

# Glossário Standard de termos usados em Testes de Software

**Versão 2.2pt (19 de outubro de 2012)**

**Produzido pelo “Glossary Working Party”  
International Software Testing Qualifications Board**

Editor :

Erik van Veenendaal (Países Baixos)  
Ana Paiva (versão Portuguesa)

Direitos de Autor

Este documento pode ser copiado na sua totalidade ou parte se a fonte for referenciada.

## **Contributos para a versão Portuguesa**

Alexandra Médicis  
Ana Paiva  
António Rodrigues  
António Sá Pinto  
Daniel Gomes  
Esteves Costa  
Filipe Nuno Carlos  
Joel Oliveira  
José Tavares  
Luis Amaral  
Mariana Caiado  
Nuno Baptista  
Patrícia Alves  
Paulo Magalhães  
Paulo Sargaço  
Pedro Pereira  
Ricardo Cordeiro

## **Contributos para o documento original**

Rex Black (USA)	Ana Paiva (Portugal)
Josie Crawford (Australia)	Dale Perry (USA)
Enst Düring (Norway)	Ander Pettersson (Sweden)
Sigrid Eldh (Sweden)	Horst Pohlmann (Germany)
Isabel Evans (UK)	Juha Pomppu (Finland)
Simon Frankish (UK)	Meile Posthuma (The Netherlands)
David Fuller (Australia)	Erkki Pöyhönen (Finland)
Annu George (India)	Maaret Pyhäjärvi (Finland)
Dorothy Graham (UK)	Andy Redwood (UK)
Mats Grindal (Sweden)	Stuart Reid (UK)
Matthias Hamburg (Germany)	Piet de Roo (The Netherlands)
Julian Harty (UK)	Steve Sampson (UK)
David Hayman (UK)	Shane Saunders (UK)
Bernard Homes (France)	Hans Schaefer (Norway)
Ian Howles (UK)	Jurriën Seubers (The Netherlands)
Juha Itkonen (Finland)	Dave Sherratt (UK)
Paul Jorgensen (USA)	Mike Smith (UK)
Vipul Kocher (India)	Andreas Spillner (Germany)
Gerard Kruijff (The Netherlands)	Lucjan Stapp (Poland)
Fernando Lamas de Oliveira (Portugal)	Richard Taylor (UK)
Tilo Linz (Germany)	Geoff Thompson (UK)
Gustavo Marquez Sosa (Spain)	Stephanie Ulrich (Germany)
Judy McKay (US)	Matti Vuori (Finland)
Don Mills (UK)	Gearrel Welvaart (The Netherlands)
Peter Morgan (UK)	Paul Weymouth (UK)
Thomas Müller (Switzerland)	Pete Williams (UK)
Avi Ofer (Israel)	

## Histórico de Alterações ao Documento Original

### Versão 2.2 d.d. 19-10-2012

Esta nova versão foi desenvolvida para suportar o programa de nível avançado (versão de 2012) e o programa de nível especialista - *Test Management* (versão de 2011). Além disso, houve também alterações para implementar pedidos de alteração para a versão 2.2 do Glossário do ISTQB.

#### Novos termos adicionados:

- análise de domínio (domain analysis)
- anti-padrão (anti-pattern)
- arquiteto de teste (test architect)
- ataque com homem no meio (man in the middle attack)
- ator (actor)
- avaliação de risco (risk assessment)
- categoria de defeitos (defect category)
- CMMI
- comité de gestão de defeitos (defect management committee)
- comité de gestão de defeitos (defect triage committee)
- condição atômica (atomic condition)
- control de qualidade (quality control)
- curto-circuito (short-circuiting)
- definição de perfil operacional (operational profiling)
- desenvolvimento guiado por características (feature-driven development)
- director/gestor de testes (test director)
- eficácia (effectiveness)
- estimativa de três pontos (three point estimation)
- fase de contenção (phase containment)
- ferramenta open source (open source tool)
- ferramenta personalizada (custom tool)
- gráfico de control (control chart)
- gráfico Shewhart (Shewhart chart)
- impacto do risco (risk impact)
- injeção de falhas (fault injection)
- intervalo de confiança (confidence interval)
- matriz RACI (RACI matrix)
- metodologia por objetivos S.M.A.R.T. (S.M.A.R.T. goal)

#### Termos alterados:

- análise estática (static analysis)
- avaliação heurística (heuristic evaluation)
- calendário de execução de testes (test execution schedule)
- caminho-dd (dd-path)
- Capability Maturity Model (CMM)
- Capability Maturity Model Integration (CMMI)
- cobertura de determinação de condições (condition determination coverage)
- cobertura de múltiplas condições (multiple condition coverage)
- cobertura modificada de decisões e condições (modified condition decision coverage)
- cobertura modificada de múltiplas condições (modified multiple condition coverage)
- complexidade ciclomática (cyclomatic complexity)
- condição (condition)
- eficiência (efficiency)
- entrega diária (daily build)
- especificação da conceção de testes (test design specification)
- especificação de caso de teste (test case specification)
- especificação de procedimento de teste (test procedure specification)
- extreme programming
- fator crítico de sucesso (critical success factor)
- ferramenta de teste de carga (load testing tool)
- Gestão de Qualidade Total (Total Quality Management)
- injeção de falhas (fault injection)
- líder de avaliação (lead assessor)
- mapa mental (mind map)

<p>methodology)</p> <ul style="list-style-type: none"> <li>- métrica de convergência (convergence metric)</li> <li>- missão de teste (test mission)</li> <li>- modelo embebido de desenvolvimento iterativo (embedded iterative development model)</li> <li>- Myers-Briggs Type Indicator (MBTI)</li> <li>- nível de integridade do software (software integrity level)</li> <li>- planeamento poker (planning poker)</li> <li>- plano de revisão (review plan)</li> <li>- predicado (predicate)</li> <li>- probabilidade do risco (risk likelihood)</li> <li>- qualidade de dados (data quality)</li> <li>- risco de qualidade (quality risk)</li> <li>- SUMI</li> <li>- teste analítico (analytical testing)</li> <li>- teste baseado em ataques (attack-based testing)</li> <li>- teste baseado em modelos (model-based testing)</li> <li>- teste baseado na experiência (experience-based testing)</li> <li>- teste combinatório (combinatorial testing)</li> <li>- teste de conformidade com processo (process-compliant testing)</li> <li>- teste consultivo (consultative testing)</li> <li>- teste de aceitação de fábrica (factory acceptance testing)</li> <li>- teste de API (API – Application Programming Interface – testing)</li> <li>- teste de conformidade com standards (standard-compliant testing)</li> <li>- teste de fluxo de control (control flow testing)</li> <li>- teste de integração aos pares (pairwise integration testing)</li> <li>- teste de integração hardware-software (hardware-software integration testing)</li> <li>- teste de integração por vizinhos (neighborhood integration testing)</li> <li>- teste metódico (methodical testing)</li> <li>- teste por terceiros (outsourced testing)</li> <li>- teste reativo (reactive testing)</li> </ul>	<ul style="list-style-type: none"> <li>- maturidade (maturity)</li> <li>- método de classificação em árvore (classification tree method)</li> <li>- origem do defeito (root cause)</li> <li>- par definição-uso (definition-use pair)</li> <li>- perfil de desempenho (performance profiling)</li> <li>- processos de teste críticos (Critical Testing Processes)</li> <li>- qualidade de software (software quality)</li> <li>- requisitos testáveis (testable requirements)</li> <li>- revisão técnica (technical review)</li> <li>- semear falhas (fault seeding)</li> <li>- SPI</li> <li>- técnica baseada em defeitos (defect-based technique)</li> <li>- técnica baseada na estrutura (structure-based technique)</li> <li>- técnica de conceção de testes baseada em defeitos (defect-based test design technique)</li> <li>- Test Improvement Model (TPI)</li> <li>- Test Maturity Model (TMM)</li> <li>- Test Maturity Model integration</li> <li>- teste ágil (agile testing)</li> <li>- teste big-bang (big-bang testing)</li> <li>- teste de comparação elementar (elementary comparison testing)</li> <li>- teste de determinação de condições (condition determination testing)</li> <li>- teste de stress (stress testing)</li> <li>- teste de thread (thread testing)</li> <li>- teste do fluxo de dados (data flow testing)</li> <li>- teste estático (static testing)</li> <li>- teste modificado de decisões e condições (modified condition decision testing)</li> <li>- teste modificado de múltiplas condições (modified multiple condition testing)</li> <li>- teste orientado a dados (data-driven testing)</li> <li>- teste orientado a palavras ação (action word driven testing)</li> <li>- teste orientado a palavras-chave (keyword-driven testing)</li> <li>- walkthrough</li> </ul>
---	--

<ul style="list-style-type: none"><li>- teste sobre falhas (failover testing)</li><li>- testes adversos à regressão (regression-averse testing)</li><li>- testes na fonte (insourced testing)</li><li>- tipo de defeito (defect type)</li><li>- TMMi</li><li>- TPI-Next</li><li>- user story</li><li>- user story testing</li><li>- WAMMI</li><li>- Website Analysis and MeasureMent Inventory (WAMMI)</li></ul>	
--	--

# Índice

Prefácio .....	7
<b>1. Introdução .....</b>	<b>7</b>
<b>2. Âmbito.....</b>	<b>7</b>
<b>3. Estrutura.....</b>	<b>8</b>
<b>4. Referências Normativas.....</b>	<b>8</b>
<b>5. Definições .....</b>	<b>9</b>
<b>A.....</b>	<b>9</b>
<b>B.....</b>	<b>13</b>
<b>C.....</b>	<b>14</b>
<b>D.....</b>	<b>20</b>
<b>E.....</b>	<b>22</b>
<b>F.....</b>	<b>24</b>
<b>G.....</b>	<b>27</b>
<b>H.....</b>	<b>29</b>
<b>I.....</b>	<b>29</b>
<b>L.....</b>	<b>31</b>
<b>M.....</b>	<b>32</b>
<b>N.....</b>	<b>35</b>
<b>O.....</b>	<b>36</b>
<b>P.....</b>	<b>37</b>
<b>Q.....</b>	<b>39</b>
<b>R.....</b>	<b>40</b>
<b>S.....</b>	<b>44</b>
<b>T.....</b>	<b>46</b>
<b>U.....</b>	<b>59</b>
<b>V.....</b>	<b>59</b>
<b>W.....</b>	<b>59</b>
<b>Anexo A (Referências) .....</b>	<b>61</b>
<b>Anexo B (Método para comentar este glossário).....</b>	<b>63</b>
<b>Anexo C (Tradução dos termos em inglês) .....</b>	<b>64</b>

## Prefácio

Na elaboração deste glossário, a equipa de trabalho tem procurado compilar um conjunto diversificado de pontos de vista e comentários, quer de pessoas da indústria e do comércio, como também de entidades e órgãos governamentais, com o objetivo de produzir uma norma internacional (padrão) de testes que encontrasse aceitação no maior número de áreas possível. O acordo unânime de todos na elaboração de um documento desta natureza é difícil ou muitas vezes irrealizável. As contribuições para este glossário foram recebidas das comunidades de testes na Austrália, Bélgica, Finlândia, França, Alemanha, Índia, Israel, Países Baixos, Noruega, Portugal, Espanha, Suécia, Suíça, Reino Unido e EUA.

Muitos testadores de software têm utilizado a norma BS 7925-1, o standard Britânico *Glossary of Software Testing Terms*, desde a sua publicação original em 1998. Inicialmente, a norma foi desenvolvida a pensar em testes de componentes mas, desde a sua publicação, têm sido vários os comentários e propostas para introdução de novas definições de forma a melhorar e alargar o âmbito da norma para abranger outros tipos de teste de software. O glossário de testes do ISTQB (International Software Testing Qualification Board) incorporou muitas dessas atualizações sugeridas e o documento foi usado como referência para elaborar o regulamento de qualificação em testes de software.

O Glossário do ISTQB tem dois objetivos principais:

- Suportar a compreensão do Programa de Cursos do ISTQB definindo os termos usados nesses diferentes programas
- Suportar a comunicação entre a comunidade internacional de testes e os seus *stakeholders*, fornecendo um vocabulário standard de teste.

Os Conselhos nacionais ou regionais do ISTQB podem usar o Glossário do ISTQB, para traduzir para a sua língua local. Estes Conselhos podem adaptar o ISTQB Glossário às suas necessidades linguísticas específicas.

## 1. Introdução

Atualmente, é desperdiçado muito tempo e esforço na indústria, comércio, governo e instituições académicas e profissionais, quando surgem ambiguidades resultantes da incapacidade de diferenciar adequadamente termos como “cobertura de instruções” (*statement coverage*) e “cobertura de decisões” (*decision coverage*); “bateria de testes” (*test suite*), “especificação do teste” (*test specification*) e “plano de testes” (*test plan*) e termos similares que suportam a comunicação entre os diversos setores da sociedade. Além disso, o uso profissional ou técnico desses termos é, frequentemente, não consensual quanto ao significado que lhes é atribuído.

## 2. Âmbito

Este documento apresenta conceitos, termos e definições com o objetivo de ajudar a comunicação na área de testes de software e disciplinas relacionadas.

### 3. Estrutura

#### Organização

O glossário contém uma única secção de definições organizada por ordem alfabética. Alguns termos são preferidos relativamente aos seus sinónimos. Nestes casos, apresenta-se a definição do termo preferido e os seus sinónimos remetem para essa mesma definição. Por exemplo, “testes estruturais” remete para “teste caixa-branca”. Para sinónimos é utilizado o termo “Ver”, e para referências cruzadas é utilizado o termo “Ver também”. Estes termos ajudam o utilizador a navegar rapidamente para o item relativo ao termo que aparece à direita. As referências cruzadas “Ver também” são igualmente usadas para relações entre um termo mais amplo e um termo mais específico, bem como para a sobreposição de significado entre dois termos.

#### Palavras Chave

O Glossário do ISTQB contém muitos termos por diferentes razões. Alguns deles são fornecidos para apoiar o leitor de um programa ISTQB na compreensão do texto. Alguns são apresentados “apenas” porque o termo foi usado numa versão anterior de um programa e existe a preocupação de aplicar as regras da compatibilidade de versões. No entanto, provavelmente os termos mais importantes são as palavras-chave (usadas em exame exame) e que são explicitamente identificadas pelos diversos programas ISTQB. Um importante grupo de utilizadores destas palavras-chave são os profissionais (de teste) que se estão a preparar para um exame ISTQB. Para apoiá-los, as palavras-chave que precisam de conhecer para um exame particular são indicadas neste glossário. Note-se que o princípio da herança é aplicável. Por exemplo, para um exame ISTQB avançado é preciso conhecer todas as palavras-chave do nível base do ISTQB. As palavras-chave são indicadas da seguinte maneira :

F : Palavra-chave do ISTQB nível base (*Foundation Syllabus*)

ATM: Palavra-chave do ISTQB nível avançado – *Test Management Syllabus*

ATA : Palavra-chave do ISTQB nível avançado – *Test Analyst Syllabus*

ATT : Palavra-chave do ISTQB nível avançado – *Technical Test Analyst Syllabus*

EITP : Palavra-chave do ISTQB nível especialista – *Improving the Testing Process Syllabus*

ETM : Palavra-chave do ISTQB nível especialista – *Test Management Syllabus*.

Note-se que, se uma palavra-chave é identificada num programa (*Syllabus*), mas se não é o termo preferido de acordo com o glossário, tanto a palavra-chave como o termo a que se refere (indicado em "Ver" ) são marcados com o indicador do programa adequado.

#### Referências

As referências deste glossário são usadas de duas formas.

- Parentesis retos sem a adição de "depois", por exemplo, [ISO 9126], significa que é usado o texto exato da referência.
- No caso da definição de uma referência ter sido adaptada com pequenas alterações para o contexto do glossário do ISTQB, a adição "depois de" é utilizada, por exemplo, [Depois de ISO 9126].

### 4. Referências Normativas

Neste documento, são usadas as seguintes referências:

- CMMI e IDEAL são marcas registadas de Carnegie Mellon University
- EFQM é uma marca registada da Fundação EFQM

- Rational Unified Process é uma marca registada da Rational Software Corporation
- STEP é uma marca registada da Software Quality Engineering
- TMap, TPA e TPI-Next são marcas registadas da Sogeti Nederland BV
- TMMi é uma marca registada da Fundação TMMi

## 5. Definições

### A

- F **abordagem de teste:** (test approach) Implementação da estratégia de teste para um projeto específico. Tipicamente inclui as decisões efetuadas com base no objetivo do projeto (de teste) e na avaliação de risco realizada, pontos de partida do processo de teste, técnicas de conceção de teste a ser aplicadas, critérios de saída e tipos de testes a ser realizados.
- ATM
- EITP **ação (IDEAL):** (acting (IDEAL)) Fase dentro do modelo IDEAL onde as melhorias são desenvolvidas, colocadas em prática e implementadas em toda a organização. A fase de ação consiste nas seguintes atividades: criação de uma solução, a solução de teste/piloto, o aperfeiçoamento da solução e a implementação da solução. Ver também *IDEAL*.
- aceitação:** (acceptance) Ver *teste de aceitação*.
- ATT **adaptabilidade:** (adaptability) Capacidade que o produto de software possui de se adaptar a diferentes ambientes sem ser necessário aplicar ações ou mecanismos adicionais para além daqueles fornecidos pelo software para esse efeito. [ISO 9126] Ver também *portabilidade*.
- adequabilidade:** (suitability) Capacidade do produto de software de fornecer um conjunto de funções apropriadas para tarefas especificadas e objetivos do utilizador. [ISO 9126] Ver também *funcionalidade*.
- alvo de teste:** (test target) Conjunto de critérios de saída.
- F **ambiente de teste:** (test environment; test bed) Ambiente que contém o hardware, instrumentação, simuladores, ferramentas de software e outros elementos de suporte necessários para realizar um teste. [Depois de IEEE 610]
- ambiente operacional:** (operational environment) Produtos de software ou hardware instalados nos locais de trabalho ou nas instalações dos utilizadores ou consumidores, onde o componente ou sistema sob teste será utilizado. O ambiente pode incluir, como software, sistemas operativos, sistemas de gestão de base de dados e outras aplicações.
- ATT **analísabilidade:** (analyzability) Capacidade que o produto de software tem para ser analisado relativamente a deficiências ou causas de falhas, ou a capacidade em se identificarem componentes que foram modificados. [ISO 9126] Ver também *manutenibilidade*.
- analisador:** (analyzer) Ver *analisador estático*.
- ATT **analisador estático:** (static analyzer) Ferramenta que efetua análises estáticas.
- analisador estático de código:** (static code analyzer) Ferramenta que leva a cabo análises estatísticas de código. A ferramenta procura certas propriedades no código fonte, tais como: conformidade com normas de codificação, métricas de qualidade ou anomalias no fluxo de dados.
- EITP **análise causal:** (causal analysis) Análise de defeitos para determinar as suas causas. [CMMI]

**análise causa-efeito:** (cause-effect analysis) Ver *teste causa-efeito*.

ATA **análise da origem do defeito:** (root cause analysis) Técnica de análise direcionada a identificar a origem dos defeitos. Ao aplicar medidas corretivas à causa de origem, espera-se que a probabilidade de recorrência do defeito seja minimizada.

EITP **Análise de Árvore de Falhas:** (Fault Tree Analysis (FTA)) Técnica utilizada para analisar as causas de defeitos. Esta técnica modela visualmente as relações lógicas entre falhas, erros humanos e eventos externos, e avalia como se podem relacionar na causa de defeitos específicos.

[N. do T.] Embora a tradução mais correta seja análise de árvore de defeitos, optou-se por manter o termo mais usual.

**Análise de Árvore de Falhas de Software:** (Software Fault Tree Analysis (SFTA)) Ver *análise de árvore de falhas (FTA)*.

**análise de cobertura:** (coverage analysis) Medida da cobertura de um critério específico (item de cobertura), obtida durante a execução dos testes, para determinar se são necessários testes adicionais e, em caso afirmativo, quais os casos de teste necessários.

**analisador de código:** (code analyzer) Ver *analisador estático de código*.

ATA **análise de domínio:** (domain analysis) Técnica de concepção de casos de teste caixa preta que é usada para identificar casos de teste eficientes e eficazes quando várias variáveis podem ou devem ser testadas em conjunto. Não só se baseia e generaliza o conceito de partição em classes de equivalência e de análise de valores fronteira. Ver também *análise de valor fronteira, partição de equivalência*.

ATT **análise de fluxo de controle:** (control flow analysis) Forma de análise estática com base numa representação de caminhos únicos (sequências de eventos) num componente ou sistema. A análise de fluxo de controle avalia a integridade das estruturas, olhando para possíveis anomalias nesse fluxo, tais como, circuitos fechados ou etapas inacessíveis do ponto de vista lógico.

F **análise de impacto:** (impact analysis) Avaliação de alterações necessárias com o objetivo de implementar uma dada alteração na especificação de requisitos. A avaliação dessas alterações passa pela análise da documentação de desenvolvimento, da documentação de testes e dos próprios documentos de software.

**análise de mutação:** (mutation analysis) Método que determina a capacidade da bateria de testes em discernir as diferenças entre o programa original e as suas pequenas variantes (mutantes).

EITP **análise de Pareto:** (Pareto analysis) Técnica estatística para decidir sobre a seleção de um número limitado de fatores que produzem um resultado significativo. Em termos da melhoria da qualidade, a grande maioria dos problemas (80%) são produzidos por um número pequeno de causas chave (20%).

ATM **análise de risco:** (risk analysis; hazard analysis) Processo de avaliação dos riscos identificados para estimar o seu impacto e probabilidade de ocorrência.

ATA

ATT

F

ATA

**análise de valor fronteira:** (boundary value analysis) Técnica caixa-preta de concepção de testes em que os casos de teste são concebidos com base em valores fronteira. Ver também *valor fronteira*.

ATT **análise dinâmica:** (dynamic analysis) Processo de avaliar o comportamento. Por exemplo, o desempenho da memória ou o consumo do CPU de um componente ou sistema durante a

execução. [IEEE 610]

- ATT **análise do fluxo de dados:** (data flow analysis) Forma de análise estática baseada na definição e uso de variáveis.
- análise do perfil de desempenho:** (performance profiling) Tarefa de análise, como por exemplo, a identificação de estrangulamentos de desempenho baseado nas métricas geradas, e respetivo ajuste do desempenho de um componente de software ou sistema utilizando ferramentas.
- F **análise estática:** (static analysis) Análise de artefactos de desenvolvimento de software, por  
ATT exemplo, requisitos ou código, realizadas sem a execução desses artefactos. A análise estática é geralmente realizada por meio de uma ferramenta de suporte.
- análise estática de código:** (static code analysis) Análise do código fonte levada a cabo sem a execução desse software.
- análise por Pontos de Função:** (Function Point Analysis (FPA)) Método que visa medir o tamanho da funcionalidade de um sistema de informação. A medição é independente da tecnologia. A mesma pode ser utilizada como base para a medição de produtividade, para a estimativa dos recursos necessários e/ou para o controlo de um projeto.
- EITP **análise transaccional:** (transactional analysis) Análise das transações entre pessoas e das transações dentro da mente de uma pessoa; uma transação é definida como um estímulo e uma resposta. As transações ocorrem entre pessoas e entre estados de ego (segmentos de personalidade) dentro da mente de uma pessoa.
- ATM **anomalia:** (anomaly) Qualquer condição que se desvia da expectativa criada pelas especificações de requisitos, documentos de conceção, documentos de utilizadores, normas, etc., ou a partir da perceção ou experiência de alguém. As anomalias podem ser encontradas, entre outras, durante procedimentos de revisão, teste, análise ou compilação, ou durante a utilização de produtos de software ou respetiva documentação. [IEEE 1044] Ver também *defeito, desvio, erro, falha, incidente, problema*.
- anotador:** (recorder) Ver *redator*.
- F **antecipar erros:** (error guessing) Técnica de conceção de testes onde a experiência do  
ATA testador (tester) antecipa quais os defeitos que possam estar presentes no componente ou sistema a testar, e, por conseguinte, cria casos de teste especificamente para evidenciar esses defeitos.
- ATT **anti-padrão:** (anti-pattern) Ação repetida, processo, estrutura ou solução reutilizável que inicialmente parece ser benéfico e é normalmente utilizado, mas é ineficaz e/ou contraproducente na prática.
- apontador:** (pointer) Dado que indica a localização de um outro dado, por exemplo, uma variável que indica o endereço de um registo para ser processado. [IEEE 610]
- ATT **apontador partido:** (wild pointer) Apontador que referencia uma localização que está fora do âmbito desse apontador ou que não existe. Ver também *apontador*.
- ATA **aprendibilidade:** (learnability) Capacidade que um produto de software tem de possibilitar ao utilizador aprender a sua aplicação. [ISO 9126]. Ver também *usabilidade*.
- EITP **aprendizagem (IDEAL):** (learning (IDEAL)) Fase dentro do modelo IDEAL onde uma pessoa aprende pela sua experiência melhorando a sua capacidade de adotar novos processos e tecnologias. A fase de aprendizagem consiste nas atividades de análise e validação, e na proposta de ações futuras. Ver também *IDEAL*.

