



Unidade 2

Fundamentos do Processo de Teste

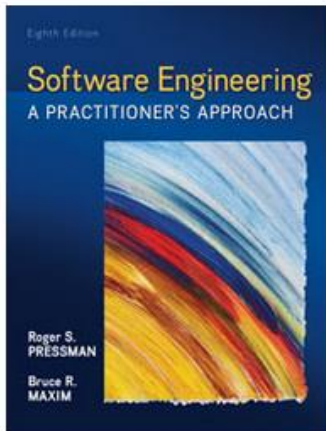


Prof. Aparecido V. de Freitas
Doutor em Engenharia
da Computação pela EPUSP
CPRE – CTFL – CTFL-AT
aparecidovfreitas@gmail.com

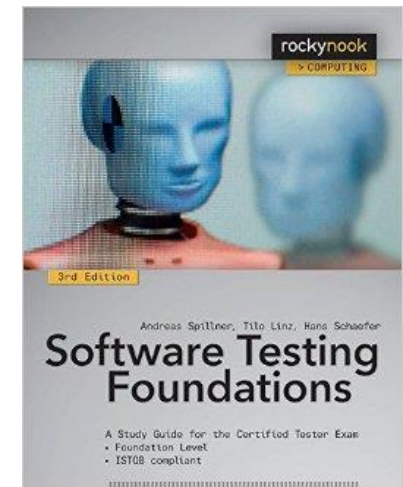
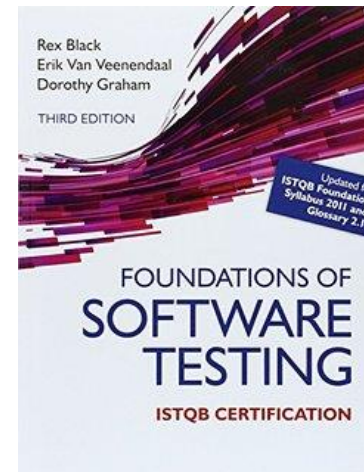
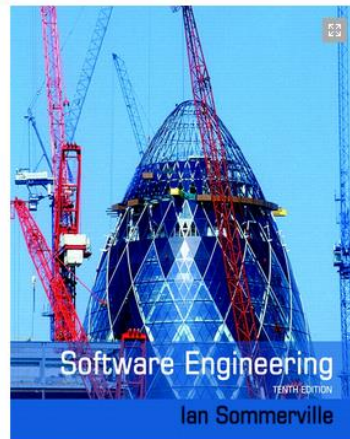


Bibliografia

- **Software Engineering – A Practitioner's Approach – Roger S. Pressman – Eight Edition – 2014**
- **Software Engineering – Ian Sommerville – 10th edition - 2015**
- Engenharia de Software – Uma abordagem profissional – Roger Pressman - McGraw Hill, Sétima Edição - 2011
- Engenharia de Software – Ian Sommerville – Nona Edição – Addison Wesley, 2007
- **Software Testing Foundations – 4th edition – Andreas Spillner, Tito Linz, Hans Schaefer – 2014**
- **Foundations of Software Testing – Third Edition – Rex Black – ISTQB Certification**



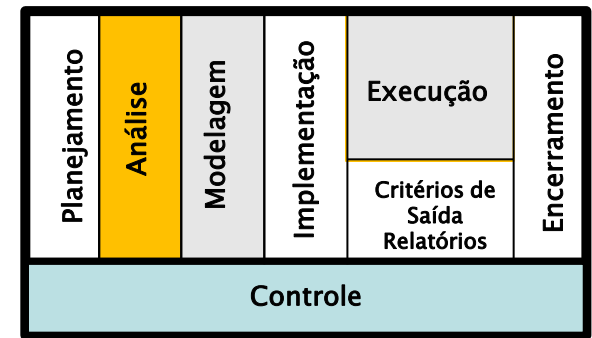
[Software Engineering: A Practitioner's Approach, 8/e](#)





