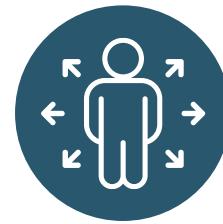
Código da aula: [SIS]ANO1C3B1S2A2

Exposição



Objetivo da aula

Conhecer a diversidade existente nos testes de software;
Explorar os diferentes tipos e possibilidades de aplicação.



Competências da Unidade (técnicas e socioemocionais)

- Desenvolver soluções de software (*back-end, front-end e full-stack*) utilizando técnicas, métodos, ferramentas e linguagens de programação diversas;
- Executar manutenção de programas;
- Resolução de problemas.



Recursos didáticos

- Caderno para anotações;
- Recurso audiovisual para exibição de vídeos e imagens;
- Acesso ao laboratório de informática e/ou internet.



Duração da aula

50 minutos

Exposição

Conhecer a diversidade nos testes de software: explorando os diferentes tipos

Conhecer os diferentes tipos de testes de software ajuda a criar uma estratégia abrangente, que avalia desde funcionalidades até desempenho e segurança. Isso assegura a construção de um produto final sólido e confiável.

- Aplicar a distinção entre testes funcionais e não funcionais;
- Comparar metodologias e ferramentas para testes automatizados e manuais;
- Conhecer testes de regressão: importância e implementação de testes de regressão para garantir que as novas mudanças não afetem as funcionalidades existentes.

Exposição

Distinção entre testes funcionais e não funcionais

	O que são?	Exemplo
Testes funcionais	Projetados para verificar se o comportamento de um sistema está de acordo com os requisitos funcionais especificados, ou seja, se concentram em testar as funcionalidades do software.	Se estivermos testando um aplicativo de e-commerce, os testes funcionais irão verificar se a adição de itens ao carrinho, o cálculo do total, a aplicação de cupons de desconto e o <i>checkout</i> funcionam conforme especificado.
Teste não funcionais	Referem-se à avaliação dos aspectos não funcionais do sistema de software , como desempenho, usabilidade, confiabilidade e segurança.	Nos mesmos testes para um aplicativo de e-commerce , os testes não funcionais incluiriam a verificação da capacidade do site de lidar com um alto volume de acessos simultâneos (teste de carga), avaliar a segurança do processamento do pagamento e a facilidade com que os novos usuários conseguem navegar e realizar uma compra (usabilidade).

Comparação das metodologias e ferramentas para testes automatizados e manuais

Exposição

O que são?	Características
Testes automatizados <p>Utilizam ferramentas de software para executar testes predefinidos sobre o aplicativo, sem intervenção humana contínua. Eles são essenciais para tarefas repetitivas e podem ser executados rapidamente.</p> <p>As ferramentas populares incluem Selenium, que simula interações do usuário em navegadores web, e JUnit para testes unitários em aplicações Java.</p>	Mais rápidos e eficientes para grandes volumes de código, podem ter dificuldades em lidar com novos cenários de teste ou mudanças no aplicativo que não foram previamente codificados.
Testes manuais <p>Realizados por testadores que executam manualmente os passos de teste, sem o auxílio de ferramentas de automação. Eles são úteis para explorar cenários de teste mais complexos ou quando o teste requer intuição humana, como a avaliação da experiência do usuário.</p>	Mais lentos, mas oferecem a flexibilidade de adaptação e identificação de problemas inesperados.

Fonte: ROCHA, 2023. Elaborado especialmente para curso.



© Getty Images

Exposição



Vamos aprofundar um pouco mais sobre como um teste de software pode ser feito em uma aplicação real?

