

SOFT

DISCIPLINA: Engenharia de Software

AULA NÚMERO: 14

DATA: ____/____/____

PROFESSOR: Andrey

APRESENTAÇÃO

O objetivo desta aula é apresentar e discutir conceitos relacionados a Testes de Software.

DESENVOLVIMENTO

Introdução

Teste de software é o processo de executar programas com o objetivo de encontrar defeitos.

Qualidade de software é a satisfação dos requisitos funcionais, de desempenho e normas de desempenho explicitamente declaradas.

É uma atividade essencial para se garantir a qualidade do software.

É uma das últimas atividades que fará a revisão do produto.

Falhas em sistemas críticos:

- Therac 25
- Ariane 5 (erro em uma conversão de ponto flutuante de 64 bits para inteiro de 16 bits)
- Titan IV e Titan IV b
- Boeing 757 na Colômbia

Teste de software gasta 40% do esforço

Objetivos do teste: Revelar erros ainda não descobertos

Características de um bom teste:

- Um bom teste tem alta probabilidade de encontrar um erro
- Um bom teste não é redundante
- Um bom teste não deve ser muito simples nem muito complexo

Estratégias de Teste

Verificação e Validação (Definição IEEE)

VERIFICAÇÃO

Confirmar por testes e com provas objetivas que requisitos especificados foram cumpridos.

Garante que os produtos de uma dada fase implementam em sua totalidade as entradas para aquela fase, ou seja, **o produto foi construído corretamente.**

VALIDAÇÃO

Confirmar por testes e com provas objetivas que requisitos particulares para um determinado uso foram cumpridos.

Prova que o software implementa cada um dos requisitos corretamente e completamente ou seja, **o produto correto foi construído.**

Teste de unidade

É o teste realizado na menor unidade de projeto de software que é o componente ou módulo de software. O teste de unidade tem como objetivo testar a lógica interna de processamento e as estruturas de dados no escopo do componente. São testados:

Interface, estruturas lógicas de dados, condições limites, caminhos independentes e caminhos de manipulação de erros.

Para realizar o teste de unidade é necessário a implementação de um pseudocontrolador (driver) e um pseudocontrolado (stub). O pseudocontrolador simula os módulos do sistema que chamam o módulo a ser testado. Os pseudocontrolados simulam os módulos do sistema que são chamados pelo módulo a ser testado.

Uma prática eficiente, vinda do desenvolvimento ágil é projetar o teste antes de iniciar a implementação do módulo.

Testes de integração

É o teste realizado para verificar se os módulos desenvolvidos vão funcionar corretamente quando colocados em conjunto. O foco principal do teste são as interfaces dos módulos.

Pode ser usada uma abordagem incremental ou Big-Bang. Na abordagem incremental o teste de integração é feito à medida que os módulos vão sendo desenvolvidos, enquanto que na Big-Bang ele é feito todo de uma vez ao final do desenvolvimento.

A Integração dos módulos pode ser feita de maneira descendente ou ascendente.

Teste de regressão. Reexecução de algum subconjunto de testes para garantir que a adição de um novo módulo não propaguem efeitos colaterais indesejáveis.

Teste fumaça. Teste realizado de maneira rápida e freqüente de forma a testar o sistema de ponta a ponta e expor os problemas principais do sistema durante o desenvolvimento.

Teste de Validação

Teste realizado quando os componentes individuais já foram integrados. Focaliza ações visíveis para o usuário e saídas do sistema reconhecidas pelo usuário. A validação do sistema irá mostrar o quanto o sistema construído está em conformidade com os requisitos de software especificados.

Revisão de configuração. Tem como objetivo garantir que todos os componentes da configuração do software estão devidamente implementados e documentados.

Testes alfa e beta. Os testes alfa são realizados na instalação do desenvolvedor, com os usuários finais e desenvolvedores, em ambiente controlado. Os testes beta são realizados nas instalações dos usuários finais, num ambiente que não pode ser controlado e sem a presença dos desenvolvedores.

Teste de Sistema

É um conjunto de diferentes testes cuja finalidade é principal é exercitar por completo o sistema baseado em computador. Estes testes contribuem para verificar se os elementos do sistema estão adequadamente integrados e executam as funções a eles alocadas.

Tipos de testes de sistema:

- de recuperação,
- de segurança,
- de estresse e
- de desempenho

Técnicas de teste

- Funcional (caixa preta)
- Estrutural (caixa branca)

- Baseada em erros

Teste Estrutural ou Caixa Branca

É um método de projeto de testes que usa a estrutura de controle do projeto procedimental para derivar casos de teste (Pressman, 2006)

Baseia-se num minucioso exame dos detalhes procedimentais

Caminhos lógicos do software são testados

Não é viável testar todos os caminhos lógicos de um programa (teste exaustivo)

Teste Exaustivo

Programa Pascal com 100 linhas e dois ciclos aninhados que executam entre 1 e 20 vezes cada um dependendo do dado da entrada.

Dentro do ciclo interior 4 construções se-então-senão.

10^{14} caminhos possíveis de execução

Se cada caso de teste for executado por um processador "mágico" de testes em 1 mseg

3170 anos para completar os testes.

Teste de Caminho Básico

Técnica de teste de caixa branca

McCabe (1976)

Conjunto Básico de caminhos de execução

Os casos de teste derivados para executar os caminhos básicos tem a garantia de executar cada instrução pelo menos uma vez

Casos de Teste

Um caso de teste é composto de um dado de entrada (dado de teste) e de uma saída esperada

Um bom caso de teste é aquele que tem alta probabilidade de revelar um defeito ainda não descoberto

Os casos de teste no teste estrutural devem:

Garantir que todos os caminhos independentes de um módulo tenham sido exercitados pelo menos uma vez

Exercitem todas as decisões lógicas em seus lados verdadeiro e falso

Executem todos os ciclos nos seus limites e dentro de seus intervalos operacionais

Exercitem as estruturas de dados internas

Notação Grafo de Fluxo

O grafo de fluxo mostra o fluxo de controle

Nós representam um ou mais processos

Arestas representam o fluxo de controle

Regiões do grafo são áreas limitadas pelas arestas e nós (incluindo a área fora do grafo)