**apresentação estruturada de documento / travessia estruturada:** (structured walkthrough) Ver *apresentação de documento*.

**aprovação:** (pass) Um teste será aprovado se o resultado obtido for igual ao resultado esperado.

**aprovação de teste:** (test pass) Ver *aprovação*.

**armazenamento:** (storage) Ver *utilização de recursos*.

ETM **arquiteto de teste:** (test architect) (1) Uma pessoa que fornece orientações e direções estratégicas para uma organização de teste e para a sua relação com outras disciplinas. (2) Uma pessoa que define a forma como o teste é estruturado para um determinado sistema, incluindo tópicos como ferramentas de teste e gestão de dados de teste.

**árvore de classificação:** (classification tree) Árvore mostrando as partições de equivalência hierarquicamente ordenadas, que é utilizada para efetuar a conceção de testes segundo o método de classificação em árvore. Ver também *método de classificação em árvore*.

**assistente de instalação:** (installation wizard) Software fornecido por qualquer meio para guiar o processo de instalação. Normalmente executa o processo de instalação, fornece *feedback* sobre os resultados da instalação e apresenta opções de personalização.

F **ataque:** (attack; fault attack) Tentativa focada e dirigida para avaliar a qualidade, em especial a fiabilidade, de um objeto de teste, tentando forçar a ocorrência de falhas específicas. Ver também *teste negativo*.

**ataque por interposta pessoa:** (man in the middle attack) Interceção de uma comunicação, através de imitação e/ou alteração, com a subsequente retransmissão por um terceiro (por exemplo, transações com cartão de crédito) de forma a que o utilizador não se aperceba desse terceiro.

**ataque de software:** (software attack) Ver *ataque*.

**ator:** (actor) Utilizador ou qualquer outra pessoa ou sistema que interage com o sistema alvo de teste de uma forma específica.

ATA **atratividade:** (attractiveness) Capacidade do produto de software em ser atraente para o utilizador final. [ISO 9126] Ver também *usabilidade*.

**atributo de qualidade:** (quality attribute) Característica com impacto direto na qualidade de um item. [IEEE 610]

ATM **auditoria:** (audit) Avaliação independente de produtos de software ou processos para verificar a conformidade, com base em critérios objetivos, com normas, diretrizes, especificações e/ou procedimentos, incluindo documentos que especificam:  
(1) a forma ou o conteúdo dos produtos a serem produzidos  
(2) o processo pelo qual os produtos devem ser produzidos  
(3) a forma como deve ser avaliado o cumprimento de normas ou diretrizes. [IEEE 1028]

**auditoria de configuração:** (configuration auditing) A função de verificar o conteúdo de bibliotecas de itens de configuração, por exemplo, para o cumprimento de normas. [IEEE 610]

**automação de execução de teste:** (test execution automation) Utilização de software, por ex., ferramentas de captura/reprodução, para controlar a execução de testes, efetuar a comparação entre o resultado obtido e o esperado, preparar as pré-condições de testes ou outras atividades de controlo de testes ou para disponibilizar gráficos/relatórios.

**automação de testes:** (test automation) Utilização de software para executar ou apoiar as

atividades de teste. Por exemplo, na gestão de testes, na conceção de testes, na execução de testes e na verificação de resultados.

**avaliação:** (evaluation) Ver *testar*.

EITP **avaliação de processo:** (process assessment) Avaliação disciplinada dos processos de software de uma organização face a um modelo de referência. [Depois de ISO 15504]

ATM **avaliação de risco:** (risk assessment) Processo de avaliação de um determinado projeto ou  
ATT produto para determinar o seu nível de risco, geralmente atribuindo uma classificação de probabilidade e de impacto e, depois, agregando essas avaliações numa única classificação de prioridade de risco. Ver também *risco de produto*, *risco de projeto*, *risco, impacto do risco*, *nível do risco*, *probabilidade do risco*.

ATA **avaliação heurística:** (heuristic evaluation) Técnica de análise de usabilidade que tem como alvo os problemas de usabilidade na interface com o utilizador ou com a design de interface. Com esta técnica, os revisores examinam a interface e avaliam a sua conformidade com os princípios de usabilidade reconhecidos (as "heurísticas").

EITP **avaliador:** (assessor) Pessoa que realiza uma avaliação; qualquer membro de uma equipa de avaliação.

## B

EITP **balanced scorecard:** (balanced scorecard) Ferramenta estratégica de gestão de desempenho para medir se as atividades operacionais de uma empresa estão alinhadas com os seus objetivos em termos de negócio, visão e estratégia. Ver também *painel de instrumentos corporativos*, *scorecard*.

**baseline:** (baseline) Especificação ou produto de software que tenha sido formalmente revisto ou acordado e que, em seguida, serve de base para o desenvolvimento podendo apenas ser alterado através de um processo formal de controlo de alterações. [Depois de IEEE 610]

**base para testes aprovada:** (frozen test basis) Documento de base para testes que só pode ser alterado por um processo formal de controlo de alterações. Ver também *baseline*.

F **base para testes:** (test basis) Todos os documentos a partir dos quais os requisitos de um componente ou sistema podem ser inferidos. É a documentação na qual os casos de teste são baseados. Se um documento apenas puder ser alterado por meio de um procedimento de alteração formal, a base para testes é chamada de base para testes aprovada. [Depois de TMap]

**bateria base de testes:** (basis test set) Conjunto de casos de teste derivado da estrutura interna de um componente ou da sua especificação que garante o alcance de 100% de cobertura relativamente a um critério de cobertura especificado.

F **bateria de testes:** (test suite; test set) Conjunto de vários casos de teste para um componente ou sistema alvo de teste, onde a pós-condição de um teste é frequentemente utilizada como pré-condição para o teste seguinte.

**bloco básico:** (basic block) Sequência de uma ou mais instruções executáveis consecutivas que não contém ramos. Nota: Um nó num grafo de controlo de fluxo representa um bloco básico.

**buffer:** (buffer) Área ou dispositivo usado para armazenar dados temporariamente de forma a colmatar as diferenças na taxa de processamento dos dados, diferenças na ocorrência de

eventos ou na quantidade de dados que pode ser tratada pelos dispositivos ou pelos processos envolvidos na transferência ou utilização desses dados. [IEEE 610]

**buffer overflow:** (buffer overflow) Falha de acesso à memória devido à tentativa, por parte de um processo, de armazenar dados além dos limites de um *buffer* de comprimento fixo, resultando na substituição de dados em áreas de memória adjacentes ou à ocorrência de uma exceção de *overflow*. Ver também *buffer*.

F **bug:** (bug) Ver *defeito*.

## C

**caminho:** (path) Sequência de eventos, por exemplo, instruções executáveis de um componente ou sistema a partir de um ponto de entrada até um ponto de saída.

**caminho-dd:** (dd-path) Caminho entre duas decisões de um algoritmo, ou dois nós de decisão do grafo correspondente, que não inclui outras decisões. Ver também *caminho*.

**caminho de auditoria:** (audit trail) Caminho pelo qual a entrada original de um processo (por exemplo, dados) pode ser rastreada ao longo de todo o processo tendo como ponto de partida da análise a saída desse mesmo processo. Isto facilita a análise de defeitos e permite a realização de um processo de auditoria. [TMap]

**caminho executável:** (feasible path) Caminho para o qual existe um conjunto de valores de entrada (*input*) e de pré-condições que fazem com que ele seja executado.

**caminho não executável:** (infeasible path) Caminho que não pode ser executado por nenhum conjunto possível de valores de entrada (*input*).

**caminho no fluxo de controle:** (control flow path) Ver *caminho*.

**camuflagem de defeitos:** (defect masking; fault masking) Ocorrência de um defeito que previne a detecção de outro defeito. [IEEE 610]

ATM **Capability Maturity Model Integration (CMMI):** (CMMI) Estrutura que descreve os  
EITP elementos chave de um desenvolvimento eficaz de produto, bem como, do seu processo de manutenção. O Capability Maturity Model Integration abrange as melhores práticas de planejamento, engenharia e gestão de desenvolvimento e manutenção de produtos. O CMMI é o sucessor do CMM. [CMMI]

**característica:** (feature) Atributo de um componente ou sistema especificado ou atributo implícito na documentação de requisitos (por exemplo, restrições de fiabilidade, de usabilidade ou de concepção). [Depois de IEEE 610]

**característica de qualidade:** (quality characteristic) Ver *atributo de qualidade*.

**características de qualidade do software:** (software quality characteristics) Ver *atributo de qualidade*.

**características de um produto de software:** (software product characteristic) Ver *atributo de qualidade*.

**CASE:** (CASE) Sigla para Engenharia de Software Assistida por Computador (Computer Aided Software Engineering).

F **caso de teste:** (test case) Conjunto de valores de entrada, de pré-condições de execução, de  
ATM resultados esperados e de pós-condições de execução, desenvolvido para um objetivo

específico ou condição de teste, como seja, o acionar de um determinado ramo de execução ou a verificação do cumprimento de um requisito específico. [Depois de IEEE 610]

**caso de teste abstrato:** (abstract test case) Ver *caso de teste de alto nível*.

**caso de teste bloqueado:** (blocked test case) Caso de teste que não pode ser executado porque as pré-condições não são cumpridas.

**caso de teste concreto:** (concrete test case) Ver *caso de teste detalhado*.

ATA **caso de teste de alto nível:** (high level test case) Caso de teste sem valores concretos (ao nível de implementação) para os dados de entrada (*inputs*) e para os resultados esperados. Utilizam operadores lógicos e as instâncias dos valores reais ainda não estão definidas e/ou disponíveis. Ver também *caso de teste detalhado*.

ATA **caso de teste detalhado:** (low level test case) Caso de teste com dados concretos (ao nível da implementação) para os dados de entrada (*inputs*) e para os resultados esperados. Os operadores lógicos dos casos de teste de alto nível são substituídos pelos valores correspondentes aos objetivos desses operadores lógicos. Ver também *caso de teste de alto nível*.

ATA **caso de teste lógico:** (logical test case) Ver *caso de teste de alto nível*.

**caso de uso:** (use case) Sequência de transações num diálogo entre um ator e um componente ou sistema que produza um resultado tangível, onde esse ator pode ser um utilizador ou qualquer dispositivo que possa trocar informação com o sistema.

**CAST:** (CAST) Sigla para Teste de Software Assistido por Computador (Computer Aided Software Testing). Ver também *automação de testes*.

**categoria de defeito:** (defect category) Ver *tipo de defeito*.

**categorias de risco** (risk category) Ver *tipo do risco*.

**cenário de teste:** (test scenario) Ver *especificação do procedimento de teste*.

**certificação:** (certification) Processo de confirmação de que um componente, sistema, ou pessoa cumpre com os requisitos especificados. Um exemplo de mecanismo de certificação é a realização de um exame.

EITP **ciclo de Deming:** (Deming cycle) Ciclo iterativo de 4 passos para a resolução de problemas (planear, executar, verificar, agir), tipicamente utilizado em processos de melhoria. [Depois de Deming]

**ciclo de teste:** (test cycle) Execução do processo de teste sobre a versão mais elementar de um objeto de teste.

EITP **ciclo de vida do software:** (software lifecycle) Período de tempo que começa quando um produto de software é concebido e termina quando esse software já não está disponível para utilização. O ciclo de vida do software inclui tipicamente uma fase de conceptualização, uma fase de requisitos, uma fase de conceção, uma fase de implementação, uma fase de testes, uma fase de instalação e validação, uma fase de operação e manutenção e, por vezes, uma fase de descontinuação. De notar que estas fases podem-se sobrepor ou podem ser realizadas iterativamente.

**classe de equivalência:** (equivalence class) Ver *partição de equivalência*.

ATM **CMMI:** (CMMI) Ver *Capability Maturity Model Integration*.

EITP

**cobertura:** (coverage) Grau de execução, expresso em porcentagem, de um dado elemento de cobertura quando sujeito a uma bateria de testes.

**cobertura de caminhos:** (path coverage) Porcentagem de caminhos que foram executados por uma bateria de testes. 100% de cobertura de caminho implica 100% de cobertura LCSAJ.

F **cobertura de código:** (code coverage) Um método de análise que determina que seções do software foram executadas (cobertas) por uma bateria de testes e que seções não foram executadas. Por exemplo, cobertura de instruções, cobertura de decisões ou cobertura de condições.

**cobertura de combinação de condições:** (condition combination coverage) Ver *cobertura de múltiplas condições*.

**cobertura de combinações de condições de ramos:** (branch condition combination coverage) Ver *cobertura de múltiplas condições*.

**cobertura de condições:** (condition coverage) Porcentagem de resultados das condições que foram executadas por uma bateria de testes. A cobertura de 100% exige que cada condição elementar de uma decisão seja testada como Verdadeiro e Falso.

**cobertura de condições de ramo:** (branch condition coverage) Ver *cobertura de condições*.

**cobertura de condições e decisões:** (decision condition coverage) Porcentagem de todos os resultados de condições e resultados de decisões que foram executados por uma bateria de testes. 100% de cobertura de condições e decisões implica quer 100% de cobertura de condições, quer 100% de cobertura de decisões.

F **cobertura de decisões:** (decision coverage) Porcentagem de resultados de decisões de saída que foram executados por uma bateria de testes. 100% de cobertura de decisões implica quer 100% de cobertura de caminhos, quer 100% de cobertura de instruções.

**cobertura de determinação de condições:** (condition determination coverage) Ver *cobertura modificada de decisões e condições*.

**cobertura de fluxo de dados:** (data flow coverage) Porcentagem de pares definição-uso que foram executados por uma bateria de testes.

F **cobertura de instruções:** (statement coverage) Porcentagem de instruções executáveis que foram exercitadas por uma bateria de testes.

**cobertura de múltiplas condições** (multiple condition coverage) Porcentagem de combinações, de todos os resultados de condições individuais dentro de uma instrução, que foram executadas por uma bateria de testes. 100% de cobertura de múltiplas condições implica 100% de cobertura modificada de decisões e condições.

**cobertura de partição de equivalência:** (equivalence partition coverage) Porcentagem de partições de equivalência que foram executadas por uma bateria de testes.

**cobertura de ramos:** (branch coverage) Porcentagem de ramos que foram executados por uma bateria de testes. 100% de cobertura de ramos implica quer 100% de cobertura de decisões quer 100% de cobertura de instruções.

**cobertura de sequência linear de código e salto:** (LCSAJ coverage) Porcentagem de LCSAJs de um componente que foi executado por uma bateria de testes. 100% de cobertura LCSAJs implica 100% de cobertura de decisões.

- F **cobertura de teste:** (test coverage) Ver *cobertura*.
- cobertura de valores fronteira:** (boundary value coverage) Percentagem de valores fronteira que foram executados por uma bateria de testes.
- cobertura estrutural:** (structural coverage) Medidas de cobertura baseadas na estrutura interna de um componente ou sistema.
- cobertura modificada de decisões e condições:** (modified condition decision coverage) Percentagem de resultados individuais de condições (que afetam de forma independente o resultado de decisões) que foram executados por uma bateria de testes. 100% de cobertura modificada de decisões e condições implica 100% de cobertura de condições e decisões.
- cobertura modificada de múltiplas condições:** (modified multiple condition coverage) Ver *cobertura modificada de decisões e condições*.
- cobertura N-switch:** (N-switch coverage) Percentagem de sequências de n+1 transições que foram executadas por uma bateria de testes. [Chow]
- ATT **condição atômica:** (atomic condition) Condição que não pode ser decomposta, isto é, uma condição que não contém duas ou mais condições individuais unidas por um operador lógico (AND, OR, XOR).
- código:** (code) Instruções informáticas e definições de dados expressos numa linguagem de programação ou numa forma de saída de um assembler, compilador ou outro tradutor. [IEEE 610]
- código inatingível:** (unreachable code) Código que não se pode atingir e, por isso, impossível de ser executado.
- código morto:** (dead code) Ver também *código inatingível*.
- ATT **co-existência:** (co-existence) Capacidade de um produto de software de coexistir com outros softwares independentes num ambiente de partilha de recursos comuns. [ISO 9126] Ver *portabilidade*.
- comité de controlo de alterações:** (change control board) Ver *comité de controlo de configurações*.
- ATM **comité de controlo de configurações (CCB):** (configuration control board) Grupo de pessoas responsável por avaliar e aprovar, ou reprovar, as alterações propostas para os elementos de configuração e por assegurar a implementação das alterações aprovadas. [IEEE 610]
- ATM **comité de gestão defeitos:** (defect management committee; defect triage committee) Equipa multifuncional cujos intervenientes gerem os defeitos reportados, desde a sua deteção inicial até à sua resolução final (por resolução entenda-se a correção do defeito, o adiamento da correção ou o cancelamento do registo desse defeito). Em alguns casos, este comité é representado pelo mesmo grupo que compõe o comité de controlo de configurações. Ver também *comité de controlo de configurações*.
- comparação de teste:** (test comparison) Processo de identificação das diferenças entre os resultados produzidos pelo componente ou sistema alvo de teste e os resultados esperados. A comparação de teste pode ser efetuada durante a execução do teste (comparação dinâmica) ou no final da execução do teste.
- comparação dinâmica:** (dynamic comparison) Comparação dos resultados esperados com os resultados obtidos, sendo essa comparação efetuada durante a execução do software.

Por exemplo, através de uma ferramenta de execução de testes.

**comparação pós-execução:** (post-execution comparison) Comparação dos resultados obtidos versus os esperados, no final da execução do software.

**comparador:** (comparator) Ver *comparador de teste*.

F **comparador de teste:** (test comparator) Ferramenta de teste que permite realizar a comparação automática dos resultados obtidos com os resultados esperados.

F **compilador:** (compiler) Ferramenta de software que traduz programas expressos numa linguagem de ordem superior na sua linguagem máquina equivalente. [IEEE 610]

F **complexidade:** (complexity) Grau de dificuldade em entender, manter e verificar a conceção, e/ou a estrutura interna, de um componente ou sistema. Ver também *complexidade ciclomática*.

ATT **complexidade ciclomática:** (cyclomatic complexity) O número máximo de caminhos lineares e independentes de um programa. Complexidade ciclomática é definida como:  $L - N + 2P$ , onde

- L = o número de arestas/ligações de um grafo;
- N = número de nós de um grafo;
- P = número de partes desconectadas do grafo (por exemplo, uma chamada a um segundo grafo ou subrotina). [Depois de McCabe]

**componente:** (component) Um item de software mínimo que pode ser testado isoladamente.

**comportamento:** (behavior) Resposta de um componente ou sistema a um conjunto de valores de entrada e de pré-condições.

EITP **comportamento codependente:** (codependent behavior) Dependência emocional ou psicológica excessiva sobre outra pessoa, especificamente em tentar mudar o comportamento atual (indesejável) dessa pessoa, enquanto se apoia a continuidade desse comportamento. Por exemplo, em testes de software, queixar-se da entrega tardia para os testes e, ao mesmo tempo, desfrutar do “heroísmo” das horas de trabalho adicionais necessárias para ganhar tempo quando a entrega está atrasada, reforçando esse atraso.

**comportamento no tempo:** (time behavior) Ver *desempenho*.

ATA **compreensibilidade:** (understandability) Capacidade do produto de software permitir ao utilizador a compreensão da adequação do mesmo e como ele pode ser utilizado para determinadas tarefas e condições de utilização. [ISO 9126] Ver também *usabilidade*.

F **conceção de teste:** (test design) (1) Ver especificação de conceção de testes. (2) Processo de  
ATM transformação dos objetivos de teste em condições de teste tangíveis e destas em casos de  
ATA teste.

**condição:** (condition) Expressão lógica que pode ser avaliada como Verdadeiro ou Falso. Por exemplo,  $A > B$ . Ver também *condição de teste*.

**condição composta:** (compound condition) Duas ou mais condições individuais unidas por meio de um operador lógico (AND, OR ou XOR). Por exemplo, ' $A > B$  AND  $C > 1000$ '.

**condição de ramo:** (branch condition) Ver *condição*.

F **condição de teste:** (test condition) Elemento ou evento de um componente ou sistema que  
ATM pode ser verificado por um ou mais casos de teste, ex., uma função, uma transação, uma  
ATA funcionalidade, um atributo de qualidade ou um elemento estrutural.

**configuração:** (configuration) Composição de um componente ou sistema, definido pelo

número, natureza e interligações das suas partes constituintes.

**conformidade:** (compliance) Capacidade que um produto de software tem para aderir a normas, convenções ou regulamentações previstas nas leis e prescrições similares. [ISO 9126]

**conjunto de casos de teste:** (test case suite) Ver *bateria de testes*.

**consistência:** (consistency) Grau de uniformidade, padronização e ausência de contradições entre os documentos ou partes de um componente ou sistema. [IEEE 610]

F **controlador:** (driver) Componente de software ou ferramenta de teste que substitui o componente responsável pelo controlo e/ou chamada de um componente ou sistema. [Depois de TMap]

**controlador de testes:** (test driver) Ver *controlador*.

**controlo de alterações:** (change control) Ver *controlo de configuração*.

**controlo de configuração:** (configuration control) Um elemento da gestão de configurações, que consiste na avaliação, coordenação, aprovação ou reprovação, e implementação de alterações nos elementos de configuração após a respetiva identificação formal. [IEEE 610]

**controlo de qualidade:** (quality control) Técnicas e atividades operacionais, da gestão da qualidade, que estão focadas em cumprir com os requisitos de qualidade. [Depois de ISO 8402]

ATM **controlo de risco:** (risk control) Processo através do qual se tomam decisões e se  
ATA implementam medidas de redução do risco para um dado nível, ou para o manter dentro  
ATT de determinados limites.

F **controlo de testes:** (test control) Atividade de gestão de testes que lida com o  
ATM desenvolvimento e aplicação de um conjunto de ações corretivas para voltar a colocar um  
ATA projeto de testes dentro do planeado quando a monitorização revela um desvio relativamente ao planeamento. Ver também *gestão de testes*.

F **controlo de versão:** (version control) Ver *controlo de configuração*.

F **COTS:** (COTS) Sigla inglesa para software comercial de prateleira (commercial off-the-shelf). Ver *software comercial de prateleira*.

**critério de aceitação:** (acceptance criteria) Critério de saída que um componente ou sistema deve cumprir para ser aceite por um utilizador, cliente ou outra entidade autorizada. [IEEE 610]

**critério de conclusão:** (completion criteria) Ver *critério de saída*.

**critério de conclusão dos testes:** (test completion criteria; test target) Ver *critério de saída*.

F **critério de entrada:** (entry criteria) Conjunto de condições genéricas e específicas que permitem que um processo avance uma determinada tarefa como é o exemplo do avançar para a fase de testes. O objetivo dos critérios de entrada é prevenir o início de uma tarefa/etapa que representaria um maior (desperdício de) esforço comparado com o esforço necessário para cumprir com o critério de entrada. [Gilb e Graham]

**critério de recomeço:** (resumption criteria) Critérios utilizados para reiniciar a totalidade ou uma parte das actividades de teste que foram anteriormente suspensas.

F **critério de saída:** (exit criteria) Conjunto de condições genéricas e específicas acordadas  
ATM com as partes interessadas (stakeholders) para se considerar um processo como

- ATA oficialmente terminado. O objetivo do critério de saída é prevenir que uma dada tarefa seja considerada terminada quando persistem partes significativas dessa tarefa que permanecem por finalizar. O critério de saída é utilizado para planejar e determinar quando se deve terminar os testes. [Gilb e Graham]
- critério de suspensão:** (suspension criteria) Critério usado para parar (temporariamente) todas, ou uma parte, das atividades de teste sobre os itens de teste. [Depois de IEEE 829]
- critérios para aprovação/reprovação:** (pass/fail criteria) Regras de decisão utilizadas para determinar se um item de teste (função) ou uma dada característica foi aprovada ou reprovada nesse teste. [IEEE 829]
- F **cronograma de execução de testes:** (test execution schedule) Esquema para a execução dos procedimentos de teste. Nota: Os procedimentos de teste são incluídos no cronograma de execução de testes segundo um contexto e na ordem pela qual serão executados.
- cronograma de testes:** (test Schedule) Lista de atividades, tarefas ou eventos do processo de teste, identificando as datas pretendidas para início e fim e/ou duração, e suas interdependências.
- ATM **CTP:** (CTP) Ver *processos críticos de teste*.
- EITP
- ATT **curto-circuito:** (short-circuiting) Técnica de linguagem de programação/interpretação para avaliar condições compostas em que a condição de um lado de um operador lógico poderá não ser avaliada se a condição do outro lado for suficiente para determinar o resultado final.
- custo da qualidade:** (cost of quality) Custos totais incorridos nas atividades e nos problemas de qualidade e geralmente divididos em custos de prevenção, custos de avaliação, custos de falhas internas e custos de falhas externas.