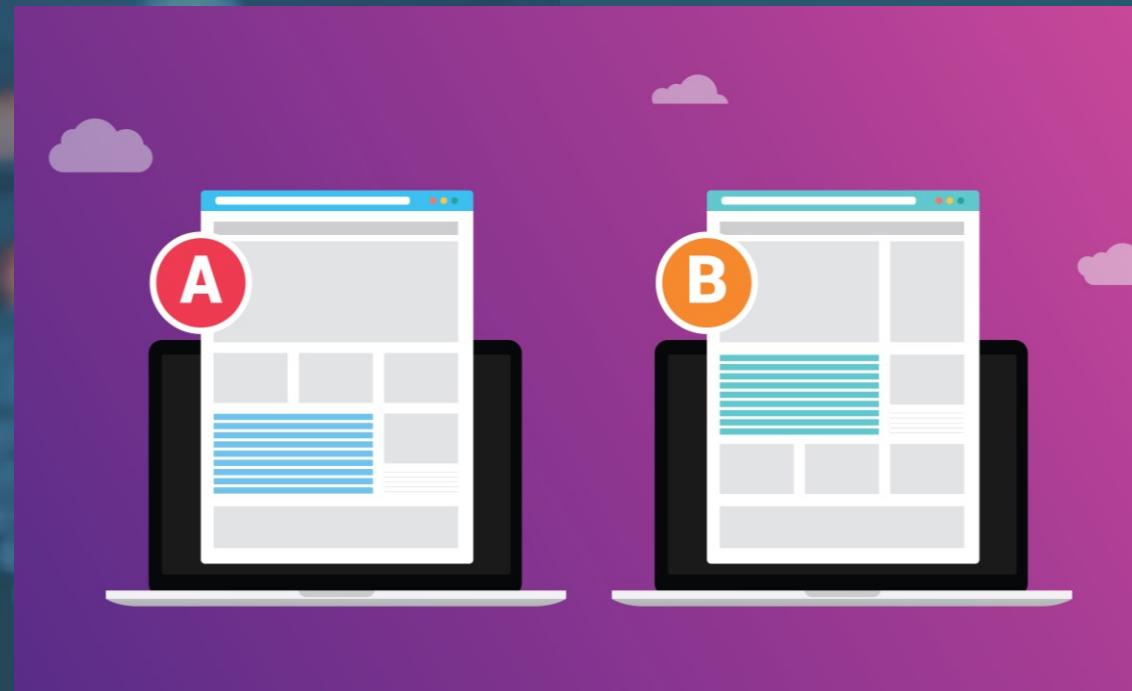


© Getty Images

ALURA. Criando um projeto com testes. Disponível em:
<https://cursos.alura.com.br/course/testes-net-teste-software/task/97789>.
Acesso em: 4 jan. 2024.

Exposição

Testes de regressão: importância e implementação



© Getty Images

O que são os testes de regressão?

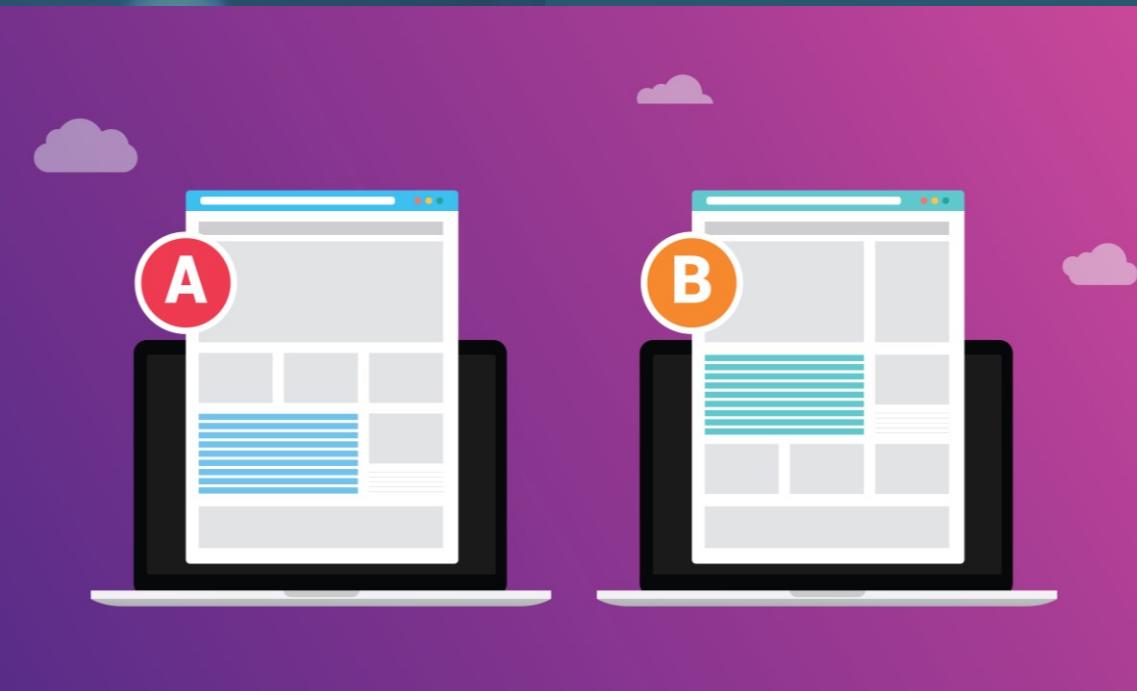
Os testes de regressão são um tipo de teste de software que verifica se as modificações recentes em um programa ou ambiente afetam adversamente as funcionalidades existentes.

