

Aula 6

Teste de Software

Profª Maristela Weinfurter

1

Conversa Inicial

2

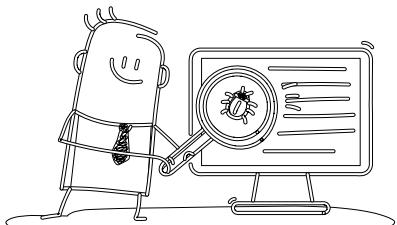


Maria Vorotna / shutterstock

3

- Técnicas importantes
 - Estruturais
 - ✓ As técnicas estruturais testam o código fonte, cada linha que for possível, fluxos básicos e alternativos do código
 - Funcionais
 - ✓ Testes de stress, execução, recuperação, operação, segurança

4



Maria Vorotna / shutterstock

5

Testes estáticos

6

Testes estáticos

- O teste estático e a análise de produtos de trabalho, tanto documentação quanto código, contribuem de forma mensurável para aumentar a qualidade do produto



AnnaKoles/shutterstock

7

Testes estáticos

- Técnica subestimada: o teste estático (ou análise estática) pode ser executado em um ambiente baseado em ferramentas ou manualmente, e é uma técnica de teste frequentemente negligenciada
- O teste estático pode assumir a forma de exame minucioso por uma ou mais pessoas ou pode ser realizado usando ferramentas de análise apropriadas

8

Testes estáticos

- Tudo é prevenção: o conceito subjacente do teste estático é a prevenção. Erros e outros desvios do plano são identificados antes que possam ter um efeito negativo durante o desenvolvimento posterior

9

Testes estáticos

- Os produtos de trabalho que podem ser verificados usando análise estática incluem:
 - Especificações
 - Requisitos de negócios
 - Requisitos funcionais
 - Requisitos não funcionais
 - Requisitos de segurança

10

Testes estáticos

- Os tipos de defeitos que a análise estática revela incluem:
 - Inconsistências
 - Ambiguidades
 - Contradições
 - Lacunas
 - Imprecisões
 - Redundâncias

11

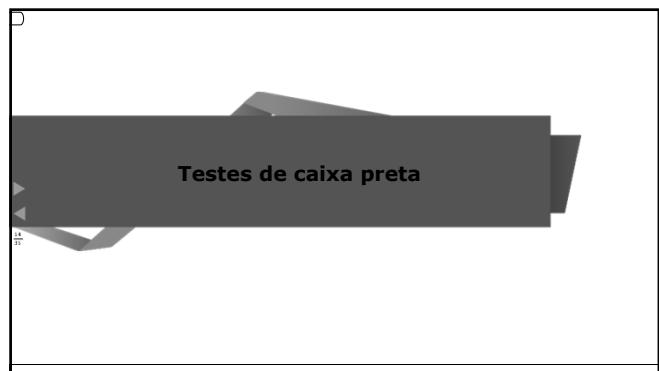
Testes estáticos

- Outras técnicas:
 - Revisão
 - Inspeção

12



13



14

Testes de caixa preta

- É impossível testarmos todas as entradas possíveis em nosso software, então, precisamos agrupá-las em categorias que demonstrem o comportamento de toda uma classe de entradas. Sem dúvida, já o fazemos em nossos planos de teste, e a isto chamamos de *particionamento de classe de equivalência*

15

Testes de caixa preta

- Dentro de uma partição, nem todos os valores são igualmente importantes
- Os erros são muito mais prováveis nos limites das partições onde os erros *off-by-one* podem estar presentes, por exemplo, se *maior-que-ou-igual* for usado em vez de *igual*
- A ideia é bastante simples: quando há um limite, precisamos testar os valores mais baixos e mais altos para garantir que a divisão esteja no lugar certo

16

Testes de caixa preta

Exemplo	Teste
<i>String</i> incluindo espaços	"hello world"
<i>String</i> incluindo caractere de acento	"Elan"
<i>String</i> incluindo letras maiúsculas	"Hello"
<i>String</i> com caracteres especiais	";!@£\$%íèà{{(")}}"

Weinfurter, 2023

17

Testes de caixa preta

- Por exemplo, para testarmos se uma caixa de texto pode lidar com entradas incluindo espaços, podemos usar a *string* "um dois"
- Essa *string* é um exemplo de todas as *strings* possíveis que incluem um espaço entre as palavras
- Eles são todos equivalentes, então, nós os particionamos juntos, e esse exemplo é nosso único teste para verificar todo o grupo

18

Testes de caixa preta

19

Testes de caixa preta

■ Tabelas de decisão

- Para capturarmos interações complexas entre variáveis dependentes e o comportamento do sistema, pode ser útil escrever as possibilidades em uma tabela
 - Isso fornece uma base para escrever casos de teste e garante que todas as condições sejam cobertas
 - Ao expandir as variáveis de maneira sistemática, confirmamos que não perdeu combinação alguma

20

Testes de caixa preta

- Consideraremos como exemplo um aplicativo da web com suporte básico ou avançado, dependendo do sistema operacional e do navegador da web no qual é executado
 - O modo avançado não substitui o modo básico; alguns usuários podem escolher o modo básico mesmo que o modo avançado esteja disponível

21

Testes de caixa preta

Tabela de Decisão

Sistema Operacional	Navegador	Modo do Aplicativo	Suportado?
Windows	Chrome	Básico	Sim
Windows	Chrome	Avançado	Sim
Windows	Edge	Básico	Sim
Windows	Edge	Avançado	Sim
Windows	Safari	Básico	Não
Windows	Safari	Avançado	Não
Mac	Chrome	Básico	Sim
Mac	Chrome	Avançado	Sim
Mac	Edge	Básico	Sim
Mac	Edge	Avançado	Não
Mac	Safari	Básico	Sim
Mac	Safari	Avançado	Não