## D

- F **dados de teste:** (test data) Dados existentes (por exemplo, numa base de dados) antes da execução do teste que afetam o componente ou sistema alvo de teste.
- debugger:** (debugger) Ver *ferramenta de debug*.
- F **debugging:** (debugging) Processo de encontrar, analisar e remover as causas da falha no software.
- decisão:** (decision) Ponto no programa onde existem dois ou mais caminhos alternativos para o fluxo do programa. Pode ser representado como um nó com duas ou mais ligações a ramos separados.
- F **defeito:** (defect; fault) Imperfeição num componente ou sistema que pode provocar o mau funcionamento desse componente ou sistema. Por exemplo, uma instrução ou definição de dados incorreta. Um defeito, se for detetado durante a execução do software, pode provocar uma falha no componente ou sistema.
- ATM
- ETM **definição de perfil operacional:** (operational profiling) O processo de desenvolvimento e implementação de um perfil operacional. Ver também *perfil operacional*.
- definição dos dados:** (data definition) Instrução executável na qual uma variável recebe um determinado valor.

- F **densidade de defeitos:** (defect density; fault density) Número de defeitos identificados num componente ou sistema dividido pelo tamanho desse componente ou sistema (expresso em termos de medida padrão como são exemplo as linhas de código, o número de classes ou os pontos de função).
- desempenho:** (performance) Grau de cumprimento de um sistema ou componente de acordo com restrições de tempo de processamento e da taxa de rendimento (*throughput*). [IEEE 610]. Ver também *eficiência*.
- EITP **desenvolvimento ágil de software:** (agile software development) Grupo de metodologias de desenvolvimento de software baseado em desenvolvimento incremental e iterativo, onde os requisitos e as soluções evoluem através da colaboração entre equipas auto-organizadas e multifuncionais.
- desenvolvimento guiado por características:** (feature-driven development) Processo de desenvolvimento de software iterativo e incremental conduzido a partir da perspetiva de características (funcionalidade) com valor para o cliente. Desenvolvimento guiado por características é usado principalmente em desenvolvimento ágil de software. Ver também *desenvolvimento ágil de software*.
- F **desenvolvimento orientado a testes:** (test driven development) Processo de  
ETM desenvolvimento de software onde os casos de teste são concebidos e, frequentemente automatizados, antes do software estar desenvolvido.
- desvio:** (deviation) Ver *incidente*.
- EITP **diagnóstico (IDEAL):** (diagnosing (IDEAL)) Fase dentro do modelo IDEAL onde se avalia a situação atual face à desejada. Esta fase consiste nas atividades de caracterização do estado atual e do estado desejado bem como as respetivas recomendações de evolução. Ver também *IDEAL*.
- EITP **diagrama causa-efeito:** (cause-effect diagram) Representação gráfica utilizada para organizar e exibir as inter-relações das várias causas possíveis (raízes) de um problema. As possíveis causas de um defeito ou falha real, ou potencial, são organizadas em categorias e subcategorias numa estrutura em árvore, representada na horizontal com o defeito (potencial) ou falha representados como nó na raiz. [Depois de Juran].
- diagrama de estados:** (state diagram) Diagrama que descreve os estados que um componente ou sistema pode assumir, bem como os eventos ou circunstâncias que podem causar e/ou resultar da transição de um estado para outro. [IEEE 610]
- ETM **diagrama de Ishikawa:** (Ishikawa diagram) Ver *diagrama causa-efeito*.
- diagrama em espinha:** (fishbone diagram) Ver *diagrama causa-efeito*.
- ATM **diretor de testes:** (test director) Gestor sénior que administra os gestores de teste. Ver também *gestor de teste*.
- disponibilidade:** (availability) Grau em que um componente ou sistema está operacional e acessível ao longo do período em que é necessária a sua utilização. Muitas vezes, este valor é expresso em percentagem. [IEEE 610]
- domínio:** (domain) Conjunto a partir do qual os valores válidos de entrada (*input*) e de saída (*output*) podem ser selecionados.
- domínio de entrada:** (input domain) Conjunto a partir do qual se podem selecionar os valores de entrada (*input*) válidos. Ver também *domínio*.
- domínio de saída:** (output domain) Conjunto a partir do qual se podem selecionar valores de

saída (*output*) válidos. Ver também *domínio*.

## E

- F **efeito da monitorização:** (probe effect) Efeito causado no componente ou sistema pelo instrumento de medição quando esse componente ou sistema está a ser medido, nomeadamente, por uma ferramenta de teste ou monitorização de desempenho. Por exemplo, o desempenho do software poderá ser um pouco pior quando as ferramentas de teste de desempenho estão a ser utilizadas.
- ATM **eficácia:** (effectiveness) Capacidade de produção de um resultado pretendido. Ver também *eficiência*.
- ATM **eficiência:** (efficiency) (1) Capacidade que um produto de software tem de responder com o desempenho apropriado, relativamente à quantidade de recursos utilizados sob determinadas circunstâncias. [ISO 9126] (2) A capacidade de um processo para produzir o resultado pretendido, em relação à quantidade de recursos utilizados.
- ATT
- elemento de configuração:** (configuration item) Agregação de hardware, software ou ambos, que é designada para a gestão de configurações e tratada como uma entidade única nesse processo de gestão de configurações. [IEEE 610]
- emulador:** (emulator) Dispositivo, programa de computador ou sistema que aceita os mesmos valores de entrada (*inputs*) e produz os mesmos valores de saída (*outputs*) de um dado sistema. [IEEE 610]. Ver também *simulador*.
- F **engano:** (mistake) Ver *erro*.
- entrada:** (input) Variável lida por um componente (armazenada dentro ou fora desse componente).
- entrada de teste:** (test input) Dados recebidos a partir de uma fonte externa pelo objeto alvo de teste durante a execução do teste. A fonte externa pode ser hardware, software ou humana.
- entrada especificada:** (specified input) Entrada para a qual a especificação prevê o resultado.
- entrega:** (deliverable) Qualquer produto (de trabalho) que tem de ser disponibilizado a alguém que não o autor do produto (de trabalho).
- entrega de teste:** (test deliverable) Qualquer produto (de trabalho) de teste que tem de ser disponibilizado a alguém que não o autor do produto (de trabalho). Ver também *entrega (deliverable)*.
- entrega diária:** (daily build) Atividade diária de desenvolvimento na qual um sistema é compilado e completamente integrado (normalmente durante a noite) para que esse sistema esteja e contenha as últimas alterações em qualquer momento.
- F **equipamento de teste:** (test harness) Ambiente de teste composto pelos simuladores e controladores necessários para executar um teste.
- F **erro:** (error) Ação humana que produz um resultado incorreto. [IEEE 610]
- escala de medida:** (measurement scale) Escala que restringe o tipo de análise de dados que

pode ser desempenhado sobre si própria. [ISO 14598]

**escalabilidade:** (scalability) Capacidade de atualização de um produto de software de modo a receber aumentos de carga na sua utilização. [Depois de Gerrard]

**especificação:** (specification) Documento que especifica, de preferência de uma forma completa, precisa e verificável, os requisitos, conceção, comportamento ou outras características de um componente ou sistema e, frequentemente, os procedimentos para determinar se estas necessidades foram satisfeitas. [IEEE 610]

F **especificação de caso de teste:** (test case specification) Documento que especifica um conjunto de casos de teste (objetivo, dados de entrada, ações de teste, resultados esperados, e pré-condições de execução) para um item de teste. [Depois de IEEE 829] Ver também *especificação do teste*.

**especificação de componentes:** (component specification) Descrição da função de um componente em termos dos valores de saída para determinados valores de entrada, sob determinadas condições, bem como o comportamento não-funcional requerido (por exemplo, na utilização de recursos).

**especificação da conceção de testes:** (test design specification) Documento que especifica as condições de teste (itens contemplados) para um item de teste, bem como a abordagem de testes escolhida e a identificação dos casos de teste de alto nível relacionados. [Depois de IEEE 829]. Ver também *especificação do teste*.

F **especificação do procedimento de teste:** (test procedure specification) Documento que especifica a sequência de ações para a execução de um teste. Também conhecido como guião de teste ou guião de teste manual. [Depois de IEEE 829] Ver também *especificação do teste*.

ATM **especificação do teste:** (test specification) Documento que consiste na especificação da conceção do teste, do caso de teste e/ou do procedimento de teste.

EITP **estabelecer (IDEAL):** (establishing (IDEAL)) Fase dentro do modelo IDEAL, onde são planeados os detalhes de como uma organização vai chegar ao destino desejado. A fase estabelecer consiste nas atividades: estabelecer prioridades, desenvolver a abordagem e planear as ações. Ver também *IDEAL*.

ATT **estabilidade:** (stability) Capacidade do produto de software em evitar efeitos inesperados após ter sofrido alterações. [ISO 9126] Ver também *manutenibilidade*.

ATM **estimativa de teste:** (test estimation) Cálculo aproximado tendo em conta vários aspetos do teste (por exemplo, esforço, data de conclusão, custos envolvidos, número de casos de teste, etc.) que é utilizado mesmo quando os dados de entrada estão incompletos, incertos ou difusos.

**estimativa de três pontos:** (three point estimation) Método de estimativa de teste que usa valores estimados para o "melhor caso", para o "pior caso", e para o "caso mais provável" do tema a estimar, para aumentar o grau de certeza associado à respetiva estimativa.

F **estratégia de teste:** (test strategy) Descrição de nível superior dos vários níveis de teste a executar, e dos testes dentro desses níveis, para uma organização ou para um programa (um ou mais projetos).

ATM

ATA

ETM

**exatidão:** (accuracy) Capacidade que um produto de software tem para fornecer os resultados ou efeitos corretos ou acordados dentro do nível desejado de precisão. [ISO 9126] Ver

também *funcionalidade*.

F **execução de teste:** (1) (test execution) Processo de execução de um teste no componente ou  
ATM sistema alvo de teste produzindo resultados. (2) (test run) Execução de um teste numa  
ATA versão específica do objecto de teste.

**executado:** (exercised) Um elemento de um programa diz-se executado por um caso de teste quando o valor de entrada (*input*) provoca a execução desse elemento, seja ele uma instrução, uma decisão ou qualquer outro elemento estrutural.

EITP **eXtreme Programming:** (extreme programming) Metodologia de desenvolvimento utilizada em desenvolvimento ágil de software (*agile software development*) onde as práticas fundamentais são a programação em pares, a realização extensiva de revisões de código, testes unitários a todo o código, e uma grande simplificação e clareza no código. Ver também *desenvolvimento ágil de software*.

## F

EITP **fator crítico de sucesso:** (critical success factor) Elemento necessário para que uma organização ou projeto alcancem a sua missão. Fatores críticos de sucesso são os fatores críticos ou atividades necessárias para assegurar o sucesso.

F **falha:** (failure) Desvio na entrega, no serviço ou no resultado esperado de um componente  
ATM ou sistema. [Depois de Fenton]

ATM **falso negativo:** (false-negative result; false-pass result) Resultado de teste que não identifica a presença de um defeito que realmente existe nesse objeto de teste.

ATM **falso positivo:** (false-positive result; false-fail result) Resultado de teste que indica a presença de um defeito que realmente não existe nesse objeto de teste.

ATA **fase de contenção:** (phase containment) Percentagem de defeitos que são removidos na mesma fase do ciclo de vida de software em que foram introduzidos.

**fase de definição de requisitos:** (requirements phase) Período de tempo no ciclo de vida de um software durante o qual os requisitos são definidos e documentados. [IEEE 610]

**fase de execução de teste:** (test execution phase) Período de tempo no ciclo de vida de desenvolvimento de software no qual os componentes de um produto de software são executados e o produto de software é avaliado para determinar se os requisitos foram ou não satisfeitos. [IEEE 610]

**fase de teste:** (1) (test phase) Conjunto distinto de atividades de teste agrupadas numa fase gerível do projeto, por exemplo, as atividades de execução de um nível de teste. [Gerrard]. (2) (test stage) Ver *nível de teste*.

ATM **fecho dos testes:** (test closure) Durante o fecho de um processo de testes é recolhida informação das atividades concluídas de modo a consolidar a experiência, *testware*, os factos e os números. O fecho dos testes consiste na finalização e arquivo do *testware*, na avaliação do processo de teste, incluindo ainda a preparação de um relatório de avaliação do teste. Ver também *processo de teste*.

F **ferramenta de análise dinâmica:** (dynamic analysis tool) Ferramenta que fornece informação em tempo real (tempo de execução) sobre o estado do código. Estas ferramentas são normalmente utilizadas para identificar apontadores por atribuir

(unassigned pointers), para verificar a aritmética dos apontadores (pointer arithmetic) e para monitorar a alocação de memória através do uso e libertação do espaço de endereçamento, bem como, para assinalar memória perdida (memory leak).

- F **ferramenta de análise estática:** (static analysis tool) Ver *analizador estático*.
- ATT **ferramenta de captura/reprodução:** (capture/playback tool; capture/replay tool; record/playback tool) Tipo de ferramenta de execução de testes onde as entradas são registadas durante os testes manuais a fim de gerar os guiões de teste automatizados de forma a serem executados mais tarde (isto é, repetidos). Essas ferramentas são frequentemente utilizadas para apoiar os testes de regressão automatizados.
- F **ferramenta de cobertura:** (coverage tool) Ferramenta que indica, de forma objetiva, quais os elementos estruturais (por ex: instruções ou ramos) que foram executados por uma bateria de testes.
- ferramenta de controlo de defeitos:** (bug tracking tool) Ver *ferramenta de gestão de defeitos*.
- F **ferramenta de conceção de testes:** (test design tool) Ferramenta que auxilia a atividade de  
ATA conceção de teste gerando entradas de teste a partir de uma especificação que pode estar guardada num repositório CASE. Por exemplo, uma ferramenta de gestão de requisitos que gera casos de teste a partir de condições de teste especificadas na própria ferramenta, ou ferramenta que gera casos de teste a partir do código.
- F **ferramenta de debug:** (debugging tool) Ferramenta utilizada pelos programadores para  
ATT reproduzir as falhas, investigar o estado dos programas e encontrar o defeito (defect) correspondente. Os debuggers permitem executar os programas passo-a-passo, parar o programa em qualquer linha de código e examinar e atribuir valores às variáveis desse programa.
- F **ferramenta de execução de testes:** (test execution tool) Tipo de ferramenta que é capaz de  
ATA executar outro software utilizando um *script* automatizado, ex., captura/reprodução.  
ATT [Fewster and Graham]
- F **ferramenta de gestão de configurações:** (configuration management tool) Ferramenta que fornece suporte para a identificação e controlo dos elementos de configuração, para a identificação do seu estado decorrente das alterações e das várias versões, bem como da entrega de uma *baseline* de elementos de configuração.
- ferramenta de gestão de defeitos:** (defect management tool) Ferramenta que facilita o registo e controlo das alterações, bem como o controlo dos estados dos defeitos. Geralmente estas ferramentas disponibilizam funcionalidades orientadas ao fluxo de trabalho para rastrear e controlar a alocação, correção e re-teste dos defeitos, além de fornecerem relatórios. Ver também *ferramenta de gestão de incidentes*.
- F **ferramenta de gestão de incidentes:** (incident management tool) Ferramenta que facilita o registo e o controlo dos estados dos incidentes. Geralmente estas ferramentas disponibilizam funcionalidades orientadas ao fluxo de trabalho para rastrear e controlar a alocação, correção e re-teste dos incidentes, além de fornecerem relatórios. Ver também *ferramenta de gestão de defeitos*.
- F **ferramenta de gestão de requisitos:** (requirements management tool) Ferramenta que suporta o registo de requisitos e respetivos atributos (por ex: prioridade e responsável, entre outros). Também facilita o rastreamento, por níveis de requisitos, e a gestão de alterações desses requisitos. Algumas ferramentas de gestão de requisitos também

oferecem recursos para análise estática, tais como, verificação de consistência e violação de regras predefinidas.

F **ferramenta de gestão de testes:** (test management tool) Ferramenta que apoia a gestão de testes e que permite o controlo de parte do processo de testes. Possui frequentemente várias funcionalidades, tais como, a gestão do *testware*, o cronograma de testes, o registo dos resultados, o acompanhamento do progresso, a gestão de incidentes e a produção de relatórios de testes.

**ferramenta de injeção de erros:** (error seeding tool) Ver *ferramenta de injeção de falhas*.

ATT **ferramenta de injeção de falhas:** (fault seeding tool) Ferramenta para a injeção (ou seja, inserção propositada) de falhas num componente ou sistema.

**ferramenta de medição de cobertura:** (coverage measurement tool) Ver *ferramenta de cobertura*.

F **ferramenta de modelação:** (modeling tool) Ferramenta que suporta a criação, alteração e verificação dos modelos do software ou sistema. [Graham]

F **ferramenta de monitorização:** (monitoring tool) Ver *monitor*.

F **ferramenta de preparação de dados de teste:** (test data preparation tool) Tipo de ATA ferramenta que possibilita que os dados sejam selecionados, criados, gerados, manipulados e editados para serem utilizados em testes, a partir de bases de dados existentes.

**ferramenta de controlo de defeitos:** (defect tracking tool) Ver *ferramenta de gestão de defeitos*.

F **ferramenta de revisão:** (review tool) Ferramenta que dá apoio ao processo de revisão. As funcionalidades típicas deste tipo de ferramentas incluem suporte para planeamento e acompanhamento da revisão, suporte na comunicação, revisão colaborativa e um repositório para recolha e divulgação das métricas.

F **ferramenta de segurança:** (security tool) Ferramenta que dá suporte à segurança operacional.

EITP **ferramenta de teste:** (test tool) Produto de software que suporta uma ou mais atividades de teste, tais como, o planeamento e controlo, especificação, construção de ficheiros e dados iniciais, execução de testes e análise de testes. [TMap] Ver também *CAST*.

ATT **ferramenta de teste a hiperligações:** (hyperlink test tool) Ferramenta utilizada para verificar que não existem hiperligações perdidas num portal web.

F **ferramenta de teste de carga:** (load testing tool) Ferramenta para apoiar o teste de carga que permite simular carga aumentando, por exemplo, o número de utilizadores e/ou transações simultâneas dentro de um dado período de tempo. Ver *ferramenta de teste de desempenho*.

F **ferramenta de teste de desempenho:** (performance testing tool) Ferramenta que dá suporte ATT ao teste de desempenho e que, normalmente, tem dois recursos principais: a geração de carga e a medição do tempo da transação alvo de teste. A geração de carga pode simular tanto utilizadores múltiplos como altos volumes de dados de entrada (*input*). Durante a execução, as medições dos tempos de resposta são feitas a partir de transações selecionadas e depois registadas. Normalmente, as ferramentas de teste de desempenho fornecem relatórios baseados nos registos de teste e nos gráficos de carga em relação aos tempos de resposta.

**ferramenta de testes de segurança:** (security testing tool) Ferramenta que dá suporte ao teste de características de segurança e de vulnerabilidades.

F **ferramenta de teste de stress:** (stress testing tool) Ferramenta que suporta os testes de stress.

ATM **ferramenta *open source*:** (open source tool) Ferramenta de software que está disponível para todos os utilizadores potenciais em forma de código-fonte, geralmente através da Internet; a sua utilização é permitida geralmente sob licença, para estudar, mudar, melhorar e, por vezes, para distribuir o software.

ATM **ferramenta personalizada:** (custom tool) Ferramenta de software desenvolvida especificamente para um conjunto de utilizadores ou clientes.

**fiabilidade:** (reliability) Capacidade do produto de software para executar as funções necessárias, sob determinadas condições, por um período de tempo ou por um número determinado de execuções. [ISO 9126]

ATA **ficha de testes:** (test charter; charter) Declaração dos objetivos de teste e de possíveis ideias de como testar. As fichas de testes são utilizadas em testes exploratórios. Ver também *testes exploratórios*.

F **fluxo de controlo:** (control flow) A sequência de eventos (caminhos) na execução de um componente ou sistema.

F **fluxo de dados:** (data flow) Representação abstrata da sequência e possíveis mudanças do estado dos objetos de dados, sendo que os estados possíveis de um objeto são: criação, utilização e destruição. [Beizer]

F ***framework* de testes unitários:** (unit test framework) Ferramenta que fornece um ambiente para testes unitários ou de componente no qual esse componente pode ser testado. O teste pode ser feito isoladamente ou com recurso a simuladores e controladores apropriados. A *framework* também fornece outros apoios para o programador como, por exemplo, capacidades de debugging. [Graham]

**funcionalidade:** (functionality) Capacidade do produto de software de fornecer funcionalidades que respondam às necessidades pedidas ou implícitas sempre que o software é utilizado sob determinadas condições. [ISO 9126]

**funcionalidade de software:** (software feature) Ver *funcionalidade*.

## G

**garantia de qualidade:** (quality assurance) Parte da gestão da qualidade que se foca em fomentar confiança de que os requisitos de qualidade serão atingidos. [ISO 9000]

**gerador de testes:** (test generator) Ver *ferramenta de preparação de dados de teste*.

**gestão da mudança:** (change management) Abordagem estruturada para ajudar a transição de indivíduos, equipas e organizações de um estado corrente para um estado futuro desejado.

**gestão da qualidade:** (quality management) Atividades coordenadas para dirigir e controlar a qualidade numa organização. A direção e o controlo de qualidade geralmente incluem o estabelecimento da política e dos objetivos de qualidade, assim como o planeamento, o controlo, a garantia e a melhoria da qualidade. [ISO 9000]

**gestão de alterações:** (change management) (1) Abordagem estruturada para transitar

indivíduos ou organizações de um estado atual para um estado futuro. (2) Forma controlada para efetuar uma mudança, ou uma proposta de alteração, a um produto ou serviço. Ver também *gestão de configurações* e *gestão da mudança*.

- F **gestão de configurações:** (configuration management) Disciplina de aplicação de diretivas técnicas e administrativas, bem como, de monitorização para identificar e documentar as características funcionais e físicas de um elemento de configuração. Esta disciplina permite ainda controlar as alterações dessas características, registar e relatar o processo de alterações e verificar o estado da implementação e da conformidade com os requisitos especificados. [IEEE 610]
- gestão de dados de teste:** (test data management) Processo de análise de requisitos e de elaboração de estruturas de dados de teste, bem como criação e manutenção desses dados de teste.
- gestão de defeitos:** (defect management) Processo de identificação, investigação, ação e eliminação de defeitos. Envolve o registo e classificação de defeitos e ainda a avaliação do seu impacto. [Depois de IEEE 1044]
- F **gestão de incidentes:** (incident management) Processo que reconhece, investiga, toma ações e soluciona incidentes. Envolve o registo de incidentes, sua classificação e identificação do impacto. [Depois de IEEE 1044]
- gestão de problemas:** (problem management) Ver *gestão de defeitos*.
- EITP **Gestão de Qualidade Total:** (Total Quality Management) Abordagem de gestão de toda a organização centrada na qualidade, baseada na participação de todos os seus membros e visando o sucesso de longo prazo através da satisfação do cliente e benefícios para todos os membros da organização e para a sociedade. Gestão de Qualidade Total consiste em planear, organizar, dirigir, controlar e garantir. [Depois de ISO 8402]
- ATM **gestão de risco:** (risk management) Aplicação sistemática de procedimentos e práticas às tarefas de identificação, análise, priorização e controlo de riscos.
- ATA
- ATM **gestão de testes:** (test management) Planeamento, estimativa, monitorização e controlo das atividades de teste. Atividades geralmente levadas a cabo pelo gestor de testes.
- gestão de testes baseada em sessões:** (session-based test management) Método para medir e gerir testes baseados em sessões, por ex., testes exploratórios.
- F **gestor de testes:** (test manager) Pessoa responsável pela gestão de um projeto de testes, nomeadamente pela gestão das atividades, dos recursos e da avaliação de um objeto de teste. A pessoa que dirige, controla, administra, planeia e regula a avaliação de um objeto de teste.
- EITP **GQM:** (GQM – Goal Question Metric) Ver *Métrica Questão Objetivo*.
- ETM **gráfico de controlo:** (control chart) Ferramenta de controlo estatístico usada para monitorar um processo e determinar se é estatisticamente controlado. Representa graficamente o valor médio e os limites superior e inferior (os valores mais altos e mais baixos) de um processo.
- gráfico de invocações:** (call graph) Representação abstrata de relações de invocações entre subrotinas de um programa.
- ETM **Gráfico Shewhart:** (Shewhart chart) Ver *gráfico de controlo*.
- EITP **grafo causa-efeito:** (cause-effect graph) Representação gráfica de dados de entrada, e/ou estímulos (causas) com as suas saídas associadas (efeitos), que pode ser utilizada para a

conceção de testes.

**grafo de fluxo de controlo:** (control flow graph) Representação abstrata de todas as possíveis sequências de eventos (caminhos) na execução de um componente ou sistema.

**gravação do registo de teste:** (test logging; test recording) Processo de gravação de informações acerca dos testes executados num registo (*log*) de testes.