Exemplo

Se o aplicativo de e-commerce lançar uma nova funcionalidade de recomendação de produtos, os testes de regressão garantirão que essa nova adição não quebre a funcionalidade de pesquisa de produtos.

Exposição

Testes de regressão: importância e implementação



© Getty Images

Por que é importante?

Capacidade de identificar problemas causados por mudanças, frequentemente sutis, que não foram previstas pelos desenvolvedores.

Como implementar?

A implementação eficaz dos testes de regressão pode ser feita automatizando os casos de teste críticos que precisam ser executados após cada mudança. Ferramentas, como GitLab CI/CD ou Jenkins, podem ser usadas sempre que o código é alterado e antes que ele seja fundido (ou "*merged*") na base de código principal.

Estratégia

A chave para uma estratégia de teste de regressão eficiente é ter um conjunto bem definido de casos de teste que cubra as funcionalidades centrais do sistema.



Vamos
fazer um
quiz

Qual das seguintes opções é um exemplo de teste funcional?

Verificar o tempo de resposta de uma aplicação sob carga pesada.

Avaliar a facilidade com que um usuário pode navegar pela interface do usuário.

Testar se a função de adicionar um item ao carrinho de compras está trabalhando como esperado.

Examinar a capacidade do software de se recuperar de falhas do sistema operacional.



Vamos
fazer um
quiz

Qual das seguintes opções é um exemplo de teste funcional?



Verificar o tempo de resposta de uma aplicação sob carga pesada.



Avaliar a facilidade com que um usuário pode navegar pela interface do usuário.



Testar se a função de adicionar um item ao carrinho de compras está trabalhando como esperado.



Examinar a capacidade do software de se recuperar de falhas do sistema operacional.

RESPOSTA ERRADA! Isso é um exemplo de um teste de desempenho, que é um tipo de teste não funcional. O foco aqui está no comportamento do sistema sob condições específicas de carga, não nas funções específicas do software.

RESPOSTA ERRADA! Isso descreve um teste de usabilidade, que é considerado um teste não funcional. Está mais relacionado à experiência do usuário do que à verificação de uma funcionalidade específica.

RESPOSTA CORRETA! Essa é uma tarefa clássica de um teste funcional, em que se verifica se uma função específica do aplicativo — neste caso, adicionar um item ao carrinho — está operando de acordo com os requisitos definidos.

RESPOSTA ERRADA! Isso é um teste de recuperação, que faz parte dos testes de confiabilidade, uma categoria de teste não funcional. Ele se preocupa com a capacidade do software de continuar operando após condições de falha ou erros.



Vamos
fazer um
quiz

Qual ferramenta é comumente usada para automatizar testes de interfaces de usuário em navegadores web?

JMeter

Selenium

JUnit

LoadRunner



Vamos
fazer um
quiz

Qual ferramenta é comumente usada para automatizar testes de interfaces de usuário em navegadores web?



JMeter

RESPOSTA ERRADA! O JMeter é mais comumente usado para testes de desempenho e carga, não para automação de interfaces de usuário de navegadores web.



Selenium

RESPOSTA CORRETA! Selenium é uma ferramenta de automação de testes amplamente utilizada para interfaces de usuário em navegadores web. Ela permite que scripts de teste interajam com elementos da web como se fosse um usuário real.