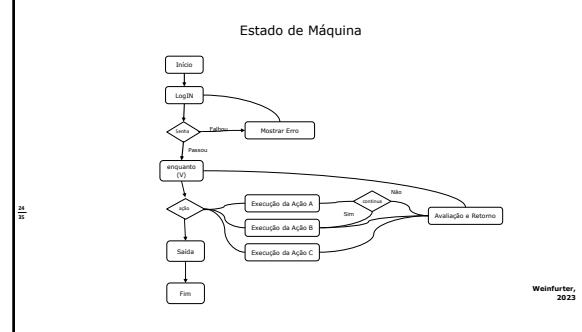
Weinfurter, 2023

22

Tabelas de Dados

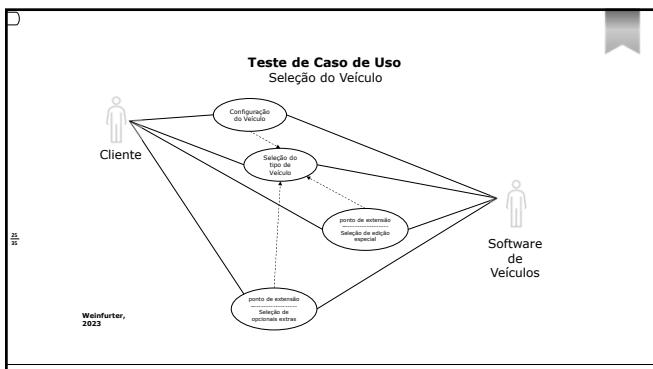
Sistema Operacional	Navegador	Modo do Aplicativo	Suportado?
Windows	Chrome	Básico	Sim
Windows	Chrome	Avançado	Sim
Sistema Operacional	Navegador	Modo do Aplicativo	Suportado?
Windows	Safari	Avançado	Não
Windows	Edge	Avançado	Sim
Mac	Edge	Avançado	Não
Mac	Safari	Básico	Sim
Mac	Safari	Avançado	Não

Avaliação



23

24



25



26

Teste de caixa branca

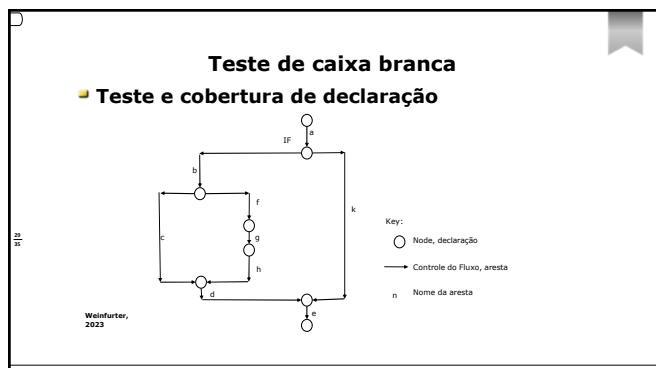
- As técnicas de teste de caixa branca são baseadas na estrutura interna do objeto de teste e geralmente são chamadas de testes estruturais ou baseados em estrutura. Outro termo para essa técnica é teste relacionado a código

27

Teste de caixa branca

- Podemos diferenciar entre os seguintes tipos de técnicas de teste de caixa branca:
 - Teste de declaração
 - Teste de decisão
 - Teste de condição
 - Teste de condição de ramificação
 - Teste de combinação de condição de ramificação
 - Teste de cobertura de decisão de condição modificada
 - Teste de caminho

28



29

Testes baseados na experiência e intuição

30

Testes baseados na experiência e intuição

- Cobertura apenas parcialmente ou nada mensurável
 - O teste baseado em experiência pode ser usado para descobrir defeitos que muitas vezes são ignorados pelo teste sistemático. Geralmente, é prudente aumentar os testes sistemáticos com testes baseados na experiência. Esses tipos de técnicas podem variar em sua eficácia na descoberta de defeitos, dependendo das áreas de especialização do testador

31

Testes baseados na experiência e intuição

- Derivação intuitiva de casos de teste (erro ao adivinhar)
 - O design de caso de teste intuitivo é baseado na capacidade intuitiva de usar erros, falhas e falhas esperadas para selecionar casos de teste adequados

32

Testes baseados na experiência e intuição

- Registre suas experiências em listas de verificação
 - Uma maneira metódica de apoiar essa técnica é manter listas de erros, falhas e falhas que você descobriu no decorrer de projetos anteriores, ou que você compilou com base em seu próprio conhecimento das razões pelas quais o software falha, e usá-las para ajudar a derivar novos casos de teste

33

Testes baseados na experiência e intuição

- Teste baseado em lista de verificação
 - Como o nome sugere, essa técnica usa listas de verificação como base para derivar casos de teste. Uma lista de verificação contém aspectos de um programa (ou seja, condições de teste) que precisam ser testados

34

Testes baseados na experiência e intuição

- Mantenha suas listas de verificação atualizadas
 - As listas de verificação são amplamente baseadas na experiência de um testador individual
- Cobertura
 - Podemos medir a cobertura calculando a porcentagem de entradas de lista que são cobertas por casos de teste. Se todos os aspectos da lista forem testados, alcançamos 100% de cobertura e é possível interromper o teste

35

36