ATM **gravidade:** (severity) Grau de impacto que um defeito tem no desenvolvimento ou operação de um componente ou sistema. [Depois de IEEE 610]

EITP **Grupo do Processo de Teste:** (Test Process Group) Conjunto de especialistas (em testes) que facilitam a definição, manutenção, e melhoria do processo de testes utilizado por uma organização [Depois de CMMI].

**guia de instalação:** (installation guide) Instruções fornecidas por qualquer meio que orientam o processo de instalação. Pode ser um guia manual, um procedimento passo a passo, um assistente de instalação ou um qualquer outro processo similar.

F **guião de teste:** (test script) Geralmente utilizado para referir uma especificação do procedimento de teste, especialmente usado nos procedimentos automáticos.

ATM

## H

**hiperligação:** (hyperlink) Apontador numa página *web* que transporta para uma outra página *web*.

EITP **histórico de projeto:** (project retrospective) Processo estruturado para a captura das lições aprendidas (*lessons learned*) e a criação de um plano de ação tendo em vista melhorar o próximo projeto ou a próxima fase do projeto em curso.

## I

EITP **IDEAL:** Modelo de melhoria organizacional que funciona como guia para iniciar, planear e implementar ações de melhoria. O modelo IDEAL recebe o seu nome de acordo com as cinco fases que descreve: inicializar, diagnosticar, estabelecer, agir e aprender (learning).

**identificação da configuração:** (configuration identification) Elemento de gestão de configurações que consiste na seleção dos elementos de configuração de um sistema e no registo das suas características físicas e funcionais na documentação técnica. [IEEE 610]

ATM **identificação de risco:** (risk identification) Processo de identificação de riscos utilizando técnicas, tais como, *brainstorming*, listas de verificação ou histórico de falhas.

ATA

**impacto do risco:** (risk impact) O dano que será causado caso o risco se torne um resultado ou evento real.

ATM **implementação do teste:** (test implementation) Processo de desenvolvimento e priorização de procedimentos de teste, criação de dados de teste e, opcionalmente, preparação dos equipamentos de teste e escrita de guiões automáticos de teste.

ATA

F **incidente:** (incident) Qualquer evento ocorrido que requeira investigação. [Depois de IEEE 1008]

**incidente de teste:** (test incident) Ver *incidente*.

**incidente de teste de software:** (software test incident) Ver *incidente*.

F **independência do teste:** (independence of testing) Separação de responsabilidades que  
ATM encoraja a realização de testes objetivos e imparciais. [DO-178b]

EITP **indicador:** (indicator) Medida que pode ser utilizada para estimar ou prever uma outra  
medida. [ISO 14598]

**indicador de desempenho:** (performance indicator) Métrica de eficácia e/ou eficiência de alto nível utilizada para guiar e controlar o progresso do desenvolvimento. Por exemplo, atrasos no projeto de desenvolvimento de software. [CMMI]

**indicador de desempenho de teste:** (test performance indicator) Métrica de alto nível de eficácia e/ou eficiência utilizada para guiar e controlar o progresso do desenvolvimento de testes como, por exemplo, percentagem de deteção de defeitos (DDP - Defect Detection Percentage).

**indicador principal de desempenho:** (key performance indicator) Ver *indicador de desempenho*.

**indicador tipo Myers-Briggs:** (Myers-Briggs indicator) Um indicador de preferência psicológica representando as diferentes personalidades e estilos de comunicação das pessoas.

**infraestrutura de teste:** (test infrastructure) Artefactos organizacionais necessários para a execução de testes que consistem nos ambientes de teste, ferramentas de teste, ambiente físico e procedimentos.

EITP **inicializar (IDEAL):** (initiating (IDEAL)) Fase dentro do modelo IDEAL onde o trabalho de campo é efetuado para que o esforço de melhoria tenha sucesso. A fase “inicializar” consiste nas atividades: definir contexto, criação de apoios e infraestrutura. Ver também *IDEAL*.

**injeção de erros:** (error seeding) Ver *injeção de falhas*.

**injeção de falhas:** (fault injection) Processo de adição intencional de defeitos num sistema com a finalidade de descobrir se o sistema pode detectar e, eventualmente, recuperar, um defeito. Injecção de falhas destina-se a imitar as falhas que possam ocorrer em produção. Ver também *tolerância a falhas*.

[N. do T.] Embora a tradução mais correta seja injeção de defeitos, optou-se por manter o termo mais usual.

F **inspeção:** (inspection) Tipo de revisão realizada por pares para detetar defeitos sendo  
ATM baseada no exame visual de documentos. Por exemplo, violações das normas de  
EITP desenvolvimento e não conformidades face à documentação de alto nível. Trata-se da técnica de revisão mais formal e, portanto, está sempre baseada num procedimento documentado. [IEEE 610, IEEE 1028]. Ver também *revisão por pares*.

**inspetor:** (inspector) Ver *revisor*.

ATT **instalabilidade:** (installability) Capacidade que um produto de software tem para ser instalado num ambiente específico. [ISSO 9126]. Ver também *portabilidade*.

**instrução:** (statement) Entidade numa linguagem de programação que é, tipicamente, a unidade de execução mais pequena e indivisível.

**instrução de código fonte:** (source statement) Ver *instrução*.

**instrução executável:** (executable statement) Instrução que, quando compilada, é traduzida em código máquina que será executado proceduralmente na execução do programa e pode efetuar uma ação sobre dados.

**instrumentação:** (instrumentation) Inserção de código adicional no programa a fim de recolher informações sobre o seu comportamento durante a sua execução. Por exemplo, para medir a cobertura de código.

**instrumentador:** (instrumenter) Ferramenta de software utilizada para proceder à instrumentação.

**instrumentador de programa:** (program instrumenter) Ver *instrumentador*.

F **integração:** (integration) Processo que combina componentes ou sistemas em conjuntos maiores.

**integração funcional:** (functional integration) Abordagem de integração que combina os componentes ou sistemas com a finalidade de fazer com que a funcionalidade básica funcione o mais cedo possível. Ver também *teste de integração*.

**inteligência emocional:** (emotional intelligence) A capacidade ou habilidade para identificar, avaliar e gerir as suas próprias emoções, as dos outros e as do grupo.

**interoperabilidade:** (interoperability) Capacidade que um produto de software tem para interagir com um ou mais componentes ou sistemas específicos. [ISO 9126]. Ver também *funcionalidade*.

ETM **intervalo de confiança:** (confidence interval) Na gestão de riscos do projeto, o período de tempo em que uma ação de contingência deve ser implementada a fim de ser eficaz em reduzir o impacto do risco.

**item de cobertura:** (coverage item) Entidade ou propriedade utilizada como base para a cobertura de teste. Por exemplo, partições de equivalência ou instruções de código.

**item de teste:** (test item) Elemento individual a ser testado. Geralmente existe um objeto de teste e vários itens de teste. Ver também *objeto de teste*.

## L

EITP **líder de avaliação:** (lead assessor) Pessoa que lidera uma avaliação. Em alguns casos, por exemplo, CMMI e TMMi, quando as avaliações formais são realizadas, o líder de avaliação deve ser formado e certificado.

F **líder de teste:** (test leader) Ver *gestor de testes*.

F **linguagem de script:** (scripting language) Linguagem de programação em que são escritos os guiões de teste executáveis e utilizados por uma ferramenta de execução de testes (por ex.: uma ferramenta de captura/reprodução).

## M

**manifesto ágil:** (agile manifesto) Declaração sobre os valores que sustentam o desenvolvimento ágil de software. Os valores são:

- Indivíduos e interações mais do que processos e ferramentas;
- Software funcional mais do que documentação abrangente;
- Colaboração com o cliente mais do que negociação contratual;
- Responder à mudança mais do que seguir um plano.

EITP **manifesto de melhoria do processo de teste:** (test process improvement manifesto) Declaração que sobressai do manifesto ágil e que define os valores para melhoria do processo de testes. Os valores são:

- flexibilidade mais do que processos detalhados;
- melhores práticas mais do que modelos (templates);
- orientação à entrega (*deployment*) que orientação ao processo;
- revisões por pares mais do que (departamentos de) garantia de qualidade;
- orientação pelo negócio mais do que pelo modelo. [Veenendaal08]

**manutenção:** (maintenance) Modificação feita num produto de software após a sua entrega, tendo por finalidade corrigir defeitos, melhorar o desempenho ou outros atributos, ou adaptar o produto a uma alteração do ambiente. [IEEE 1219]

**manutenibilidade:** (maintainability) Facilidade com que um produto de software pode ser modificado para corrigir defeitos, responder a novos requisitos, facilitar manutenções futuras ou adaptar-se a um ambiente diferente. [ISO 9126]

**mapa mental:** (mind map) Diagrama utilizado para representar palavras, ideias, tarefas ou outros itens relacionados e organizados à volta de uma palavra ou ideia chave central. Mapas mentais são utilizados para gerar, visualizar, estruturar e classificar ideias, sendo ainda uma ajuda no estudo, organização, resolução de problemas, tomadas de decisão e escrita de documentos.

**máquina de estados finita:** (finite state machine) Modelo computacional que consiste num número finito de estados e de transições entre esses estados, podendo incluir as respetivas ações. [IEEE 610]

**marco:** (milestone) Determinado ponto de um projeto onde as entregas (*deliverables*) intermédias e os resultados devem estar prontos.

**marco de qualidade:** (quality gate) Marco (*milestone*) especial num projeto. Marcos de qualidade são colocados entre fases de projeto que dependem, em grande medida, dos resultados de uma fase anterior. Um marco de qualidade inclui uma verificação formal aos documentos da fase anterior.

ATA **matriz ortogonal:** (orthogonal array) Matriz bidimensional construída com propriedades matemáticas especiais de modo que, selecionando qualquer par de colunas da matriz, é possível obter todos os pares de combinações dos valores possíveis dessas colunas.

ETM **matriz RACI:** (RACI matrix) Matriz que descreve a participação de vários papéis na conclusão das tarefas ou entregas para um projeto ou processo. É especialmente útil para esclarecer os papéis e responsabilidades. RACI é um acrónimo derivado das quatro responsabilidades mais normalmente utilizados: (*Responsible*) Executa, (*Accountable*) Verifica, (*Consulted*) é Consultado, (*Informed*) tem de ser Informado.

ATT **maturidade:** (maturity) (1) Grau de eficácia e eficiência de uma organização relativamente aos seus processos e práticas de trabalho. Ver também *Capability Maturity Model*

*Integration, Test Maturity Model Integration.* (2) Capacidade do produto de software de evitar falhas resultantes de defeitos no software. [ISO 9126]. Ver também *fiabilidade*.

**medição:** (measurement) O processo de atribuir um número ou categoria a uma entidade a fim de descrever determinado atributo dessa entidade. [ISO 14598]

EITP **medida:** (measure) Número ou categoria que especifica um atributo de uma entidade por meio de medição. [ISO 14598]

**melhores práticas:** (best practice) Método superior ou práticas inovadoras que contribuam para a melhoria do desempenho de uma organização num determinado contexto, geralmente reconhecido como o “melhor” por outras organizações/pares.

**melhoria de processos:** (process improvement) Programa de atividades concebido para melhorar o desempenho e maturidade dos processos organizacionais bem como o resultado do próprio programa. [CMMI]

EITP **melhoria do processo de desenvolvimento de software:** (Software Process Improvement) Programa de atividades concebido para melhorar o desempenho e maturidade do processo de desenvolvimento de software da organização bem como os resultados do próprio programa. [Depois de CMMI]

ATT **memória perdida:** (memory leak) Defeito encontrado na lógica de alocação de armazenamento dinâmico de um dado programa, impedindo que liberte toda a memória após a ter utilizado, acabando por causar uma falha devido à falta de memória livre no sistema.

ATA **método de classificação em árvore:** (classification tree method) Técnica caixa-preta de conceção de testes na qual os casos de teste são descritos através de uma árvore de classificação e são concebidos para executar combinações representativas de domínios de entrada e/ou saída. [Grochtmann]

ETM **metodologia por objetivos S.M.A.R.T.:** (S.M.A.R.T. goal methodology) Metodologia em que os objetivos são definidos de forma muito específica em vez de genérica. SMART é uma sigla derivada dos atributos associados aos objetivos que se quer definir: específicos, mensuráveis, atingíveis, relevantes e definidos no tempo (*Specific, Measurable, Attainable, Relevant and Timely*).

F **métrica:** (metric) Escala e método utilizados para medição. [ISO 14598]

EITP

**métricas de cobertura de Chow:** (Chow's coverage metrics) Ver *cobertura N-switch*. [Chow]

ETM **métrica de convergência:** (convergence metric) Métrica que mostra o progresso em direção a um critério definido, por exemplo, convergência do número total de testes executados com o número total de testes planeados para execução.

EITP **Métrica Questão Objetivo:** (Goal Question Metric) Abordagem à medida do software recorrendo a um modelo com 3 níveis: nível conceptual (Objetivo), nível operacional (Questão) e nível quantitativo (Métrica).

ETM **missão de teste:** (test mission) O objetivo do teste para uma organização, muitas vezes documentada como parte da política de teste. Ver também *política de teste*.

ATM **mitigação do risco:** (risk mitigation) Ver *controlo de risco*.

ATA

ATT

- modelo baseado em conteúdo:** (content-based model) Modelo de processo que fornece uma descrição detalhada de boas práticas de engenharia. Por exemplo, práticas de teste. Ver também *CTP, processos críticos de teste, fator crítico de sucesso*.
- modelo de aumento da fiabilidade:** (reliability growth model) Modelo que ilustra o aumento da fiabilidade ao longo do tempo, através de testes contínuos, a um componente ou sistema, como consequência da remoção de defeitos que afetavam essa fiabilidade.
- EITP **modelo de ciclo de vida:** (lifecycle model) Divisão da vida de um produto ou projeto em fases. [CMMI] Ver também *ciclo de vida do software*.
- F **modelo de desenvolvimento incremental:** (incremental development model) Ciclo de vida de desenvolvimento que divide um projeto numa série de etapas, cada uma das quais entrega uma parte numa dada etapa da funcionalidade dos requisitos gerais desse projeto. Os requisitos são priorizados e entregues de acordo com a sua ordem de prioridade. Em algumas (mas, não em todas) versões deste modelo de ciclo de vida, cada subprojecto segue um “mini modelo em V”, com as suas próprias fases de conceção, codificação e teste.
- F **modelo de desenvolvimento iterativo:** (iterative development model) Ciclo de vida de desenvolvimento no qual um projeto é normalmente dividido num número grande de iterações. Dá-se o nome de iteração a um ciclo (*loop*) de desenvolvimento completo que resulta numa entrega (interna ou externa) de um produto executável num subconjunto do produto final em desenvolvimento, o qual cresce de iteração em iteração até se tornar no produto final.
- EITP **modelo de excelência da EFQM (European Foundation for Quality Management):** (EFQM (European Foundation for Quality Management) excellence model) Estrutura, não prescritiva, para o sistema de gestão da qualidade de uma organização, que foi definida e é propriedade da European Foundation for Quality Management, baseada em 5 critérios de “Capacidade” (englobando a atividade de uma organização) e 4 critérios de “Resultado” (englobando o que uma organização alcança).
- EITP **modelo de processos:** (process model) *Framework* onde processos da mesma natureza são classificados num modelo global como, por exemplo, um modelo de melhoria de testes.
- F **modelo em V:** (V-model) Estrutura para descrever as atividades do ciclo de vida de desenvolvimento do software, desde a especificação de requisitos até à manutenção. O modelo em V demonstra como as atividades de teste podem ser integradas em cada fase do ciclo de desenvolvimento do software.
- modelo embebido de desenvolvimento iterativo:** (embedded iterative development model) Submodelo de ciclo de vida de desenvolvimento que aplica uma abordagem iterativa com o objetivo de detalhar a conceção, codificação e teste dentro de um modelo sequencial global. Neste caso, os documentos de conceção (requisitos) de alto nível são preparados e aprovados para a globalidade do projeto, mas os documentos detalhados, para a conceção, desenvolvimento de código e testes, são realizados em iterações.
- F **moderador:** (moderator; inspection leader) Líder ou principal responsável pela inspeção ou  
ATM por qualquer outro processo de revisão.
- modo de falha:** (failure mode) Manifestação física ou funcional de uma falha. Por exemplo, um sistema em modo de falha (*failure mode*) pode ser caracterizado por lentidão na resposta, resultados incorretos ou pelo término completo da execução. [IEEE 610]
- EITP **modo de falha e análise de efeitos:** (Failure Mode and Effect Analysis (FMEA))

Abordagem sistemática na identificação de riscos e na análise de possíveis modos de falha e respectivos mecanismos de prevenção. Ver também *modo de falha, efeitos e análise crítica (FMECA)*.

**modo de falha de software e análise de efeitos:** (Software Failure Mode and Effect Analysis (SFMEA)) Ver *modo de falha e análise de efeitos (FMEA)*.

**modo de falha, efeitos e análise crítica:** (Failure Mode, Effects, and Criticality Analysis (FMECA)) Extensão do FMEA que inclui a análise crítica, sendo esta utilizada para mapear a probabilidade dos modos de falha confrontada com o impacto das suas consequências. O resultado destaca os modos de falha com maior probabilidade e maior impacto nas consequências, permitindo assim direcionar os esforços de correção para onde estes acrescentarão maior valor. Ver também *modo de falha e análise de efeitos (FMEA)*.

**modo de falha de software, efeitos e análise crítica:** (Software Failure Mode, Effects, and Criticality Analysis (SFMECA)) Ver *modo de falha, efeitos e análise crítica (FMECA)*.

**modelo de maturidade:** (maturity model) Coleção estruturada de elementos que descrevem certos aspetos da maturidade de uma organização e suportam a definição e compreensão dos processos organizacionais. Um modelo de maturidade geralmente fornece uma linguagem comum, uma visão partilhada e uma estrutura para a priorização das ações de melhoria.

**módulo:** (module) Ver *componente*.

**monitor:** (monitor) Ferramenta de software ou dispositivo de hardware, que é executado simultaneamente com o componente/sistema sob teste, e que supervisiona, grava e/ou analisa o comportamento desse componente/sistema. [Depois de IEEE 610]

F **monitorização de teste:** (test monitoring) Tarefa de gestão de testes que lida com as  
ATM atividades relacionadas com a verificação periódica do estado de um projeto de teste. Os  
ATA relatórios são construídos de modo a comparar o estado atual com o planeado. Ver  
também *gestão de testes*.

**MTBF:** Ver *tempo médio entre falhas*.

**MTTR:** Ver *tempo médio para reparar*.

**múltiplas condições:** (multiple condition) Ver *condição composta*.

ATT **mutabilidade:** (changeability) Capacidade que um produto de software tem para permitir a  
implementação de modificações especificadas. [ISO 9126] Ver também  
*manutenibilidade*.

## N

**não conformidade:** (non-conformity) Trata-se do não cumprimento de um requisito especificado. [ISO 9000]

**nível de integridade de software:** (software integrity level) O grau de conformidade sobre o qual um software cumpre ou deve cumprir face a um conjunto de softwares previamente selecionados pelos “stakeholders” (intervenientes) e/ou face a características de sistemas suportados por software (por exemplo, a complexidade do software, avaliação do risco, nível de segurança, desempenho desejado, fiabilidade ou custo) que

são definidas para refletir a importância desse software para os seus “stakeholders”.

EITP **nível de maturidade:** (maturity level) Grau de melhoria de um processo por todo um conjunto predefinido de áreas de processo, medido através de um conjunto predefinido de objetivos a atingir. [TMMi]

ATM **nível de risco:** (risk level) A importância de um risco é definida pelas suas características de impacto e probabilidade de ocorrência. O nível de risco pode ser utilizado para  
ATA determinar a intensidade dos testes a executar. Um nível de risco pode ser expresso tanto  
ATT em termos qualitativos (por ex: alto, médio, baixo) como quantitativos.

F **nível de teste:** (test level) Grupo de atividades de teste que são organizadas e geridas em  
ATM conjunto. O nível de teste está ligado às responsabilidades num projeto. Como exemplos de níveis de testes, temos testes de componentes, testes de integração, testes de sistema e testes de aceitação. [Depois de TMap]

EITP **norma:** (standard) Conjunto de requisitos formais, possivelmente obrigatórios, desenvolvidos e utilizados para estabelecer abordagens consistentes para a forma de trabalhar ou para fornecer orientações (por ex: normas da ISO/IEC, normas IEEE e normas da organização). [Depois de CMMI]

**nota de entrega:** (release note) Documento que identifica os itens de teste, as suas configurações, o seu estado atual e outras informações entregues pelo desenvolvimento para teste e, possivelmente, outras informações relevantes para outros intervenientes (*stakeholders*) no início da fase de execução de testes. [Depois de IEEE 829]

**número ciclomático:** (cyclomatic number) Ver *complexidade ciclomática*.

## O

**objeto de teste:** (test object) Componente ou sistema a ser testado. Ver também *item de teste*.

F **objetivo de teste:** (test objective) Razão ou propósito para conceber e executar um teste.

ATA **operabilidade:** (operability) Capacidade que um produto de software tem para facilitar a sua operação e controlo por parte do utilizador. [ISO 9126]. Ver também *usabilidade*.

**oráculo:** (oracle) Ver *oráculo de teste*.

**oráculo de teste:** (test oracle) Fonte para determinar os resultados esperados a comparar com o resultado obtido do software sob teste. Um oráculo pode ser o sistema existente (para referência), outro software, um manual de utilizador, ou uma pessoa com conhecimento especializado na área, mas não deverá ser o próprio código. [Depois de Adrion]

ATM **origem do defeito:** (root cause) Fonte de um defeito que, sendo removida, faz com que a ocorrência desse tipo de defeitos seja menor ou eliminada. [CMMI]

EITP **otimizador do processo de teste:** (test process improver) Pessoa que implementa melhorias no processo de teste baseado num plano de melhoria.

## P

- ETM **painel de instrumentos:** (dashboard) Representação de medidas dinâmicas do desempenho operacional de organizações ou atividades, com recurso a métricas representadas metaforicamente através de “contadores” e outros instrumentos, sendo similar aos disponíveis no painel de instrumentos de um automóvel, para que os efeitos dos eventos ou atividades possam ser facilmente compreendidos e relacionados com os objetivos operacionais. Ver também *painel de instrumentos corporativo, scorecard*.
- EITP **painel de instrumentos corporativo:** (corporate dashboard) Representação em forma de painel dos dados de desempenho da empresa. Ver também *balanced scorecard, painel de instrumentos*.
- par definição-uso:** (definition-use pair) Associação da definição de uma variável com a sua utilização. A utilização de variáveis inclui o uso computacional (por ex: multiplicação) ou o uso para direcionar o caminho de execução (expressão Booleana).
- partição de equivalência:** (equivalence partition) Subconjunto do domínio de valores de entrada (*input*) ou saída (*output*) para o qual se assume, baseado na especificação, que o comportamento do componente ou sistema é o mesmo.
- F  
ATA **particionar por equivalências:** (equivalence partitioning) Técnica caixa-preta de conceção de testes em que os casos de teste executam representantes das partições de equivalência. Por princípio, os casos de teste são definidos para executar cada partição, pelo menos, uma vez.
- EITP **percentagem de deteção de defeitos (PDD):** (Defect Detection Percentage (DDP); Fault Detection Percentage (FDP)) Número de defeitos encontrados numa fase de testes dividido pela soma desse número com o número de defeitos encontrados nas fases subsequentes.
- perfil de carga:** (load profile) Especificação das atividades que um componente ou sistema em testes pode encontrar em produção. Um perfil de carga consiste num número específico de utilizadores virtuais que processam um conjunto definido de transações num período de tempo delimitado e de acordo com o perfil operacional pré-definido. Ver também *perfil operacional*.
- perfil de desempenho:** (performance profiling) Tarefa de analisar, por exemplo, a identificação de estrangulamentos de desempenho baseado nas métricas geradas, e ajustar o desempenho de um componente de software ou sistema utilizando ferramentas.
- ETM  
ATT **perfil operacional:** (operational profile) Representação de um conjunto distinto de tarefas executadas pelo componente ou sistema, possivelmente baseado no comportamento do utilizador aquando da sua interação com esse componente ou sistema e a sua probabilidade de ocorrência. A tarefa é preferencialmente lógica versus física e é executada em várias máquinas ou executada em segmentos de tempo não contíguos.
- ATM  
ATA **planeamento de testes:** (test planning) Atividade de estabelecer ou atualizar o plano de testes.
- ETM **planeamento poker:** (planning poker) Técnica de estimativa baseada no consenso, principalmente utilizada para estimar esforço ou tamanho relativo de *user stories* no desenvolvimento de software ágil. É uma variação do método Wide Band Delphi usando um baralho de cartas com os valores que representam as unidades em que as equipas estimam. Ver também *desenvolvimento ágil de software, Wide Band Delphi*.
- plano de fase de testes:** (phase test plan) Plano de testes que normalmente aborda uma fase

de teste. Ver também *plano de testes*.