JUnit

RESPOSTA ERRADA! JUnit é uma framework que é usada principalmente para testes unitários em aplicações Java, não para automação de testes de UI em navegadores.



LoadRunner

RESPOSTA ERRADA! LoadRunner é uma ferramenta para testar desempenho e carga e não é projetada especificamente para automação de testes de interface do usuário.



Vamos
fazer um
quiz

Por que os testes de regressão são importantes no desenvolvimento de software?

Eles garantem que o software possa lidar com um grande número de usuários simultaneamente.

Eles avaliam a conformidade do software com os padrões de codificação internacionais.

Eles asseguram que os patches de segurança não introduzam novas vulnerabilidades.

Eles verificam se novas mudanças no código não afetaram negativamente as funcionalidades existentes.



Vamos
fazer um
quiz

Por que os testes de regressão são importantes no desenvolvimento de software?



Eles garantem que o software possa lidar com um grande número de usuários simultaneamente.



Eles asseguram que os patches de segurança não introduzam novas vulnerabilidades.



Eles avaliam a conformidade do software com os padrões de codificação internacionais.



Eles verificam se novas mudanças no código não afetaram negativamente as funcionalidades existentes.

RESPOSTA ERRADA! Isso descreve o objetivo dos testes de carga, não dos testes de regressão. Testes de carga são um tipo de teste não funcional.

RESPOSTA ERRADA! Embora isso possa ser parte de um processo de teste de regressão se a segurança for uma funcionalidade previamente estabelecida, a resposta não captura a essência geral dos testes de regressão, que é assegurar a integridade das funcionalidades existentes após qualquer mudança.

RESPOSTA ERRADA! A conformidade com padrões de codificação é, geralmente, avaliada durante revisões de código e testes de qualidade do código, não por meio de testes de regressão que focam na funcionalidade do software.

RESPOSTA CORRETA! Essa é a definição central dos testes de regressão. Eles são críticos após qualquer atualização ou melhoria do software para garantir que as funcionalidades existentes permaneçam inalteradas.



© Getty Images

O que nós
aprendemos
hoje?

Hoje desenvolvemos:

- 1** O conhecimento sobre a distinção entre os modelos de testes funcionais e não funcionais existentes na área de desenvolvimento;
- 2** A compreensão do funcionamento da configuração de teste em um projeto por meio de um exemplo real;
- 3** A identificação de características de funcionamento dos testes de regressão.



Saiba mais

```
void _decode_(char cbuff **buff
{
    if (step == AES LOC PASS) {
        src = cbuff->load();
        dest = getaddr(&step, mod,
        if ((mod != NULL) && dest)
            dest = mod->func(mod);
        if (dest != NULL)
            dest->func(dest);
    }
}
```

A área de TI é muito ampla e oferece muitas possibilidades de desenvolvimento dentro da carreira. Uma das funções muito procuradas é a de **analista de testes**. Que tal conhecer um pouco mais sobre esse papel?

CÓDIGO FONTE TV. O QUE FAZ UM ANALISTA DE TESTES?. Disponível em:

<https://www.youtube.com/watch?v=O7rB5XoakEc>.

Acesso em: 4 jan. 2024.

Referências da aula

ALURA. Criando um projeto com testes. Disponível em:

<https://cursos.alura.com.br/course/testes-net-teste-software/task/97789>.

Acesso em: 4 jan. 2024.

CÓDIGO FONTE TV. O QUE FAZ UM ANALISTA DE TESTES?. Disponível em:

<https://www.youtube.com/watch?v=O7rB5XoakEc>. Acesso em: 4 jan. 2024.

ROCHA, A. C. Principais tipos de teste de software. Medium, 2019. Disponível em:

https://medium.com/@annerocha_qa/principais-tipos-de-teste-de-software-4aeeb7fd23f1. Acesso em: 4 jan. 2024.

Identidade visual: imagens © Getty Images

Educação Profissional Paulista

**Técnico em
Desenvolvimento
de Sistemas**

Secretaria da
Educação



SÃO PAULO
GOVERNO DO ESTADO