EITP **plano de melhoria de testes:** (test improvement plan) Plano para alcançar objetivos organizacionais de melhoria do processo de teste baseado na compreensão profunda dos pontos fortes e fracos desse processo e dos seus ativos na organização. [CMMI]

ATM **plano de revisão:** (review plan) Documento descrevendo a abordagem, recursos e calendário das atividades de revisão. Este documento identifica, entre outros: documentos e código a serem revistos, tipos de revisão a ser utilizados, participantes, assim como, critérios de entrada e saída a serem aplicados no caso de revisões formais, e as razões para a sua escolha. É um registo do processo de planeamento de revisão.

F **plano de testes:** (test plan) Documento descrevendo o âmbito, abordagem, recursos e  
ATM calendarização das atividades de teste. Identifica, entre outros itens de teste, as funcionalidades a serem testadas, as tarefas de teste, quem irá efetuar cada tarefa, o grau de independência do testador (*tester*), o ambiente de testes, as técnicas de conceção de teste e critérios de entrada e saída a serem utilizados, o racional para a sua escolha e quaisquer riscos que necessitem de planos de contingência. É um registo do processo de planeamento de testes. [Depois de IEEE 829]

**plano de teste do projeto:** (project test plan) Ver *plano mestre de testes*.

ATM **plano mestre de testes:** (master test plan) Plano de testes que aborda múltiplos níveis de teste. Ver também *plano de testes*.

ATM **plano para um nível de testes:** (level test plan) Plano de testes que aborda um nível de teste. Ver também *plano de testes*.

**plataforma de teste:** (test rig) Ver *ambiente de teste*.

F **política de teste:** (test policy) Documento de alto nível descrevendo os princípios, as  
ATM abordagens e os principais objetivos da organização em relação aos testes.

EITP  
ETM

**ponto de entrada:** (entry point) Instrução executável ou etapa de processo que define um ponto onde um determinado processo deve iniciar.

**ponto de saída:** (exit point) Instrução executável ou etapa de processo que define um ponto onde um determinado processo deve terminar.

**portabilidade:** (portability) Facilidade com que o produto de software pode ser transferido de um ambiente de hardware ou de software para outro ambiente diferente. [ISO 9126]

**pós-condição:** (postcondition) Condições de estado e ambiente que devem ser atingidos após a execução de um teste ou procedimento de teste.

**pré-condição:** (precondition) Condições de ambiente e de estado que devem ser atingidos antes que um componente ou sistema possa ser executado por um determinado teste ou procedimento de teste.

**pré-teste:** (pretest) Ver *teste de admissão*.

**predicado:** (predicate) Declaração que pode avaliar para verdadeiro ou falso e pode ser usado para determinar a fluxo de controlo de lógica de decisão posterior. Ver também *decisão*.

ATM **prioridade:** (priority) Nível de importância (de negócio) atribuído a um item. Por exemplo, um defeito.

**probabilidade do risco:** (risk likelihood) A probabilidade estimada do risco se vir a tornar um resultado ou evento real.

**problema:** (problem) Ver *defeito*.

F **procedimento de teste:** (test procedure) Ver *especificação do procedimento de teste*.  
ATM

**processo:** (process) Conjunto de atividades inter-relacionadas que transforma entradas (*inputs*) em saídas (*outputs*). [ISO 12207]

**processo de teste:** (test process) O Processo de Teste Fundamental abrange o planeamento e controlo de testes, a análise e conceção de teste, a sua implementação e execução, a avaliação do critério de saída e o *reporting* dos resultados assim como as atividades de fecho de teste.

ATM **processo sistemático de teste e avaliação:** (Systematic Test and Evaluation Process)  
EITP Metodologia de teste estruturada, também utilizada como modelo baseado em conteúdo para a melhoria do processo de testes. O processo sistemático de teste e avaliação (STEP) não obriga a que as melhorias ocorram numa ordem específica. Ver também *modelo baseado em conteúdo*.

ATM **processos de teste críticos:** (Critical Testing Processes) Modelo baseado em conteúdo para  
EITP melhoria do processo de teste e que foi construído em torno de doze processos críticos. Esses processos incluem processos de grande visibilidade, onde os intervenientes e gestores avaliam a sua capacidade de entrega e a sua relação com operações críticas (*mission critical*) em que o desempenho afeta diretamente os lucros e a reputação da empresa. Ver também *modelo baseado em conteúdo*.

**programação em pares:** (pair programming) Abordagem de desenvolvimento de software onde as linhas de código (produção e/ou teste) de um componente são escritas por dois programadores a trabalhar num único computador. Implicitamente, isto significa que as revisões de código são realizadas em tempo real.

**projeto:** (project) Um projeto é o conjunto único de atividades coordenadas e controladas, com datas para início e fim, realizado com a finalidade de alcançar um objetivo que responda aos requisitos específicos, inclusive os de restrição de tempo, custos e recursos. [ISO 9000]

**pseudoaleatório:** (pseudo-random) Série que aparenta ser aleatória, mas que, de facto, foi gerada de acordo com uma sequência pré-estabelecida.

## Q

F **qualidade:** (quality) Grau que estabelece em que medida um componente, sistema ou processo responde aos requisitos especificados e/ou às necessidades e expectativas do utilizador/consumidor. [Depois de IEEE 610]

EITP **qualidade baseada na produção:** (manufacturing-based quality) Visão de qualidade, onde a qualidade é medida pelo grau de conformidade de um produto ou serviço relativamente à sua especificação/conceção e requisitos. A qualidade deriva do(s) processo(s) utilizado(s). [Garvin]. Ver também *qualidade baseada no produto, qualidade baseada na transcendência, qualidade baseada no utilizador, qualidade baseada no custo*.

- EITP qualidade baseada na transcendência:** (transcendent-based quality) Visão de qualidade, onde a qualidade não pode ser definida com precisão, mas que é percebida quando está presente ou estamos conscientes da sua ausência quando tal não acontece. Qualidade depende da percepção e dos sentimentos afetivos de um indivíduo ou grupo de indivíduos para um produto. [Depois de Garvin] Ver também *qualidade baseada na produção, qualidade baseada no produto, qualidade baseada no utilizador, qualidade baseada no custo*.
- EITP qualidade baseada no custo:** (value-based quality) Visão de qualidade, onde a qualidade é definida pelo preço. Um produto ou serviço de qualidade é aquele que fornece o desempenho desejado a um custo aceitável. A qualidade é determinada por meio de um processo de decisão com as partes interessadas sobre os compromissos entre os aspetos de tempo, esforço e custo. [Depois de Garvin] Ver também *qualidade baseada na produção, qualidade baseada no produto, qualidade baseada na transcendência, qualidade baseada no utilizador*.
- EITP qualidade baseada no produto:** (product-based quality) Visão de qualidade, onde a qualidade é baseada num conjunto bem definido de atributos de qualidade. Estes atributos devem ser medidos de uma forma objetiva e quantitativa. As diferenças na qualidade dos produtos de um mesmo tipo podem ser atribuídas ao modo como atributos de qualidade específicos foram implementados. [Depois de Garvin] Ver também *qualidade baseada na produção, atributo de qualidade, qualidade baseada na transcendência, qualidade baseada no utilizador, qualidade baseada no custo*.
- EITP qualidade baseada no utilizador:** (user-based quality) Visão de qualidade, onde a qualidade é a capacidade de satisfazer necessidades e desejos do(s) utilizador(es). Um produto ou serviço que não preenche as necessidades dos utilizadores é improvável que atinja o sucesso. Esta é uma abordagem de contingência à qualidade, condicionada ao contexto, já que as diferentes características de negócio requerem diferentes qualidades para um produto. [Depois de Garvin] Ver também *qualidade baseada na produção, qualidade baseada no produto, qualidade baseada na transcendência, qualidade baseada no custo*.
- qualidade de dados:** (data quality) Atributo de dados que indica exatidão no que diz respeito a alguns critérios pré-definidos, por exemplo, expectativas de negócio, requisitos de integridade de dados, a consistência dos dados.
- qualidade de software:** (software quality) Totalidade das funcionalidades e características de um produto de software que afetam a sua capacidade de satisfazer necessidades explícitas ou implícitas. [Depois de ISO 9126] Ver também *qualidade*.
- qualificação:** (qualification) Processo de demonstração da capacidade de responder a requisitos específicos. Nota: o termo “qualificado” é utilizado para designar o estado associado à qualificação. [ISO 9000]

## R

**ramo:** (branch) Bloco básico que pode ser selecionado para execução com base na construção do programa em que um de dois ou mais caminhos alternativos estão disponíveis. Por exemplo, caminhos alternativos derivados de instruções *Case, Jump, Go-To, ou If-Then-Else*.

**F rastreabilidade:** (traceability) Capacidade de identificar os itens relacionados na

documentação e no software, tais como requisitos com testes associados. Ver também *rastreabilidade horizontal*, *rastreabilidade vertical*.

**rastreabilidade horizontal:** (horizontal traceability) Rastreabilidade dos requisitos para um dado nível de teste por meio de camadas de documentação de testes (por exemplo, plano de testes e especificação de concepção de testes, casos de teste, procedimento de testes ou guião de testes).

**rastreabilidade vertical:** (vertical traceability) Mapeamento dos requisitos através das várias camadas desde a documentação de desenvolvimento até aos componentes.

EITP **Rational Unified Process:** (Rational Unified Process) *Framework* proprietária para o processo de desenvolvimento de software, que é adaptável e iterativa, e que consiste num ciclo de 4 fases: início, elaboração, construção e transição.

**recuperabilidade:** (recoverability) Capacidade que um produto de software tem para restabelecer um nível específico de desempenho e de recuperar os dados diretamente afetados em caso de falha. [ISO 9126]. Ver também *fiabilidade*.

F **redator:** (scribe) Pessoa que regista num formulário cada defeito mencionado, bem como todas as sugestões de melhoria, durante uma reunião de revisão de processo. O redator deve assegurar a legibilidade e inteligibilidade do formulário de registo.

**registo da execução do teste:** (test run log) Ver *registo de teste*.

**registo de estados:** (status accounting) Elemento da gestão de configurações, que consiste no registo e reporte da informação necessária a uma gestão efetiva de configurações. Esta informação inclui uma lista da identificação da configuração aprovada, o estado das alterações propostas à configuração e o estado da implementação das alterações aprovadas. [IEEE 610]

F **registo de incidentes:** (incident logging) Registo dos detalhes de qualquer incidente ocorrido, por exemplo, durante o teste.

F **registo de teste:** (test log; test record) Registo cronológico dos detalhes relevantes sobre a  
ATM execução de testes. [IEEE 829]

EITP **relatório de avaliação:** (assessment report) Documento que resume os resultados da avaliação, por exemplo, conclusões, recomendações e resultados. Ver também *avaliação de processo*.

**relatório de avaliação de teste:** (test evaluation report) Documento produzido no final do processo de testes que resume todas as atividades e resultados do teste. Também contém a avaliação do processo de teste e as lições aprendidas.

**relatório de defeitos:** (defect report; bug report) Documento que reporta qualquer falha num componente ou sistema que pode provocar o mau funcionamento desse componente ou sistema. [IEEE 829]

**relatório de desvios:** (deviation report) Ver *relatório de incidentes*.

F **relatório de incidentes:** (incident report) Documento que relata qualquer evento ocorrido, por exemplo, durante o teste, que requeira investigação. [Depois de IEEE 829]

**relatório de incidentes de teste:** (test incident report) Ver *relatório de incidentes*.

**relatório de incidentes de teste de software:** (software test incident report) Ver *relatório de incidentes*.

**relatório de problema:** (problem report) Ver *relatório de defeitos*.

**relatório de progresso de testes:** (test progress report) Documento que resume as atividades e os resultados de teste, que é produzido em intervalos regulares, para reportar o progresso das atividades face a um plano base (nomeadamente o plano de testes original) e para comunicar riscos e medidas alternativas que necessitem de uma decisão da gestão.

**relatório de testes:** (test report) Ver *relatório sumário de testes, relatório de progresso de testes*.

**relatório de transmissão de item:** (item transmittal report) Ver *nota de entrega*.

**relatório de transmissão de item de teste:** (test item transmittal report) Ver *nota de entrega*.

F ATM **relatório sumário de testes:** (test summary report) Documento que resume as atividades de teste e os respetivos resultados. Contém também uma avaliação dos itens de teste contra os critérios de saída. [IEEE 829]

**representação contínua:** (continuous representation) Estrutura de um modelo de capacidade e maturidade em que os níveis de capacidade fornecem uma ordem recomendada para a abordagem de melhoria dos processos dentro de determinadas áreas de processo. [CMMI]

EITP **representação por níveis:** (staged representation) Estrutura de modelo em que, assim que os objetivos de um conjunto de áreas do processo sejam atingidos, é estabelecido um nível de maturidade; cada nível serve de base para os níveis subsequentes. [CMMI]

**reprodutibilidade dos testes:** (test reproducibility) Atributo de teste indicando se é sempre obtido o mesmo resultado de cada vez que o teste é executado.

F **requisito:** (requirement) Condição ou capacidade necessária para resolver um problema ou alcançar um objetivo que deve ser obrigatoriamente disponibilizado por um sistema, ou por um componente de sistema, a fim de satisfazer um contrato, norma, especificação ou um outro documento formalmente exigido. [Depois de IEEE 610]

**requisitos de recomeço:** (resumption requirements) Conjunto definido de atividades de teste que deve ser repetido quando o teste é reiniciado depois de uma suspensão. [IEEE 829]

**requisito de teste:** (test requirement) Ver *condição de teste*.

F **requisito funcional:** (functional requirement) Requisito que especifica uma função que um determinado componente ou sistema deve desempenhar. [IEEE 610]

F **requisito não funcional:** (non-functional requirement) Requisito que não se refere à funcionalidade, mas sim a atributos tais como fiabilidade, eficiência, usabilidade, manutenibilidade e portabilidade.

**requisitos testáveis:** (testable requirements) Requisitos expressos de forma a permitirem a conceção dos testes (e, posteriormente, casos de teste) e execução dos testes para determinar se os requisitos foram cumpridos. [Depois de IEEE 610]

**resultado:** (result; outcome) Consequência/resultado da execução de um teste. Inclui capturas de ecrã, mudanças nos dados, relatórios e comunicações enviadas. Ver também *resultado obtido, resultado esperado*.

**resultado da decisão:** (decision outcome) Resultado de uma decisão (que, por sua vez, determina quais os ramos a seguir).

**resultado de condição:** (condition outcome) Avaliação de uma condição em Verdadeiro ou

Falso.

**resultado do teste:** (test outcome; test result) Ver *resultado*.

**resultado esperado:** (expected result; expected outcome; predicted outcome)

Comportamento do componente ou sistema previsto pela especificação, ou por outra fonte, de acordo com as condições especificadas.

**resultado obtido:** (actual result; actual outcome) Comportamento produzido/observado quando um componente ou sistema é testado.

F **re-teste:** (re-testing) Fase onde se volta a executar casos de teste que falharam na última execução, de forma a verificar o sucesso das ações corretivas.

EITP **reunião de análise retrospectiva:** (retrospective meeting) Reunião realizada no final de um projeto na qual os membros da equipa desse projeto fazem a sua avaliação e retiram lições que podem ser aplicadas no projeto seguinte.

**reunião pós-projeto:** (post-project meeting) Ver *reunião de análise retrospectiva*.

F **revisão:** (review) Avaliação de um produto ou do estado de um projeto para averiguar  
ATM discrepâncias face aos resultados planeados e para recomendar melhorias. Exemplos: revisões da gestão, revisões informais, revisões técnicas, inspeções e apresentação de documento. [IEEE 1028]

**revisão ad hoc:** (ad hoc review) Ver *revisão informal*.

ATM **revisão de gestão:** (management review) Avaliação sistemática da aquisição, do fornecimento, do desenvolvimento, da operação ou do processo de manutenção de software. Tal avaliação pode ser efetuada pela equipa de gestão ou por alguém, em seu nome, a fim de monitorizar o progresso, determinar o estado dos planos e calendários, confirmar os requisitos e os seus sistemas de alocação ou para avaliar a eficácia da abordagem da gestão para fins a atingir. [IEEE 610, IEEE 1028]

**revisão de testabilidade:** (testability review) Verificação detalhada da base para testes com o objetivo de determinar se esta está num nível adequado de qualidade para servir como documento de entrada para o processo de teste. [Depois de TMap]

F **revisão formal:** (formal review) Revisão caracterizada por procedimentos e requisitos documentados. Por exemplo, uma inspeção.

F **revisão informal:** (informal review) Revisão que não é baseada num procedimento formal  
ATM (documentado).

F **revisão por pares:** (peer review) Revisão do trabalho de um produto de software feita por colegas do fabricante do produto, com a finalidade de identificar defeitos e sugerir melhorias. Como exemplo, podemos citar a inspeção, a revisão técnica e a apresentação de documento (*walkthrough*).

F **revisão técnica:** (technical review) Atividade de discussão em grupo que se foca em  
ATM alcançar um consenso sobre a abordagem técnica a ser adotada. [Gilb e Graham], [IEEE 1028] Ver também *revisão por pares*.

F **revisor:** (reviewer; checker) Pessoa envolvida na revisão que identifica e descreve  
ATM anomalias no produto ou no projeto sob revisão. Os revisores podem ser escolhidos para representar diferentes pontos de vista e diferentes papéis durante a revisão.

F **risco:** (risk) Fator que pode resultar em consequências negativas no futuro; normalmente é  
ATM expresso em termos de impacto ou probabilidade de ocorrência.

- F **risco de produto:** (product risk) Risco diretamente relacionado com o objeto de teste. Ver  
ATM também *risco*.
- F **risco de projeto:** (project risk) Risco relacionado com a gestão e o controle do projeto (de  
ATM teste), por exemplo, falta de pessoal, prazos rigorosos, alteração de requisitos, etc. Ver  
também *risco*.
- ATM **risco de qualidade:** (quality risk) Risco relacionado com um atributo de qualidade. Ver  
também *atributo de qualidade, risco de produto*.
- ATT **robustez:** (robustness) Grau até ao qual um componente ou sistema pode funcionar  
corretamente na presença de entradas inválidas ou na presença de condições de ambiente  
extremas [IEEE 610] Ver também *tolerância ao erro, tolerância a falhas*.
- EITP **RUP:** (RUP) Ver *Rational Unified Process*.

## S

- saída:** (output) Variável (armazenada dentro ou fora de um componente) escrita por um  
componente.
- scorecard:** (scorecard) Representação agregada de medidas de desempenho que ilustram o  
progresso em relação à implementação de objetivos de longo prazo. Um *scorecard*  
fornece medições estáticas do desempenho ao longo, ou no fim, de um determinado  
intervalo. Ver também *balanced scorecard, painel de instrumentos*.
- EITP **SCRUM:** (SCRUM) *Framework* iterativa e incremental de gestão de projetos largamente  
utilizada no desenvolvimento ágil de software. Ver também *desenvolvimento ágil de  
software*.
- segmento:** (subpath) Sequência de instruções executáveis num componente.
- segurança:** (1) (safety) Capacidade de um produto de software em atingir níveis aceitáveis  
no risco de causar danos a pessoas, negócios, software, propriedade ou meio ambiente,  
num contexto de utilização específico. [ISO 9126]. (2) (security) Atributos de um produto  
de software que se baseia na sua capacidade de prevenir acessos não autorizados, tanto  
deliberados como acidentais, a programas e dados. [ISO 9126] Ver também  
*funcionalidade*.
- sem sucesso:** (fail) Um teste considera-se como “sem sucesso” se o resultado obtido não  
corresponde ao resultado esperado.
- semear falhas:** (fault seeding; bebugging) Processo que acrescenta intencionalmente  
defeitos àqueles que já estão no componente ou sistema com a finalidade de monitorar a  
taxa de deteção e remoção, e estimar o número de defeitos restantes. Semear falhas faz  
normalmente parte do teste de desenvolvimento (pré-lançamento) e pode ser realizada em  
qualquer nível de teste (componente, integração ou sistema). [Depois IEEE 610]
- [N. do T.] Embora a tradução mais correta seja semear defeitos, optou-se por manter o  
termo mais usual.
- sensibilização de caminho:** (path sensitizing) Escolha de um conjunto de valores de entrada  
(*input*) para forçar a execução de um dado caminho.
- sequência linear de código e salto:** (LCSAJ - Linear Code Sequence and Jump) Consiste  
nos três itens seguintes (convencionalmente identificados pelo número da linha numa

listagem de código fonte): o início da sequência linear de instruções executáveis, o fim dessa sequência linear e a linha alvo para a qual o fluxo de controlo é transferido no final da sequência linear.

**sessão de teste:** (test session) Período ininterrupto de tempo despendido na execução de testes. Nos testes exploratórios, cada sessão de teste é focada numa ficha de testes, mas os testadores podem explorar novas oportunidades ou questões durante a sessão. O testador cria e executa casos de teste em tempo real e regista os seus progressos. Ver também *testes exploratórios*.

**simulação:** (simulation) Representação, por um sistema, de características comportamentais selecionadas de outro sistema físico ou abstrato. [ISO 2382/1]

**simulador:** (1) (simulator) Dispositivo, programa de computador ou sistema utilizado durante os testes, que se comporta ou opera como um determinado sistema quando recebe um conjunto de entradas (*inputs*) controladas. [IEEE 610, DO178b]. Ver também *emulador*. (2) (stub) Esqueleto, ou implementação específica de um componente de software, usado para desenvolver ou testar o componente que o chama, ou outro componente que dependa de si. Substitui o componente invocado. [Depois de IEEE 610]

**sistema:** (system) Conjunto de componentes organizados de modo a realizar uma função específica ou um conjunto de funções. [IEEE 610]

**sistema composto:** (system of systems) Sistemas heterogéneos múltiplos e distribuídos, que são incorporados em redes de vários níveis e em domínios múltiplos interligados, abordando em grande escala os problemas e objetivos (interdisciplinares) comuns, sem ter, geralmente, uma estrutura comum de gestão.

**sistema crítico:** (safety critical system) Sistema cuja falha ou mau funcionamento pode resultar na morte ou em lesões graves a pessoas, ou na perda ou danos severos a equipamentos, ou danos ambientais.

**sistema de sistemas:** (system of systems) Ver *sistema composto*.

**situação de teste:** (test situation) Ver *condição de teste*.

**software:** (software) São programas de computador, procedimentos, eventuais dados e documentação associada referente à operação de um sistema de computador. [IEEE 610]

**software à medida:** (bespoke software) Software desenvolvido especificamente para um conjunto de utilizadores ou clientes. O oposto é o software de prateleira (*off-the-shelf software*).

F **software comercial de prateleira:** (commercial off-the-shelf software) Ver *software de prateleira*.

**software de prateleira:** (off-the-shelf software) Produto de software desenvolvido para o mercado em geral, ou seja, para um grande número de consumidores e entregue a vários consumidores num mesmo formato.

**software personalizado:** (custom software) Ver *software à medida*.

**software standard:** (standard software) Ver *software de prateleira*.

ATA **Software Usability Measurement Inventory (SUMI):** Técnica de testes de usabilidade baseada em questionários para avaliar a qualidade do software do ponto de vista do utilizador final. [Veenendaal04]

**SPI:** (SPI) Ver *melhoria do processo de desenvolvimento de software*.

- ATM **STEP:** (STEP) Ver *processo sistemático de teste e avaliação*.
- EITP
- ATT **substitutibilidade:** (replaceability) Capacidade que um produto de software tem para ser utilizado em vez de um outro produto de software para os mesmos fins e no mesmo ambiente. [ISO 9126]. Ver também *portabilidade*.
- ATA **SUMI:** Ver *Software Usability Measurement Inventory*.

## T

- tabela de decisão:** (decision table) Tabela que associa os valores de entrada (*inputs*) e/ou estímulos (causas) com os respectivos valores de saída (*outputs*) e/ou ações (efeitos), podendo ser utilizada para a concepção de casos de teste.
- tabela de decisão causa-efeito:** (cause-effect decision table) Ver *tabela de decisão*.
- tabela de estados:** (state table) Grelha que exhibe as transições resultantes de cada estado combinado com cada evento possível mostrando, tanto as transições válidas, como as inválidas.
- F **taxa de falhas:** (failure rate) Rácio do número de falhas de uma dada categoria para uma dada unidade de medida. Por exemplo, falhas por unidade de tempo, por número de transações ou por número de execuções de computador. [IEEE 610]
- ATA **taxonomia de defeitos:** (defect taxonomy; bug taxonomy) Sistema de categorias (hierárquicas) concebido para facilitar a classificação dos defeitos.
- técnica baseada em defeitos:** (defect-based technique) Ver *técnica de concepção de testes baseada em defeitos*.
- ATT **técnica baseada na estrutura:** (structure-based technique) Ver *técnica caixa-branca de concepção de testes*.
- ATA **técnica baseada na experiência:** (experience-based technique) Ver *técnica de concepção de testes baseada na experiência*.
- ATA **técnica baseada nas especificações:** (specification-based technique) Ver *técnica caixa-preta de concepção de testes*.
- técnica caixa-branca:** (white-box technique) Ver *técnica caixa-branca de concepção de testes*.
- F **técnica caixa-branca de concepção de testes:** (white-box test design technique)
- ATT Procedimento para obter e/ou selecionar casos de teste baseado numa análise da estrutura interna de um componente ou sistema.
- técnica caixa-preta:** (black box technique) Ver *técnica caixa-preta de concepção de testes*.
- F **técnica caixa-preta de concepção de testes:** (black box test design technique) Procedimento
- ATA para obter e/ou selecionar casos de teste com base numa análise da especificação, funcional ou não funcional, de um componente ou sistema, sem fazer referência à sua estrutura interna.
- técnica de concepção de casos de teste:** (test case design technique) Ver *técnica de concepção de testes*.

- técnica de concepção de teste estrutural:** (structural test design technique) Ver *técnica caixa-branca de concepção de testes*.
- F **técnica de concepção de testes:** (test design technique) Procedimento utilizado para obter e/ou selecionar casos de teste.
- técnica de concepção de testes baseada em defeitos:** (defect-based test design technique) Procedimento para obter e/ou selecionar casos de teste direcionados a uma ou mais categorias de defeitos, com os testes a serem desenvolvidos a partir do que é conhecido sobre essas categorias específicas de defeitos. Ver também *taxonomia de defeitos*.
- técnica de concepção de testes baseada na estrutura:** (structure-based test design technique) Ver *técnica caixa-branca de concepção de testes*.
- F **técnica de concepção de testes baseada na experiência:** (experience-based test design  
ATA technique) Procedimento para derivar e/ou selecionar casos de teste baseados na experiência, conhecimento ou intuição do testador.
- técnica de concepção de testes baseada nas especificações:** (specification-based test design technique) Ver *técnica caixa-preta de concepção de testes*.
- técnica de concepção de testes funcionais:** (functional test design technique) Procedimento para obter e/ou selecionar casos de testes com base numa análise da especificação da funcionalidade de um componente ou sistema sem fazer referência à sua estrutura interna. Ver também *técnica caixa-preta de concepção de testes*.
- técnica de concepção de testes não funcionais:** (non-functional test design technique) Procedimento para obter e/ou selecionar os casos de teste para testes não funcionais. Baseia-se na análise da especificação de um componente ou sistema sem fazer referência à sua estrutura interna. Ver também *técnica caixa-preta de concepção de testes*.
- técnica de especificação de teste:** (test specification technique) Ver *técnica de concepção de testes*.
- técnica de execução de testes:** (test execution technique) Método utilizado para efetuar a execução de testes, quer manual, quer automática.
- técnica de teste:** (test technique) Ver *técnica de concepção de testes*.
- tempo médio entre falhas:** (Mean Time Between Failures) Média aritmética do tempo entre as falhas de um sistema. O MTBF faz tipicamente parte do modelo de aumento da fiabilidade e assume que o sistema após a falha é imediatamente corrigido como parte do processo de correção de defeitos. Ver também *modelo de aumento da fiabilidade*.
- tempo médio para reparar:** (Mean Time to Repair) Média aritmética do tempo que um sistema levará para recuperar após qualquer falha. Inclui tipicamente testes para validar que o defeito foi efetivamente resolvido.
- ATM **Test Maturity Model integration:** (TMMi) *Framework* de cinco níveis criada para a  
EITP melhoria do processo de teste, relacionada com o *Capability Maturity Model Integration* (CMMI), que descreve os principais elementos de um processo de teste eficaz.
- Test Point Analysis (TPA):** (Test Point Analysis (TPA)) Fórmula para cálculo da estimativa dos testes com base na análise de pontos de função. [Tmap]
- ATT **testabilidade:** (testability) A capacidade de um produto (de software) em permitir o teste ao software modificado. [ISO 9126] Ver também *manutenibilidade*.
- F **testador:** (tester) Profissional qualificado que está envolvido nos testes de um componente

ou sistema.

F **testar:** (testing) Processo que consiste em todas as atividades do ciclo de vida, tanto estáticas como dinâmicas, ligadas ao planeamento, preparação e avaliação de produtos de software e produtos de trabalho relacionados, para determinar a satisfação dos requisitos especificados, para demonstrar que se adequam ao objetivo e para detetar defeitos.

**teste:** (test) Conjunto de um ou mais casos de teste. [IEEE 829]

**teste ad hoc:** (ad hoc testing) Teste realizado informalmente, sem preparação formal, e sem recurso a nenhuma técnica de conceção de testes. Não há expectativas relativamente aos resultados e a arbitrariedade guia a atividade de execução do teste.

EITP **teste ágil:** (agile testing) Prática de teste para um projeto que utiliza metodologias ágeis incorporando técnicas e métodos, tais como eXtreme Programming (XP), e que trata o processo de desenvolvimento como cliente do processo de testes, enfatizando o paradigma “test-first design”. Ver também *desenvolvimento orientado a testes*.

**teste aleatório:** (random testing) Técnica caixa-preta de conceção de testes onde os casos de teste são selecionados, possivelmente por meio de um algoritmo de geração pseudo-aleatória, para responder a um perfil operacional. Esta técnica pode ser utilizada para testar atributos não funcionais, tais como fiabilidade e desempenho.

F **teste alfa:** (alpha testing) Teste operacional, simulado ou real, executado por utilizadores/clientes potenciais ou por uma equipa de teste independente no ambiente de desenvolvimento, mas fora da organização. O teste alfa é frequentemente utilizado para avaliar software de prateleira (“off-the-shelf”) servindo assim de teste de aceitação interno.

ETM **teste analítico:** (analytical test) Teste baseado numa análise sistemática de, por exemplo, os riscos do produto ou requisitos.

ATA **teste aos pares:** (pairwise testing) Técnica caixa-preta de conceção de testes onde os casos de teste são concebidos para executar todas as combinações discretas possíveis de cada par de valores de entrada (*inputs*). Ver também *teste matriz ortogonal*.

**teste ascendente:** (bottom-up testing) Abordagem incremental para o teste de integração, onde os componentes de nível mais baixo são testados primeiro, seguindo-se o teste de componentes de níveis superiores. Este processo é repetido até que o componente no topo da hierarquia seja testado. Ver também *teste de integração*.

ATT **teste às condições e decisões:** (decision condition testing) Técnica caixa-branca de conceção de testes onde são concebidos casos de teste para executar resultados de condições e decisões.

ATA **teste às *user story*:** (user story testing) Técnica de conceção de testes caixa-preta na qual os casos de teste são projetados com base em *user stories* para verificar a sua correcta aplicação. Ver também *user story*.

**teste baseado em ataques:** (attack-based testing) Técnica de teste baseada na experiência do testador que usa ataques de software para induzir falhas, principalmente falhas relacionadas à segurança. Ver também *ataque*.

**teste baseado em *checklists*:** (checklist-based testing) Técnica de conceção de testes baseada na experiência do testador em que ele utiliza uma lista de alto nível, de itens a serem observados, verificados ou lembrados ou um conjunto de regras ou critérios segundo as quais um produto deve ser verificado. Ver também *técnica baseada na experiência*.

**teste baseado em código:** (code-based testing) Ver *teste caixa-branca*.

ETM **teste baseado em modelos:** (model-based testing) Testes com base em modelos de componentes ou do sistema sob teste, por exemplo, modelos de aumento de fiabilidade, modelos de utilização, tais como perfis operacionais, ou modelos comportamentais como tabelas de decisão ou diagramas de transição de estado.

**teste baseado em processos de negócios:** (business process-based testing) Abordagem em que os casos de teste são concebidos com base nas descrições e/ou conhecimento dos processos de negócio.

**teste baseado em sessões:** (session-based testing) Abordagem de teste em que as atividades de teste são sessões contínuas de conceção e execução de testes, utilizada normalmente em conjunto com os testes exploratórios.

F  
ATA **teste baseado em tabelas de decisão:** (decision table testing) Técnica caixa-preta de conceção de testes que usa as tabelas de decisão para garantir que são executadas as combinações de entrada (inputs) e/ou estímulos (causas) aí representadas. [Veenendaal04]. Ver também *tabela de decisão*.

ATA **teste baseado em requisitos:** (requirements-based testing) Abordagem de teste onde os casos de testes são concebidos tendo por base os objetivos e as condições de teste derivadas dos requisitos, por ex: testes que exercitam funções específicas ou investigam atributos não funcionais, tais como, fiabilidade ou usabilidade.

**teste baseado na conceção:** (design based testing) Abordagem à execução de testes onde os casos de teste são concebidos de acordo com a arquitetura e/ou a conceção detalhada de um componente ou sistema (por ex: testes de interfaces entre componentes ou sistemas).

F **teste baseado na estrutura:** (structure-based testing) Ver *teste caixa-branca*.

**teste baseado na experiência:** (experience-based testing) Teste baseado na experiência, conhecimento e intuição do testador.

**teste baseado nas especificações:** (specification-based testing) Ver *teste caixa-preta*.

F **teste beta:** (beta testing) Testes operacionais por parte de utilizadores/clientes potenciais e/ou existentes num local externo e que não estão envolvidos com a equipa de desenvolvimento, de forma a determinar se um componente ou o sistema satisfaz, ou não, as necessidades do utilizador/cliente e se se encaixa dentro dos processos de negócio. Testes beta são muitas vezes utilizados como uma forma de teste de aceitação externo ao software de prateleira (off-the-shelf) a fim de adquirir a opinião do mercado.

**teste big-bang:** (big-bang testing) Uma forma de testes de integração no qual os elementos de software, elementos de hardware, ou ambos, são combinados de uma só vez num componente ou sistema global, em vez de serem combinados em etapas. [Depois de IEEE 610] Ver também *teste de integração*.

F **teste caixa-branca:** (white-box testing; glass box testing; clear-box testing) Testes baseados numa análise da estrutura interna do componente ou sistema.

F **teste caixa-preta:** (black box testing) Teste, funcional ou não funcional, sem referência à estrutura interna do componente ou sistema.

ATA **teste causa-efeito:** (cause-effect graphing) Técnica caixa-preta de conceção de testes em que os casos de teste são construídos a partir de grafos causa-efeito. [BS 7925/2]

**teste completo:** (complete testing) Ver *teste exaustivo*.

**teste com recurso a guiões:** (scripted testing) Execução de testes levada a cabo com recurso

a uma sequência de testes previamente documentada.

**ATA teste combinatório:** (combinatorial testing) Um meio para identificar um subconjunto de combinações de teste adequado para alcançar um determinado nível de cobertura, assumindo o teste a um objeto com múltiplos parâmetros, tendo cada parâmetro, de per si, múltiplos valores, que originam demasiadas combinações para ser viável testá-las no período de tempo disponível. Ver também *método de classificação em árvore*, *teste em pares*, *teste matriz ortogonal*.

**ETM teste consultivo:** (consultative testing) Testes conduzidos pelo aconselhamento e orientação de especialistas que são externos à equipa de teste (por exemplo, especialistas em tecnologia e / ou especialistas no domínio do negócio).

**teste da integridade da base de dados:** (database integrity testing) Teste aos métodos e processos utilizados para aceder e gerir os dados (base de dados), de forma a garantir que os métodos de acesso, processos e regras de manipulação de dados funcionam como expectável e que, no acesso à base de dados, os dados não são corrompidos, apagados, actualizados ou criados inesperadamente.

**teste da integridade de dados:** (data integrity testing) Ver *teste da integridade da base de dados*.

**teste de aceitação:** (acceptance testing) Teste formal, associado às necessidades do utilizador, requisitos e processos de negócio, realizado para determinar se existe ou não um sistema que satisfaz os critérios de aceitação e permite ao utilizador, cliente ou entidade autorizada, determinar se deve ou não aceitar o sistema. [Depois de IEEE 610]

**F teste de aceitação de utilizador:** (user acceptance testing) Ver *teste de aceitação*.

**ATT teste de aceitação operacional:** (operational acceptance testing) Testes operacionais durante a fase de testes de aceitação, tipicamente executados num ambiente operacional (simulado) pela equipa de operações e/ou administração de sistemas focados nos aspetos operacionais, por exemplo, recuperabilidade, consumo de recursos, instalabilidade e conformidade técnica. Ver também *teste operacional*.

**teste de aceitação da produção:** (production acceptance testing) Ver *teste de aceitação operacional*.

**ATA teste de acessibilidade:** (accessibility testing) Teste para determinar a facilidade com que utilizadores com deficiências/limitações conseguem utilizar um componente ou sistema. [Gerrard]

**teste de admissão:** (intake test) Instância especial do teste rápido (*smoke test*) que ajuda a decidir se o componente ou sistema está pronto para testes mais detalhados. Este tipo de teste é normalmente realizado no início da fase de execução de teste. Ver também *teste rápido*.

**teste de algoritmo:** (algorithm test) [TMap] Ver *teste de ramos*.

**teste de API:** (API testing) Teste ao código que permite a comunicação entre os diferentes processos, programas e/ou sistemas. O teste de API envolve frequentemente o teste negativo, por exemplo, para validar a robustez do tratamento de erros. Ver também *teste de interface*.

**teste de armazenamento:** (storage testing) Ver *teste de utilização de recursos*.

**teste de benchmark:** (benchmark test) (1) Norma relativa à qual podem ser feitas medidas e comparações. (2) Teste que é usado para comparar os componentes ou sistemas entre si ou

para comparar relativamente a uma norma conforme está definido em (1). [Depois de IEEE 610]

- ATT **teste de caminho:** (path testing) Técnica caixa-branca de concepção de testes na qual os casos de teste são concebidos para executar caminhos.
- F **teste de carga:** (load testing) Tipo de teste de desempenho que mede o comportamento de um componente ou sistema por meio do aumento de carga. Por exemplo, número de utilizadores simultâneos e/ou número de transações para determinar qual a carga que pode ser suportada pelo componente ou sistema. Ver também *teste de desempenho*, *teste de stress*.
- teste de conformidade com processo:** (process-compliant testing) Teste que segue um conjunto de processos definidos, por exemplo, definido por uma entidade externa, como um comitê normativo. Ver também *teste de conformidade com standards*.
- ETM **teste de conformidade com standards:** (standard-compliant testing) Testes que atendem a um conjunto de requisitos definidos por um standard, por exemplo, um standard industrial de testes ou um standard para testar sistemas críticos. Ver também *teste de conformidade com processo*.
- F **teste de casos de uso:** (use case testing) Técnica caixa-preta de concepção de testes na qual os  
ATA casos de teste são concebidos para executar cenários definidos por casos de uso.
- teste de cenário:** (scenario testing) Ver *teste de casos de uso*.
- teste de cenário de utilizador:** (user scenario testing) Ver *teste de casos de uso*.
- teste de ciclo de processo:** (process cycle test) Técnica caixa-preta de concepção de testes onde os casos de teste são concebidos para executar procedimentos e processos de negócio. [TMap] Ver também *teste de procedimento*.
- teste de cobertura lógica:** (logic-coverage testing) Ver *teste caixa-branca*. [Myers]
- teste de combinação de condições:** (condition combination testing) Ver *teste de múltiplas condições*.
- teste de combinações de condições de ramos:** (branch condition combination testing) Ver *teste de múltiplas condições*.
- teste de compatibilidade:** (compatibility testing) Ver *teste de interoperabilidade*.
- teste de comparação elementar:** (elementary comparison testing) Técnica caixa-preta de concepção de testes onde casos de teste são concebidos para executar combinações de valores de entrada (*inputs*) utilizando o conceito de “cobertura de determinação de condições”. [TMap]
- F **teste de componentes:** (component testing) Teste aos componentes individuais de software.  
[IEEE 610]
- teste de concorrência:** (concurrency testing) Testes efetuados para determinar como a ocorrência de duas ou mais atividades dentro do mesmo intervalo de tempo, conseguida por atividades intercaladas ou por execução simultânea, é tratada pelo componente ou sistema.  
[IEEE 610]
- ATT **teste de condições** (condition testing) Técnica caixa-branca de concepção de testes na qual os casos de teste são concebidos para executar os resultados de condições.
- teste de configuração:** (configuration testing) Ver *teste de portabilidade*.

- F **teste de confirmação:** (confirmation testing) Ver *re-teste*.
- teste de conformidade:** (compliance testing; conformance testing; standards testing)  
Processo de testes para determinar a conformidade do componente ou do sistema.
- teste de conversão:** (conversion testing) Testes de software realizados aquando da conversão de dados dos sistemas existentes para serem usados nos sistemas de substituição.
- teste de decisão:** (decision testing) Técnica caixa-branca de conceção de testes onde casos de teste são concebidos para executar os resultados de decisões.
- teste de determinação de condições:** (condition determination testing) Ver *teste modificado de condições e decisões*.
- F **teste de desempenho:** (performance testing) Processo de testes que determina o  
ATT desempenho de um produto de software. Ver também *teste de eficiência*.
- teste de desenvolvimento:** (development testing) Teste formal ou informal executado pelos programadores durante a implementação de um componente ou sistema, geralmente no ambiente de desenvolvimento. [IEEE 610]
- teste de disponibilidade:** (serviceability testing) Ver *teste de manutenibilidade*.
- teste de documentação:** (documentation testing) Teste à qualidade da documentação. Por exemplo, teste ao manual de utilização ou ao manual de instalação.
- teste de eficiência:** (efficiency testing) Processo de teste para determinar a eficiência de um produto de software.
- teste de escalabilidade:** (scalability testing) Tipo de teste que determina a escalabilidade de um produto de software.
- teste de estados finito:** (finite state testing) Ver *teste de transição de estados*.
- ATA **teste de exatidão:** (accuracy testing) Processo de teste para determinar a exatidão de um produto de software.
- F **teste de fiabilidade:** (reliability testing) Processo de teste que determina a fiabilidade de um  
ATT produto de software.
- ATT **teste de fluxo de controlo:** (control flow testing) Abordagem para o teste baseado em estrutura na qual os casos de teste são projetados para executar as sequências específicas de eventos. Existem várias técnicas para teste de fluxo de controlo, por exemplo, o teste de decisão, teste de condições e o teste de caminho, onde cada uma tem a sua abordagem específica e o seu nível de cobertura do fluxo de controlo. Ver também *teste de decisão*, *teste de condições*, *teste de caminho*.
- teste de fumo:** (smoke test) Ver *teste rápido*.
- teste de funcionalidade:** (functionality testing) Processo de teste realizado para determinar a funcionalidade de um produto de software. **teste de funcionalidade:** (functionality testing) Processo de teste realizado para determinar a funcionalidade de um produto de software.
- teste de instalabilidade:** (installability testing) Processo que testa a facilidade de instalação de um produto de software. Ver também *teste de portabilidade*.
- ATT **teste de instrução:** (statement testing) Técnica caixa-branca de conceção de testes em que os casos de teste são concebidos de forma a executar instruções.
- F **teste de integração:** (integration testing) Testes realizados com a finalidade de expor

defeitos nas interfaces e nas interações entre componentes ou sistemas integrados. Ver também *teste de integração de componentes*, *teste de integração de sistemas*.

- ATT **teste de integração aos pares:** (pairwise integration testing) Uma forma de teste de integração que “ataca” pares de componentes de software que executam em conjunto de acordo com um grafo de chamadas.
- teste de integração de componentes:** (component integration testing; link testing) Testes realizados com o intuito de encontrar defeitos nas interfaces e interações entre os componentes integrados.
- teste de integração de sistemas:** (system integration testing) Testes de integração de sistemas e pacotes de software; testes das interfaces com entidades externas. (por ex: Electronic Data Interchange (EDI), Internet).
- teste de integração em larga escala:** (integration testing in the large) Ver *teste de integração de sistemas*.
- teste de integração em pequena escala:** (integration testing in the small) Ver *teste de integração de componentes*.
- teste de integração hardware-software:** (hardware-software integration testing) Ensaaios realizados para expor defeitos nas interfaces e interação entre os componentes de hardware e software. Ver também *teste de integração*.
- ATT **teste de integração por vizinhos:** (neighborhood integration testing) Uma forma de teste de integração onde todos os nós que ligam a um dado nó são a base para o ensaio de integração.
- teste de interface:** (interface testing) Tipo de teste de integração que testa as interfaces entre componentes ou sistemas.
- F **teste de interoperabilidade:** (interoperability testing) Processo que determina a  
ATA interoperabilidade de um produto de software. Ver também *teste de funcionalidade*.
- teste de isolamento:** (isolation testing) Testes a componentes individuais isolando-os dos outros componentes. Se houver necessidade, pode-se simular os outros componentes com simuladores (*stubs*) e controladores (*drivers*).
- F **teste de manutenção:** (maintenance testing) Testa as alterações feitas a um sistema operacional ou o impacto de um ambiente alterado num sistema operacional.
- F **teste de manutenibilidade:** (maintainability testing) Processo de teste que determina a  
ATT manutenibilidade de um produto de software.
- teste de migração:** (migration testing) Ver *teste de conversão*.
- teste de módulo:** (module testing) Ver *teste de componentes*.
- teste de mutação:** (mutation testing) Ver *teste paralelo*.
- ATT **teste de múltiplas condições:** (multiple condition testing) Técnica caixa-branca de concepção de testes onde os casos de testes são concebidos para executar combinações de resultados das condições individuais (dentro de uma instrução).
- teste de partição:** (partition testing) Ver *particionar por equivalências*. [Beizer]
- teste de perfil operacional:** (operational profile testing) Teste estatístico que utiliza um modelo de operações de sistema (para tarefas de curta duração) e de probabilidade de utilização mais comum. [Musa]

- F **teste de portabilidade:** (portability testing) Processo de teste que determina a portabilidade  
ATT de um produto de software.
- teste de procedimento:** (procedure testing) Teste orientado para garantir que o componente ou sistema pode operar conjuntamente com outros procedimentos de negócio ou operacionais, sejam eles novos ou já existentes.
- teste de programa:** (program testing) Ver *teste de componentes*.
- teste de ramos:** (branch testing; arc testing) Técnica caixa-branca de concepção de testes em que os casos de teste são concebidos para executar os vários ramos de um fluxo.
- ATT **teste de recuperabilidade:** (recoverability testing) Processo de teste que determina a recuperabilidade de um produto de software. Ver também *teste de fiabilidade*.
- teste de recuperação:** (recovery testing) Ver *teste de recuperabilidade*.
- F **teste de regressão:** (regression testing) Teste realizado após modificações a um programa não previamente testado, com a finalidade de assegurar que não ficaram a descoberto defeitos antigos ou que não foram introduzidos novos defeitos em áreas de software não alteradas pelas modificações referidas. Este teste é realizado quando o software ou o seu ambiente de execução são alterados.
- teste de regulamentação:** (regulation testing) Ver *teste de conformidade*.
- F **teste de robustez:** (robustness testing) Teste que determina a robustez de um produto de software
- F **testes de segurança:** (safety testing, security testing) Teste para determinar a segurança de  
ATT um produto de software. Ver também teste de funcionalidade.
- teste de sequência linear de código e salto:** (LCSAJ testing) Técnica caixa-branca de concepção de testes onde os casos de teste são concebidos para executar LCSAJs.
- F **teste de sistema:** (system testing) Processo de testar um sistema integrado de modo a verificar o cumprimento dos requisitos especificados. [Hetzl]
- F **teste de stress:** (stress testing) Tipo de teste de desempenho conduzido para avaliar um sistema ou componente no (ou além do) limite da carga esperada ou especificada, ou conduzido com reduzida disponibilidade de recursos, tais como, recursos de memória ou servidores. [IEEE 610]. Ver também *teste de desempenho*, *teste de carga*.
- F **teste de transição de estados:** (state transition testing) Técnica caixa-preta de concepção de  
ATA testes cujos casos de teste são concebidos para executar transições de estados válidas e inválidas. Ver também *teste N-switch*.
- F **teste de usabilidade:** (usability testing) Teste que determina em que medida o produto de  
ATA software é compreendido, fácil de aprender, fácil de utilizar e atraente para os utilizadores sob determinadas condições especificadas. [Depois de ISO 9126]
- ATT **teste de utilização de recursos:** (resource utilization testing) Processo de teste para determinar a utilização de um determinado recurso por um produto de software. Ver também *teste de eficiência*.
- teste de utilizador:** (user test) Teste onde utilizadores reais são envolvidos na avaliação da usabilidade de um componente ou sistema.
- teste de valor fronteira:** (boundary value testing) Ver *análise de valor fronteira*.
- teste de volume:** (volume testing) Teste onde o sistema é sujeito a grandes volumes de

dados. Ver também *teste de utilização de recursos*.

F **teste dinâmico:** (dynamic testing) Teste que implica a execução do software de um componente ou sistema.

**teste dirigido pela lógica:** (logic-driven testing) Ver *teste caixa-branca*.

**teste do fluxo de dados:** (data flow testing) Técnica caixa-branca de concepção de testes onde os casos de teste são concebidos para executar pares definição-uso de variáveis.

**teste do macaco:** (monkey testing) Teste efetuado com recurso a uma seleção aleatória a partir de um grande conjunto de dados de entrada (*input*) e pelo premir aleatório de botões, ignorando a forma como o produto está a ser usado.

F **teste em campo:** (field testing) Ver *teste beta*.

**teste em pares:** (pair testing) Teste realizado por duas pessoas, por exemplo, dois testadores, um programador e um testador, ou um utilizador final e um testador, que trabalham juntos para descobrir defeitos. De uma forma geral, eles partilham o mesmo computador e alternam no controlo durante o teste.

F **teste estático:** (static testing) Testes a um artefacto de desenvolvimento de software, por exemplo, requisitos, *design* ou código, sem a execução desses artefactos. Por exemplo, revisões ou análise estática.

**teste estatístico:** (statistical testing) Técnica de concepção de testes na qual é utilizado um modelo de distribuição estatística das entradas para construir casos de teste representativos. Ver também *teste de perfil operacional*.

F **teste estrutural:** (structural testing) Ver *teste caixa-branca*.

F **teste exaustivo:** (exhaustive testing) Abordagem de teste onde a bateria de testes inclui todas as combinações de valores de entrada (inputs) e as pré-condições.

**teste falhado:** (test fail) Ver *sem sucesso*.

F **teste funcional:** (functional testing) Teste baseado na análise da especificação da funcionalidade de um componente ou sistema. Ver também *teste caixa-preta*.

**teste incremental:** (incremental testing) Teste no qual os componentes ou sistemas são integrados e testados sozinhos, ou em pequenos grupos de cada vez, até que todos os componentes ou sistemas estejam integrados e testados.

**teste inválido:** (invalid testing) Testes onde são utilizados valores de entrada (*input*) que devem ser rejeitados pelo componente ou sistema. Ver também *tolerância ao erro, teste negativo*.

**teste matriz ortogonal:** (orthogonal array testing) Método de teste sistemático que executa todas as combinações de pares de variáveis com recurso a matrizes ortogonais. Reduz significativamente o número das combinações para testar todas as combinações 2 a 2. Ver também *teste aos pares*.

**teste metódico:** (methodical testing) Teste baseado num conjunto de testes standard, por exemplo, numa lista de verificação (checklist), num standard de qualidade, ou num conjunto de casos de teste genéricos.

**teste modificado de condições e decisões:** (modified condition decision testing) Técnica de concepção de testes caixa branca em que os casos de teste são concebidos para executar os resultados de condições individuais que afetam, de forma independente, o resultado da

decisão.

**teste modificado de múltiplas condições:** (modified multiple condition testing) Ver *teste modificado de condições e decisões*.

**teste não funcional:** (non-functional testing) Testa os atributos de um componente ou sistema que não se relacionam com a sua funcionalidade. Por exemplo, fiabilidade, eficiência, usabilidade, manutenibilidade e portabilidade.

**teste negativo:** (negative testing) Tipo de teste que visa mostrar que um componente ou sistema não funciona. O teste negativo refere-se mais à atitude do testador do que a uma abordagem de teste específica ou a uma técnica de conceção de testes. Por exemplo, o teste com valores de entrada inválidos ou com exceções. [Beizer]

**teste N-switch:** (n-switch testing) Uma forma de teste de transição de estados onde os casos de teste são concebidos para executar todas as sequências de n+1 transições válidas. [Chow]. Ver também *teste de transição de estados*.

**teste operacional:** (operational testing) Testes realizados com a finalidade de avaliar um componente ou sistema no seu ambiente operacional. [IEEE 610]

**teste orientado a dados:** (data-driven testing) Técnica de desenvolvimento de *scripts* onde os valores de entrada e os resultados esperados são guardados numa tabela ou folha de cálculo, para que um único *script* de controlo possa executar todos os testes da tabela. O teste orientado a dados é uma técnica geralmente utilizada como suporte à aplicação de ferramentas de execução de testes como as de captura/reprodução [Fewster and Graham]. Ver também *teste orientado a palavras-chave*.

**teste orientado a palavras-ação:** (action word driven testing) Ver *teste orientado a palavras-chave*.

F  
ATA  
ATT **teste orientado a palavras-chave:** (keyword-driven testing) Técnica de desenvolvimento de *scripts* que utiliza arquivos de dados para conter, não apenas os dados de teste e seus resultados esperados, mas também, as palavras-chave relacionadas com a aplicação sob teste. As palavras-chave são interpretadas por *scripts* especiais de suporte que por, sua vez, são chamados pelo *script* de controlo do teste. Ver também *teste orientado a dados*.

**teste paralelo:** (back-to-back testing) Testes em que duas ou mais variantes de um componente ou sistema são executados com as mesmas entradas (*inputs*). Em caso de discrepâncias, os valores de saída (*outputs*) são comparados e analisados. [IEEE 610]

**teste rápido:** (smoke test; confidence test; sanity test) Subconjunto de todos os casos de teste definidos/planeados que cobrem as principais funcionalidades de um componente ou sistema, de forma a confirmar se essas funcionalidades mais críticas estão a funcionar, sem preocupação com os detalhes. A entrega diária (*daily build*) e os testes rápidos estão entre as melhores práticas da indústria. Ver também *teste de admissão*.

ETM **teste reativo:** (reactive testing) Testes que respondem dinamicamente ao sistema alvo de teste e aos resultados que vão sendo obtidos. Tipicamente os testes reativos têm um período reduzido de planeamento e as fases de conceção e implementação de testes não são realizadas enquanto o objeto de teste não é recebido.

**teste por terceiros:** (outsourced testing) Testes realizados por pessoas que não estão co-localizadas com a equipa do projeto e não são colegas de trabalho.

**teste sujo:** (dirty testing) Ver *teste negativo*.

**teste unitário:** (unit testing) Ver *teste de componentes*.

**teste de *thread*:** (thread testing) Versão de testes de integração de componentes, onde a integração progressiva destes segue a implementação de subconjuntos de requisitos, em oposição à integração de componentes por níveis de hierarquia.

**teste sobre falhas:** (failover testing) Teste realizado pela simulação de modos de falha ou pelo provocar de falhas reais num ambiente controlado. Após uma falha, o mecanismo de recuperação (*failover*) é testado para assegurar que os dados não são perdidos ou corrompidos e que quaisquer níveis de serviço acordados estão assegurados (por exemplo, disponibilidade funcional ou cumprimento de tempos de resposta). Ver também *teste de recuperabilidade*.

ETM **testes adversos à regressão:** (regression-averse testing) Testes que utilizam várias técnicas para gestão do risco da regressão, por exemplo, pela conceção de *testware* reutilizável e pela automação extensiva de testes num ou em mais níveis de teste.

F **testes baseados na avaliação do risco:** (risk-based testing) Abordagem de testes com o objetivo de reduzir o nível de risco do produto e de informar as partes interessadas do seu estado, e que começa nas primeiras etapas de um projeto. Envolve a identificação dos riscos do produto e a sua classificação por níveis de risco de modo a dirigir o processo de testes.

ATM  
ATA  
ATT

**testes de aceitação de fábrica:** (factory acceptance testing) Teste de aceitação realizado no local em que o produto é desenvolvido e executado pelos funcionários do fabricante, para determinar se um componente ou sistema satisfaz ou não os requisitos, e inclui normalmente hardware e software. Ver também *testes alfa*.

**testes de aceitação locais:** (site acceptance testing) Testes de aceitação feitos por utilizadores/clientes nas suas instalações para determinar se um componente ou sistema satisfaz as necessidades desse utilizador/cliente e se se enquadra dentro dos processos de negócio que incluem, normalmente, hardware e software.

ATA **testes de adequabilidade:** (suitability testing) Processo de teste para determinar a adequação de um produto de software.

**testes de sintaxe:** (syntax testing) Técnica caixa-preta de conceção de testes na qual os casos de teste são concebidos com base na definição de um domínio de entrada e/ou de saída.

**testes descendentes:** (top-down testing) Abordagem incremental aos testes de integração, onde o componente no topo da hierarquia é testado em primeiro lugar, com simuladores dos componentes de baixo nível. Os componentes testados são então utilizados para testar os componentes de níveis mais baixos. O processo é repetido até que os componentes de nível mais baixo estejam testados. Ver também *teste de integração*.

F **testes exploratórios:** (exploratory testing) Técnica informal de conceção de testes onde o testador (tester) controla ativamente a conceção dos testes e utiliza a informação adquirida durante a execução dos testes, para conceber novos e melhores testes. [Bach]

ATA

**testes na fonte:** (insourced testing) Testes realizados por pessoas que estão co-localizadas com a equipa do projeto, mas não são colegas de trabalho.

F **testware:** (testware) Artefactos produzidos durante o processo de teste, necessários para planear, conceber e executar testes, tais como documentação, guiões (*scripts*), entradas, resultados esperados, procedimentos de preparação e limpeza, ficheiros, bases de dados, ambiente, e qualquer software adicional ou utilitários utilizados nos testes. [Depois de Fewster e Graham]

**testware automatizado:** (automated testware) Todos os artefactos utilizados em testes

automatizados, tais como os guiões (*scripts*) de uma ferramenta.

**tipo de defeito:** (defect type) Elemento de uma taxonomia de defeitos. As taxonomias de defeitos podem ser identificadas com respeito a uma variedade de considerações, incluindo, mas não limitadas a:

- Fase ou atividade de desenvolvimento em que o defeito é criado, por exemplo, um erro de especificação ou um erro de codificação
- Caracterização de defeitos, por exemplo, um defeito "off-by-one"
- Incorreções, por exemplo, um operador relacional incorreto, um erro de sintaxe da linguagem de programação, ou uma suposição inválida
- Problemas de desempenho, por exemplo, o tempo de execução excessivo, disponibilidade insuficiente.

**tipo de teste:** (test type) Conjunto de atividades de teste cujo objectivo é testar um componente ou sistema, focando um objetivo de teste específico, ou seja, teste funcional, teste de usabilidade, teste de regressão, etc. Um tipo de teste pode ocorrer num ou mais níveis ou fases de teste. [TMap]

**tipo do risco:** (risk type) Conjunto de riscos agrupados por um ou mais fatores comuns, tais como, atributos de qualidade, causa, localização ou efeito potencial do risco. Um conjunto específico de tipos de risco de um produto está relacionado com o tipo de testes que podem mitigar (manter controlados) esses tipos de risco. Por exemplo, o risco do utilizador não interpretar bem o software pode ser mitigado com testes de usabilidade.

ATM **TMMi:** (TMMi) Ver *Test Maturity Model integration*.  
EITP

**tolerância a falhas:** (fault tolerance) Capacidade que um produto de software tem para manter um nível específico de desempenho em casos de defeitos de software ou violação da sua interface específica. [ISO 9126]. Ver também *fiabilidade, robustez*.  
[N. do T.] Embora a tradução mais correta seja tolerância a defeitos, optou-se por manter o termo mais usual.

**tolerância ao erro:** (error tolerance) Capacidade de um sistema ou componente para continuar a operar normalmente apesar da presença de valores de entrada incorretos (*inputs*). [Depois de IEEE 610]

EITP **TPG:** (TPG) Ver *Grupo do Processo de Teste*.

ATM **TPI Next:** (TPI Next) Framework orientada ao negócio para melhoria contínua do processo de teste que descreve os elementos-chave de um processo de teste eficaz e eficiente.

EITP **TQM:** (TQM) Ver *Gestão de Qualidade Total*.

**transição de estados:** (state transition) Transição entre dois estados de um componente ou sistema.

**tratamento de exceções:** (exception handling) Comportamento de um componente ou sistema em resposta a um valor de entrada (*input*) incorreto, introduzido por um utilizador, por outro componente ou sistema, ou devido a uma falha interna.

## U

**unidade:** (unit) Ver *componente*.

**usabilidade:** (usability) Capacidade do software ser compreendido, aprendido, utilizado e atrativo para o utilizador quando usado sob as condições especificadas. [ISO 9126]

**user story:** (user story) Requisito de utilizador ou de negócio de alto nível comumente usado em desenvolvimento ágil de software, geralmente constituído por uma ou mais frases do quotidiano ou da linguagem dos negócios que captura a funcionalidade que um utilizador precisa, critérios não-funcionais, e também critérios de aceitação. Ver também *desenvolvimento ágil de software, requisito*.

**utilização de recursos:** (resource utilization) Capacidade que um produto de software tem de utilizar volumes e tipos de recursos apropriados, por ex: volume de memória, principal e secundária, utilizada pelo programa e o tamanho de ficheiros temporários necessários quando o software desempenha a sua função sob dadas condições. [Depois de ISO 9126]. Ver também *eficiência*.

## V

F **validação:** (validation) Confirmação, pela análise e recolha de evidências objetivas, de que os requisitos para uma utilização ou aplicação específica foram cumpridos. [ISO 9000]

**valor de entrada:** (input value) Instância de uma entrada (*input*). Ver também *entrada*.

**valor de saída:** (output value) Instância de uma saída (*output*). Ver também *saída*.

**valor fronteira:** (boundary value) Valor de entrada ou valor de saída que está na fronteira de uma partição de equivalência ou na menor distância incremental de cada lado dessa fronteira. Por exemplo, o valor mínimo ou máximo de um intervalo.

**variável:** (variable) Elemento de armazenamento num computador que é acessível por um programa de software referindo-se a ela por um nome.

F **verificação:** (verification) Confirmação, pela análise e recolha de evidências objetivas, de que os requisitos especificados foram cumpridos. [ISO 9000]

**verificação de secretária:** (desk checking) Teste de software ou de especificações pela simulação manual da sua execução. Ver também *teste estático*.

## W

F **walkthrough:** (walkthrough) Apresentação passo a passo feita pelo autor de um documento  
ATM de forma a recolher informação e estabelecer um entendimento comum do seu conteúdo. [Freedman and Weinberg, IEEE 1028] Ver também *revisão por pares*.

**WAMMI:** (WAMMI) Ver *Website Analysis and Measurement Inventory*.

ATA **Website Analysis and Measurement Inventory (WAMMI):** Técnica de teste de usabilidade baseada em questionários para medir a qualidade do software de um site web do ponto de vista do utilizador final.

**WBS:** (WBS) Ver *Work Breakdown Structure*.

ATM **Wide Band Delphi:** (Wide Band Delphi) Técnica de estimativa de testes baseada em peritos que visa fazer uma estimativa precisa com base no conhecimento (sabedoria) coletivo dos membros da equipa.

**Work Breakdown Structure:** (Work Breakdown Structure) Organização dos elementos de trabalho e da sua inter-relação com o produto final. [CMMI]

## **Anexo A (Referências)**

### **Standards:**

- [DO-178b] DO-178B:1992. Software Considerations in Airborne Systems and Equipment Certification, Requirements and Technical Concepts for Aviation (RTCA SC167).
- [IEEE 610] IEEE 610.12:1990. Standard Glossary of Software Engineering Terminology.
- [IEEE 829] IEEE 829:1998. Standard for Software Test Documentation.
- [IEEE 1008] IEEE 1008:1993. Standard for Software Unit Testing.
- [IEEE 1028] IEEE 1028:1997. Standard for Software Reviews and Audits.
- [IEEE 1044] IEEE 1044:1993. Standard Classification for Software Anomalies.
- [IEEE 1219] IEEE 1219:1998. Software Maintenance.
- [ISO 2382/1] ISO/IEC 2382-1:1993. Data processing - Vocabulary - Part 1: Fundamental terms.
- [ISO 9000] ISO 9000:2005. Quality Management Systems – Fundamentals and Vocabulary.
- [ISO 9126] ISO/IEC 9126-1:2001. Software Engineering – Software Product Quality – Part 1: Quality characteristics and sub-characteristics.
- [ISO 12207] ISO/IEC 12207:1995. Information Technology – Software Lifecycle Processes.
- [ISO 14598] ISO/IEC 14598-1:1999. Information Technology – Software Product Evaluation - Part 1: General Overview.
- [ISO 15504] ISO/IEC 15504-9: 1998. Information Technology – Software Process Assessment – Part 9: Vocabulary

### **Livros e artigos:**

- [Abbott] J. Abbot (1986), Software Testing Techniques, NCC Publications.
- [Adrion] W. Adrion, M. Branstad and J. Cherniabsky (1982), Validation, Verification and Testing of Computer Software, in: Computing Surveys, Vol. 14, No 2, June 1982.
- [Bach] J. Bach (2004), Exploratory Testing, in: E. van Veenendaal, The Testing Practitioner – 2nd edition, UTN Publishing, ISBN 90-72194-65-9.
- [Beizer] B. Beizer (1990), Software Testing Techniques, van Nostrand Reinhold, ISBN 0-442-20672-0
- [Chow] T. Chow (1978), Testing Software Design Modelled by Finite-Sate Machines, in: IEEE Transactions on Software Engineering, Vol. 4, No 3, May 1978.
- [CMM] M. Paulk, C. Weber, B. Curtis and M.B. Chrissis (1995), The Capability Maturity Model, Guidelines for Improving the Software Process, Addison-Wesley, ISBN 0-201-54664-7
- [CMMI] M.B. Chrissis, M. Konrad and S. Shrum (2004), CMMI, Guidelines for Process Integration and Product Improvement, Addison Wesley, ISBN 0-321-15496-7
- [Deming] D. W. Edwards (1986), Out of the Crisis, MIT Center for Advanced Engineering Study, ISBN 0-911379-01-052

- [Fenton] N. Fenton (1991), *Software Metrics: a Rigorous Approach*, Chapman & Hall, ISBN 0-53249-425-1
- [Fewster and Graham] M. Fewster and D. Graham (1999), *Software Test Automation, Effective use of test execution tools*, Addison-Wesley, ISBN 0-201-33140-3.
- [Freedman and Weinberg] D. Freedman and G. Weinberg (1990), *Walkthroughs, Inspections, and Technical Reviews*, Dorset House Publishing, ISBN 0-932633-19-6.
- [Garvin] D.A. Garvin (1984), What does product quality really mean?, in: *Sloan Management Review*, Vol. 26, nr. 1 1984
- [Gerrard] P. Gerrard and N. Thompson (2002), *Risk-Based E-Business Testing*, Artech House Publishers, ISBN 1-58053-314-0.
- [Gilb and Graham] T. Gilb and D. Graham (1993), *Software Inspection*, Addison-Wesley, ISBN 0-201-63181-4.
- [Graham] D. Graham, E. van Veenendaal, I. Evans and R. Black (2007), *Foundations of Software Testing*, Thomson Learning, ISBN 978-1-84480-355-2
- [Grochtmann] M. Grochtmann (1994), *Test Case Design Using Classification Trees*, in: *Conference Proceedings STAR 1994*.
- [Hetzel] W. Hetzel (1988), *The complete guide to software testing – 2nd edition*, QED Information Sciences, ISBN 0-89435-242-3.
- [Juran] J.M. Juran (1979), *Quality Control Handbook*, McGraw-Hill
- [McCabe] T. McCabe (1976), A complexity measure, in: *IEEE Transactions on Software Engineering*, Vol. 2, pp. 308-320.
- [Musa] J. Musa (1998), *Software Reliability Engineering Testing*, McGraw-Hill Education, ISBN 0-07913-271-5.
- [Myers] G. Myers (1979), *The Art of Software Testing*, Wiley, ISBN 0-471-04328-1.
- [TMap] M. Pol, R. Teunissen, E. van Veenendaal (2002), *Software Testing, A guide to the TMap Approach*, Addison Wesley, ISBN 0-201-745712.
- [TMMi] E. van Veenendaal and J. Cannegieter (2011), *The Little TMMi*, UTN Publishing, ISBN 97-89490986-03-2
- [Veenendaal04] E. van Veenendaal (2004), *The Testing Practitioner – 2nd edition*, UTN Publishing, ISBN 90-72194-65-9.
- [Veenendaal08] E. van Veenendaal (2008), *Test Improvement Manifesto*, in: *Testing Experience*, Issue 04/08, December 2008

## **Anexo B (Método para comentar este glossário)**

Todos os comentários ao presente glossário são bem-vindos, para que este possa ser melhorado, indo ao encontro das necessidades da comunidade de testes de software.

Quando fizer um comentário, assegure-se de que inclui a seguinte informação:

- Nome e detalhes de contacto;
- A versão do glossário (presentemente 2.2pt);
- Zona exata do glossário;
- Para novas entradas do glossário, inclua igualmente uma referência para o Syllabus ISTQB que usa o termo;
- Informação de suporte, tais como a razão para a alteração proposta, ou uma referência para o uso de um termo.

Pode submeter comentários nas seguintes formas, apresentadas por ordem de preferência:

1. Por E-mail para [glossario@pstqb.pt](mailto:glossario@pstqb.pt);
2. Por correio para PSTQB – FEUP, Rua Dr. Roberto Frias, 4200-465 PORTO, PORTUGAL;
3. Por FAX para +351.22.557.41.03, dirigido a PSTQB.

## Anexo C (Tradução dos termos em inglês)

Para facilitar a pesquisa do termo em português a partir do termo em inglês, apresenta-se neste anexo a lista de termos em inglês e a correspondente tradução.

Inglês	Português
<b>A</b>	
abstract test case	caso de teste abstrato
acceptance	aceitação
acceptance criteria	critério de aceitação
acceptance testing	teste de aceitação
accessibility testing	teste de acessibilidade
accuracy	exatidão
accuracy testing	teste de exatidão
acting (IDEAL)	ação (IDEAL)
action word driven testing	teste orientado a palavras-ação
actor	ator
actual outcome	resultado obtido
actual result	resultado obtido
ad hoc review	revisão ad hoc
ad hoc testing	teste ad hoc
adaptability	adaptabilidade
agile manifesto	manifesto ágil
agile software development	desenvolvimento ágil de software
agile testing	teste ágil
algorithm test	teste de algoritmo
alpha testing	teste alfa
analytical testing	teste analítico
analyzability	analísabilidade
analyzer	analisador
anomaly	anomalia
anti-pattern	anti-padrão
API testing	teste de API
arc testing	teste de ramos
assessment report	relatório de avaliação
assessor	avaliador
atomic condition	condição atômica
attack	ataque
attack-based testing	teste baseado em ataques
attractiveness	atratividade
audit	auditoria
audit trail	caminho de auditoria
automated testware	<i>testware</i> automatizado
availability	disponibilidade
<b>B</b>	
A	B
back-to-back testing	teste paralelo
balanced scorecard	<i>balanced scorecard</i>
baseline	<i>baseline</i>

<b>Inglês</b>	<b>Português</b>
basic block	bloco básico
basis test set	bateria base de testes
bebugging	injeção de falhas
behavior	comportamento
benchmark test	teste de <i>benchmark</i>
bespoke software	software à medida
best practice	melhores práticas
beta testing	teste beta
big-bang testing	teste <i>big-bang</i>
black box technique	técnica caixa-preta
black box test design technique	técnica caixa-preta de concepção de testes
black box testing	teste caixa-preta
blocked test case	caso de teste bloqueado
bottom-up testing	teste ascendente
boundary value	valor fronteira
boundary value analysis	análise de valor fronteira
boundary value coverage	cobertura de valores fronteira
boundary value testing	teste de valor fronteira
branch	ramo
branch condition	condição de ramo
branch condition combination coverage	cobertura de combinações de condições de ramos
branch condition combination testing	teste de combinações de condições de ramos
branch condition coverage	cobertura de condições de ramo
branch coverage	cobertura de ramos
branch testing	teste de ramos
buffer	<i>buffer</i>
buffer overflow	<i>buffer overflow</i>
bug	defeito
bug report	relatório de defeitos
bug taxonomy	taxonomia de defeitos
bug tracking tool	ferramenta de controlo de defeitos
<b>C</b>	
call graph	gráfico de invocações
CMM	Capability Maturity Model (CMM)
CMMI	Capability Maturity Model Integration (CMMI)
capture/playback tool	ferramenta de captura/reprodução
capture/replay tool	ferramenta de captura/reprodução
CASE	CASE
CAST	CAST
causal analysis	análise causal
cause-effect analysis	análise causa-efeito
cause-effect decision table	tabela de decisão causa-efeito
cause-effect diagram	diagrama causa-efeito
cause-effect graph	grafo causa-efeito
cause-effect graphing	teste causa-efeito
certification	certificação
change control	controlo de alterações

<b>Inglês</b>	<b>Português</b>
change control board	comité de controlo de alterações
change management	(1) gestão da mudança (2) gestão de alterações
changeability	mutabilidade
charter	ficha de testes
checker	revisor
checklist-based testing	teste baseado em <i>checklists</i>
Chow's coverage metrics	métricas de cobertura de Chow
classification tree	árvore de classificação
classification tree method	método de classificação em árvore
clear-box testing	teste caixa-branca
CMMI	CMMI
code	código
code analyzer	analisador de código
code coverage	cobertura de código
code-based testing	teste baseado em código
codependent behavior	comportamento codependente
co-existence	co-existência
combinatorial testing	teste combinatório
commercial off-the-shelf software	software comercial de prateleira
comparator	comparador
compatibility testing	teste de compatibilidade
compiler	compilador
complete testing	teste completo
completion criteria	critério de conclusão
complexity	complexidade
compliance	conformidade
compliance testing	teste de conformidade
component	componente
component integration testing	teste de integração de componentes
component specification	especificação de componentes
component testing	teste de componentes
compound condition	condição composta
concrete test case	caso de teste concreto
concurrency testing	teste de concorrência
condition	condição
condition combination coverage	cobertura de combinação de condições
condition combination testing	teste de combinação de condições
condition coverage	cobertura de condições
condition determination coverage	cobertura de determinação de condições
condition determination testing	teste de determinação de condições
condition outcome	resultado de condição
condition testing	teste de condições
confidence interval	intervalo de confiança
confidence test	teste rápido
configuration	configuração
configuration auditing	auditoria de configuração
configuration control	controlo de configuração
configuration control board	comité de controlo de configurações (CCB)
configuration identification	identificação da configuração
configuration item	elemento de configuração
configuration management	gestão de configurações

<b>Inglês</b>	<b>Português</b>
configuration management tool	ferramenta de gestão de configurações
configuration testing	teste de configuração
confirmation testing	teste de confirmação
conformance testing	teste de conformidade
consistency	consistência
consultative testing	teste consultivo
content-based model	modelo baseado em conteúdo
continuous representation	representação contínua
control chart	gráfico de controle
control flow	fluxo de controle
control flow analysis	análise de fluxo de controle
control flow graph	grafo de fluxo de controle
control flow path	caminho no fluxo de controle
control flow testing	teste de fluxo de controle
convergence metric	métrica de convergência
conversion testing	teste de conversão
corporate dashboard	painel de instrumentos corporativo
cost of quality	custo da qualidade
COTS	COTS
coverage	cobertura
coverage analysis	análise de cobertura
coverage item	item de cobertura
coverage measurement tool	ferramenta de medição de cobertura
coverage tool	ferramenta de cobertura
critical success factor	fator crítico de sucesso
Critical Testing Processes	processos críticos de teste
CTP	CTP
custom software	software personalizado
custom tool	ferramenta personalizada
cyclomatic complexity	complexidade ciclomática
cyclomatic number	número ciclomático
<b>D</b>	
daily build	entrega diária
dashboard	painel de instrumentos
data definition	definição dos dados
data flow	fluxo de dados
data flow analysis	análise do fluxo de dados
data flow coverage	cobertura de fluxo de dados
data flow testing	teste do fluxo de dados
data integrity testing	teste da integridade de dados
data quality	qualidade de dados
database integrity testing	teste da integridade da base de dados
data-driven testing	teste orientado a dados
dd-path	caminho-dd
dead code	código morto
debugger	<i>debugger</i>
debugging	<i>debugging</i>
debugging tool	ferramenta de <i>debug</i>
decision	decisão
decision condition coverage	cobertura de condições e decisões

<b>Inglês</b>	<b>Português</b>
decision condition testing	teste às condições e decisões
decision coverage	cobertura de decisões
decision outcome	resultado da decisão
decision table	tabela de decisão
decision table testing	teste baseado em tabelas de decisão
decision testing	teste de decisão
defect	defeito
defect based technique	técnica baseada em defeitos
defect based test design technique	técnica de concepção de testes baseada em defeitos
defect category	categoria de defeito
defect density	densidade de defeitos
defect detection percentage	percentagem de detecção de defeitos (PDD)
defect management	gestão de defeitos
defect management committee	comité de gestão de defeitos
defect management tool	ferramenta de gestão de defeitos
defect masking	camuflagem de defeitos
defect report	relatório de defeitos
defect taxonomy	taxonomia de defeitos
defect tracking tool	ferramenta de controlo de defeitos
defect triage committee	comité de gestão de defeitos
defect type	tipo de defeito
definition-use pair	par definição-uso
deliverable	entrega
deming cycle	ciclo de Deming
design based testing	teste baseado na concepção
desk checking	verificação de secretária
development testing	teste de desenvolvimento
deviation	desvio
deviation report	relatório de desvios
diagnosing (IDEAL)	diagnóstico (IDEAL)
dirty testing	teste sujo
documentation testing	teste de documentação
domain	domínio
domain analysis	análise de domínio
driver	controlador
dynamic analysis	análise dinâmica
dynamic analysis tool	ferramenta de análise dinâmica
dynamic comparison	comparação dinâmica
dynamic testing	teste dinâmico
<b>E</b>	
effectiveness	eficácia
efficiency	eficiência
efficiency testing	teste de eficiência
EFQM excellence model	modelo de excelência da EFQM
elementary comparison testing	teste de comparação elementar
embedded iterative development model	modelo embebido de desenvolvimento iterativo
emotional intelligence	inteligência emocional
emulator	emulador

<b>Inglês</b>	<b>Português</b>
entry criteria	critério de entrada
entry point	ponto de entrada
equivalence class	classe de equivalência
equivalence partition	partição de equivalência
equivalence partition coverage	cobertura de partição de equivalência
equivalence partitioning	particionar por equivalências
error	erro
error guessing	antecipar erros
error seeding	injeção de erros
error seeding tool	ferramenta de injeção de erros
error tolerance	tolerância ao erro
establishing (IDEAL)	estabelecer (IDEAL)
evaluation	avaliação
exception handling	tratamento de exceções
executable statement	instrução executável
exercised	executado
exhaustive testing	teste exaustivo
exit criteria	critério de saída
exit point	ponto de saída
expected outcome	resultado esperado
expected result	resultado esperado
experience-based technique	técnica baseada na experiência
experience-based test design technique	técnica de concepção de testes baseada na experiência
experience-based testing	teste baseado na experiência
exploratory testing	testes exploratórios
extreme programming	eXtreme Programming
<b>F</b>	
factory acceptance testing	testes de aceitação de fábrica
fail	sem sucesso
failover testing	teste sobre falhas
failure	falha
failure mode	modo de falha
fault injection	injeção de falhas
FMEA - Failure Mode and Effect Analysis	modo de falha e análise de efeitos
FMECA - Failure Mode, Effects, and Criticality Analysis	modo de falha, efeitos e análise crítica
failure mode	modo de falha
failure rate	taxa de falhas
false-fail result	falso positivo
false-pass result	falso negativo
false-positive result	falso positivo
false-negative result	falso negativo
Fault	defeito
fault attack	ataque
fault density	densidade de defeitos
FDP - Fault Detection Percentage	percentagem de detecção de defeitos (PDD)
fault injection	injeção de falhas
fault masking	camuflagem de defeitos

<b>Inglês</b>	<b>Português</b>
fault seeding	semear falhas
fault seeding tool	ferramenta de injeção de falhas
fault tolerance	tolerância a falhas
FTA - Fault Tree Analysis	Análise de Árvore de Falhas
feasible path	caminho exequível
Feature	característica
feature-driven development	desenvolvimento guiado por características
field testing	teste em campo
finite state machine	máquina de estados finita
finite state testing	teste de estados finito
fishbone diagram	diagrama em espinha
formal review	revisão formal
frozen test basis	base para testes aprovada
FPA - Function Point Analysis	análise por Pontos de Função
functional integration	integração funcional
functional requirement	requisito funcional
functional test design technique	técnica de concepção de testes funcionais
functional testing	teste funcional
functionality	funcionalidade
functionality testing	teste de funcionalidade
<b>G</b>	
glass box testing	teste caixa-branca
Goal Question Metric	Métrica Questão Objetivo
GQM	Métrica Questão Objetivo
<b>H</b>	
hardware-software integration testing	teste de integração hardware-software
hazard analysis	análise de risco
heuristic evaluation	avaliação heurística
high level test case	caso de teste de alto nível
horizontal traceability	rastreabilidade horizontal
hyperlink	hiperligação
hyperlink test tool	ferramenta de teste a hiperligações
<b>I</b>	
IDEAL	IDEAL
impact analysis	análise de impacto
incident	incidente
incident logging	registro de incidentes
incident management	gestão de incidentes
incident management tool	ferramenta de gestão de incidentes
incident report	relatório de incidentes
incremental development model	modelo de desenvolvimento incremental
incremental testing	teste incremental
independence of testing	independência do teste
indicator	indicador
infeasible path	caminho não exequível

<b>Inglês</b>	<b>Português</b>
informal review	revisão informal
initiating (IDEAL)	inicializar (IDEAL)
input	entrada
input domain	domínio de entrada
input value	valor de entrada
insourced testing	testes na fonte
inspection	inspeção
inspection leader	moderador
inspector	inspetor
installability	instalabilidade
installability testing	teste de instalabilidade
installation guide	guia de instalação
installation wizard	assistente de instalação
instrumentation	instrumentação
instrumenter	instrumentador
intake test	teste de admissão
integration	integração
integration testing	teste de integração
integration testing in the large	teste de integração em larga escala
integration testing in the small	teste de integração em pequena escala
interface testing	teste de interface
interoperability	interoperabilidade
interoperability testing	teste de interoperabilidade
invalid testing	teste inválido
Ishikawa diagram	diagrama de Ishikawa
isolation testing	teste de isolamento
item transmittal report	relatório de transmissão de item
iterative development model	modelo de desenvolvimento iterativo
<b>K</b>	
key performance indicator	indicador principal de desempenho
keyword-driven testing	teste orientado a palavras-chave
<b>L</b>	
LCSAJ - Linear Code Sequence And Jump	sequência linear de código e salto
LCSAJ coverage	cobertura de sequência linear de código e salto
LCSAJ testing	teste de sequência linear de código e salto
lead assessor	Líder de avaliação
learnability	aprendibilidade
learning (IDEAL)	aprendizagem (IDEAL)
level test plan	plano para um nível de testes
lifecycle model	modelo de ciclo de vida
link testing	teste de integração de componentes
load profile	perfil de carga
load testing	teste de carga
load testing tool	ferramenta de teste de carga
logic-coverage testing	teste de cobertura lógica
logic-driven testing	teste dirigido pela lógica

<b>Inglês</b>	<b>Português</b>
logical test case	caso de teste lógico
low level test case	caso de teste detalhado
<b>M</b>	
maintainability	manutenibilidade
maintainability testing	teste de manutenibilidade
maintenance	manutenção
maintenance testing	teste de manutenção
man in middle attack	ataque com homem no meio
management review	revisão de gestão
manufacturing-based quality	qualidade baseada na produção
master test plan	plano mestre de testes
maturity	maturidade
maturity level	nível de maturidade
maturity model	modelo de maturidade
measure	medida
measurement	medição
measurement scale	escala de medida
memory leak	memória perdida
methodical testing	teste metódico
metric	métrica
migration testing	teste de migração
milestone	marco
mind map	mapa mental
mistake	engano
model-based testing	teste baseado em modelos
modeling tool	ferramenta de modelação
moderator	moderador
modified condition decision coverage	cobertura modificada de decisões e condições
modified condition decision testing	teste modificado de condições e decisões
modified multiple condition coverage	cobertura modificada de múltiplas condições
modified multiple condition testing	teste modificado de múltiplas condições
module	módulo
module testing	teste de módulo
monitor	monitor
monitoring tool	ferramenta de monitorização
monkey testing	teste do macaco
MTBF	tempo médio entre falhas
MTTR	tempo médio para reparar
multiple condition	múltiplas condições
multiple condition coverage	cobertura de múltiplas condições
multiple condition testing	teste de múltiplas condições
mutation analysis	análise de mutação
mutation testing	teste de mutação
Myers-Briggs Type Indicator (MBTI)	indicador tipo Myers-Briggs

Inglês	Português
<b>N</b>	
n-switch coverage	cobertura N-switch
n-switch testing	teste N-switch
negative testing	teste negativo
neighborhood integration testing	teste de integração por vizinhos
non-conformity	não conformidade
non-functional requirement	requisito não funcional
non-functional test design technique	técnica de conceção de testes não funcionais
non-functional testing	teste não funcional
<b>O</b>	
off-the-shelf-software	software de prateleira
open source tool	ferramenta open source
operability	operabilidade
operational acceptance testing	teste de aceitação operacional
operational environment	ambiente operacional
operational profile	perfil operacional
operational profile testing	teste de perfil operacional
operational profiling	definição de perfil operacional
operational testing	teste operacional
oracle	oráculo
orthogonal array	matriz ortogonal
orthogonal array testing	teste matriz ortogonal
outcome	resultado
outsourced testing	teste por terceiros
output	saída
output domain	domínio de saída
output value	valor de saída
<b>P</b>	
pair programming	programação em pares
pair testing	teste em pares
pairwise integration testing	teste de integração aos pares
pairwise testing	teste aos pares
Pareto analysis	análise de Pareto
partition testing	teste de partição
pass	aprovação
pass/fail criteria	critérios para aprovação/reprovação
path	caminho
path coverage	cobertura de caminhos
path sensitizing	sensibilização de caminho
path testing	teste de caminho
peer review	revisão por pares
performance	desempenho
performance indicator	indicador de desempenho
performance profiling	perfil de desempenho
performance testing	teste de desempenho
performance testing tool	ferramenta de teste de desempenho

<b>Inglês</b>	<b>Português</b>
phase containment	
phase test plan	plano de fase de testes
planning poker	planeamento poker
pointer	apontador
portability	portabilidade
portability testing	teste de portabilidade
postcondition	pós-condição
post-execution comparison	comparação pós-execução
post-project meeting	reunião pós-projeto
precondition	pré-condição
predicate	predicado
predicted outcome	resultado esperado
pretest	pré-teste
priority	prioridade
probe effect	efeito da monitorização
problem	problema
problem management	gestão de problemas
problem report	relatório de problema
procedure testing	teste de procedimento
process	processo
process assessment	avaliação de processo
process-compliant testing	teste de conformidade com processo
process cycle test	teste de ciclo de processo
process improvement	melhoria de processos
process model	modelo de processos
product-based quality	qualidade baseada no produto
product risk	risco de produto
production acceptance testing	teste de aceitação da produção
program instrumenter	instrumentador de programa
program testing	teste de programa
project	projeto
project retrospective	histórico de projeto
project risk	risco de projeto
project test plan	plano de teste do projeto
pseudo-random	pseudoaleatório
<b>Q</b>	
qualification	qualificação
quality	qualidade
quality assurance	garantia de qualidade
quality attribute	atributo de qualidade
quality characteristic	característica de qualidade
quality control	controlo de qualidade
quality gate	marco de qualidade
quality management	gestão da qualidade
quality risk	risco de qualidade
<b>R</b>	
RACI matrix	matriz RACI
reactive testing	teste reativo

<b>Inglês</b>	<b>Português</b>
random testing	teste aleatório
RUP - Rational Unified Process	Rational Unified Process
reactive testing	teste reativo
recorder	anotador
record/playback tool	ferramenta de captura/reprodução
recoverability	recuperabilidade
recoverability testing	teste de recuperabilidade
recovery testing	teste de recuperação
regression-averse testing	testes adversos à regressão
regression testing	teste de regressão
regulation testing	teste de regulamentação
release note	nota de entrega
reliability	fiabilidade
reliability growth model	modelo de aumento da fiabilidade
reliability testing	teste de fiabilidade
replaceability	substitutibilidade
requirement	requisito
requirements-based testing	teste baseado em requisitos
requirements management tool	ferramenta de gestão de requisitos
requirements phase	fase de definição de requisitos
resource utilization	utilização de recursos
resource utilization testing	teste de utilização de recursos
result	resultado
resumption criteria	critério de recomeço
resumption requirements	requisitos de recomeço
re-testing	re-teste
retrospective meeting	reunião de análise retrospectiva
review	revisão
review plan	plano de revisão
review tool	ferramenta de revisão
reviewer	revisor
risk	risco
risk analysis	análise de risco
risk assessment	avaliação de risco
risk-based testing	testes baseados na avaliação do risco
risk category	categorias de risco
risk control	controlo de risco
risk identification	identificação de risco
risk impact	impacto do risco
risk level	nível de risco
risk management	gestão de risco
risk mitigation	mitigação do risco
risk likelihood	probabilidade do risco
risk type	tipo de risco
robustness	robustez
robustness testing	teste de robustez
root cause	origem do defeito
root cause analysis	análise da origem do defeito
RUP	Rational Unified Process
<b>S</b>	

<b>Inglês</b>	<b>Português</b>
safety	segurança
safety critical system	sistema crítico
safety testing	testes de segurança
sanity test	teste rápido
scalability	escalabilidade
scalability testing	teste de escalabilidade
scenario testing	teste de cenário
scorecard	scorecard
scribe	redator
scripted testing	teste com recurso a guiões
scripting language	linguagem de script
SCRUM	SCRUM
security	segurança
security testing	testes de segurança
security testing tool	ferramenta de testes de segurança
security tool	ferramenta de segurança
serviceability testing	teste de disponibilidade
session-based test management	gestão de testes baseada em sessões
session-based testing	teste baseado em sessões
severity	gravidade
short-circuiting	curto-circuito
simulation	simulação
simulator	simulador
site acceptance testing	testes de aceitação locais
smoke test	teste rápido (termo preferencial) teste de fumo (termo alternativo)
software	software
software attack	ataque de software
software integrity level	nível de integridade de software
SFMEA - Software Failure Mode and Effect Analysis	modo de falha de software e análise de efeitos
SFMECA - Software Failure Mode, Effects, and Criticality Analysis	modo de falha de software, efeitos e análise crítica
SFTA - Software Fault Tree Analysis	Análise de Árvore de Falhas de Software
software feature	funcionalidade de software
software lifecycle	ciclo de vida do software
Shewhart chart	gráfico de Shewhart
SPI - Software Process Improvement	melhoria do processo de desenvolvimento de software
software product characteristic	características de um produto de software
software quality	qualidade de software
software quality characteristics	características de qualidade do software
software test incident	incidente de teste de software
software test incident report	relatório de incidentes de teste de software
SUMI - Software Usability Measurement Inventory	SUMI - Software Usability Measurement Inventory
source statement	instrução de código fonte
S.M.A.R.T. goal methodology	metodologia por objetivos S.M.A.R.T.
specification	especificação
specification-based testing	teste baseado nas especificações

<b>Inglês</b>	<b>Português</b>
specification-based technique	técnica baseada nas especificações
specification-based test design technique	técnica de conceção de testes baseada nas especificações
specified input	entrada especificada
SPI	melhoria do processo de desenvolvimento de software
stability	estabilidade
staged representation	representação por níveis
standard	norma standard
standard-compliant testing	teste de conformidade com standards
standard software	software standard
standards testing	teste de conformidade
state diagram	diagrama de estados
state table	tabela de estados
state transition	transição de estados
state transition testing	teste de transição de estados
statement	instrução
statement coverage	cobertura de instruções
statement testing	teste de instrução
static analysis	análise estática
static analysis tool	ferramenta de análise estática
static analyser	analisador estático
static code analysis	análise estática de código
static code analyzer	analisador estático de código
static testing	teste estático
statistical testing	teste estatístico
status accounting	registo de estados
STEP	processo sistemático de teste e avaliação
storage	armazenamento
storage testing	teste de armazenamento
stress testing	teste de stress
stress testing tool	ferramenta de teste de stress
structural coverage	cobertura estrutural
structural test design technique	técnica de conceção de testes estrutural
structural testing	teste estrutural
structure-based technique	técnica baseada na estrutura
structure-based test design technique	técnica de conceção de testes baseada na estrutura
structure-based testing	teste baseado na estrutura
structured walkthrough	apresentação estruturada de documento
stub	simulador
Shewhart chart	gráfico Shewhart
subpath	segmento
suitability	adequabilidade
suitability testing	testes de adequabilidade
suspension criteria	critério de suspensão
syntax testing	testes de sintaxe
system	sistema
system integration testing	teste de integração de sistemas
system of systems	sistema composto (preferencial) sistema de sistemas

<b>Inglês</b>	<b>Português</b>
system testing	teste de sistema
STEP - Systematic Test and Evaluation Process	processo sistemático de teste e avaliação
<b>T</b>	
technical review	revisão técnica
test	teste
test approach	abordagem de teste
test architect	arquitecto de teste
test automation	automação de testes
test basis	base para testes
test bed	ambiente de teste
test case	caso de teste
test case design technique	técnica de conceção de casos de teste
test case specification	especificação de caso de teste
test case suite	conjunto de casos de teste
test charter	ficha de testes
test closure	fecho dos testes
test comparator	comparador de teste
test comparison	comparação de teste
test completion criteria	critério de conclusão dos testes
test condition	condição de teste
test control	controlo de testes
test coverage	cobertura de teste
test cycle	ciclo de teste
test data	dados de teste
test data management	gestão de dados de teste
test data preparation tool	ferramenta de preparação de dados de teste
test deliverable	entrega de teste
test design	conceção de teste
test design specification	especificação de conceção de testes
test design technique	técnica de conceção de testes
test design tool	ferramenta de conceção de testes
test director	director/gestor de testes
test driven development	desenvolvimento orientado a testes
test driver	controlador de testes
test environment	ambiente de teste
test estimation	estimativa de teste
test evaluation report	relatório de avaliação de teste
test execution	execução de teste
test execution automation	automação de execução de teste
test execution phase	fase de execução de teste
test execution schedule	cronograma de execução de testes
test execution technique	técnica de execução de testes
test execution tool	ferramenta de execução de testes
test fail	teste falhado
test generator	gerador de testes
test harness	equipamento de teste
test implementation	implementação do teste
test improvement plan	plano de melhoria de testes
test incident	incidente de teste

<b>Inglês</b>	<b>Português</b>
test incident report	relatório de incidentes de teste
test infrastructure	infraestrutura de teste
test input	entrada de teste
test item	item de teste
test item transmittal report	relatório de transmissão de item de teste
test leader	líder de teste
test level	nível de teste
test log	registo de teste
test logging	gravação do registo de teste
test management	gestão de testes
test management tool	ferramenta de gestão de testes
test manager	gestor de testes
test mission	missão de teste
test monitoring	monitorização de teste
test object	objeto de teste
test objective	objetivo de teste
test oracle	oráculo de teste
test outcome	resultado do teste
test pass	aprovação de teste
test performance indicator	indicador de desempenho de teste
test phase	fase de teste
test plan	plano de testes
test planning	planeamento de testes
test policy	política de teste
test procedure	procedimento de teste
test procedure specification	especificação do procedimento de teste
test process	processo de teste
test process improvement manifesto	manifesto de melhoria do processo de teste
test process improver	otimizador do processo de teste
test progress report	relatório de progresso de testes
test record	registo de teste
test recording	gravação do registo de teste
test report	relatório de testes
test reproducibility	reprodutibilidade dos testes
test requirement	requisito de teste
test result	resultado do teste
test rig	plataforma de teste
test run	execução de teste
test run log	registo da execução do teste
test scenario	cenário de teste
test schedule	cronograma de testes
test script	guião de teste
test session	sessão de teste
test set	bateria de testes
test situation	situação de teste
test specification	especificação do teste
test specification technique	técnica de especificação de teste
test stage	fase de teste
test strategy	estratégia de teste
test suite	bateria de testes
test summary report	relatório sumário de testes

<b>Inglês</b>	<b>Português</b>
test target	critério de conclusão dos testes
test technique	técnica de teste
test tool	ferramenta de teste
test type	tipo de teste
testability	testabilidade
testability review	revisão de testabilidade
testable requirements	requisitos testáveis
tester	testador
testing	testar
testware	<i>testware</i>
thread testing	teste de <i>thread</i>
three point estimation	estimativa de três pontos
time behavior	comportamento no tempo
top-down testing	testes descendentes
TMMi - Test Maturity Model Integration	Test Maturity Model Integration (TMMi)
TPA - Test Point Analysis	Test Point Analysis (TPA)
TQM - Total Quality Management	Gestão de Qualidade Total
TPG - Test Process Group	Grupo do Processo de Teste
TPI	Melhoria do Processo de Teste
TPI Next	TPI Next
TQM	Gestão de Qualidade Total
traceability	rastreabilidade
transactional analysis	análise transacional
transcendent-based quality	qualidade baseada na transcendência
<b>U</b>	
understandability	compreensibilidade
unit	unidade
unit test framework	<i>framework</i> de testes unitários
unit testing	teste unitário
unreachable code	código inatingível
usability	usabilidade
usability testing	teste de usabilidade
use case	caso de uso
use case testing	teste de casos de uso
user acceptance testing	teste de aceitação de utilizador
user-based quality	qualidade baseada no utilizador
user scenario testing	teste de cenário de utilizador
user story testing	teste às user story
user story	user story
user test	teste de utilizador
<b>V</b>	
V-model	modelo em V
validation	validação
value-based quality	qualidade baseada no custo
variable	variável
verification	verificação

<b>Inglês</b>	<b>Português</b>
version control	controlo de versão
vertical traceability	rastreabilidade vertical
volume testing	teste de volume
<b>W</b>	
walkthrough	apresentação de documento
WAMMI	WAMMI
WBS	Work Breakdown Structure
Website Analysis and Measurement Inventory (WAMMI)	Website Analysis and Measurement Inventory (WAMMI)
white-box technique	técnica caixa-branca
white-box test design technique	técnica caixa-branca de conceção de testes
white-box testing	teste caixa-branca
Wide Band Delphi	Wide Band Delphi
wild pointer	apontador partido
Work Breakdown Structure	Work Breakdown Structure