Processos de Teste

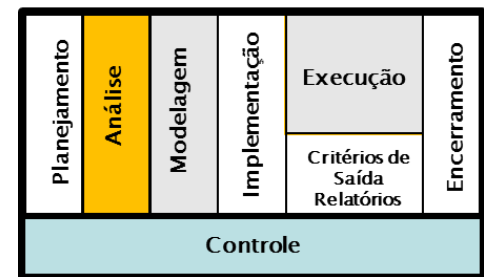
- ✓ Não existe um **processo universal** de Teste de Software;
- ✓ Porém, há conjuntos **comuns** de **atividades** de teste sem as quais os testes terão menor probabilidade de atingir seus objetivos estabelecidos;
- ✓ Esses conjuntos de atividades de teste são um **Processo de Teste**.





Fatores que influenciam o Processo de Teste

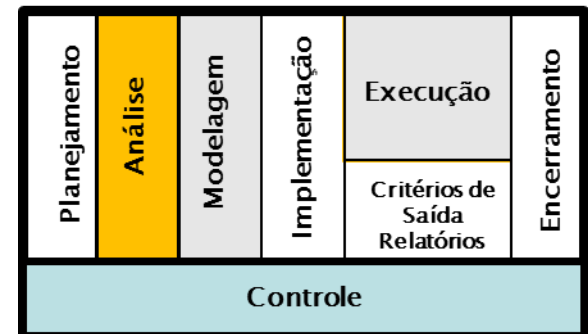
- ✓ Modelo de **ciclo** de **vida** de desenvolvimento de software;
- ✓ **Metodologias** de projeto utilizados;
- ✓ **Riscos** de Produto e Projeto;
- ✓ Domínio do **Negócio**;
- ✓ **Orçamento** e Recursos;
- ✓ Escalas de **Tempo**;
- ✓ **Complexidade**;
- ✓ Requisitos **contratuais**;
- ✓ Políticas, **Normas** internas e Externas necessárias





Normas para o Processo de Testes

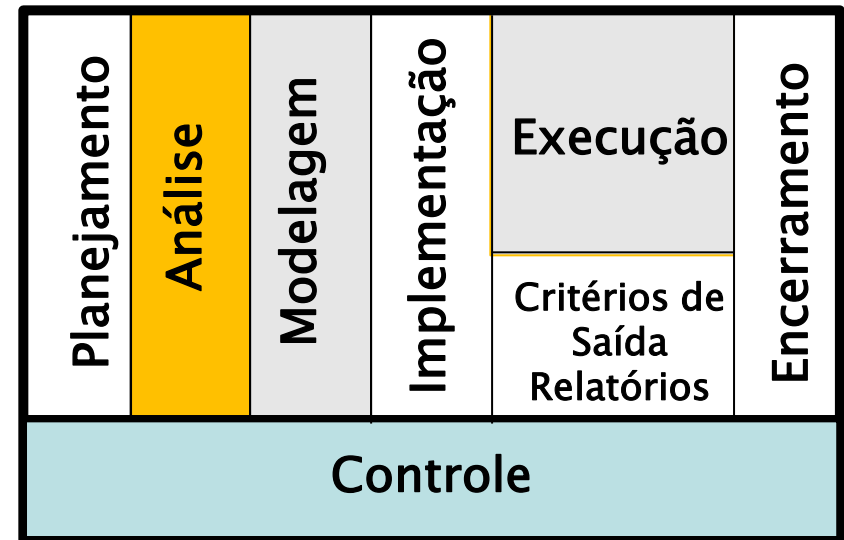
✓ ISO / IEC / IEEE 29119-2





Atividades do Processo de Teste – ISTQB

- ✓ **Planejamento** do Teste
- ✓ **Monitoramento e Controle** do Teste
- ✓ **Análise e Modelagem** do Teste
- ✓ **Implementação e Execução** de Teste
- ✓ **Avaliação** do critério de saída e Relatório
- ✓ Atividades de **Encerramento** de Teste



Linha de tempo do projeto →

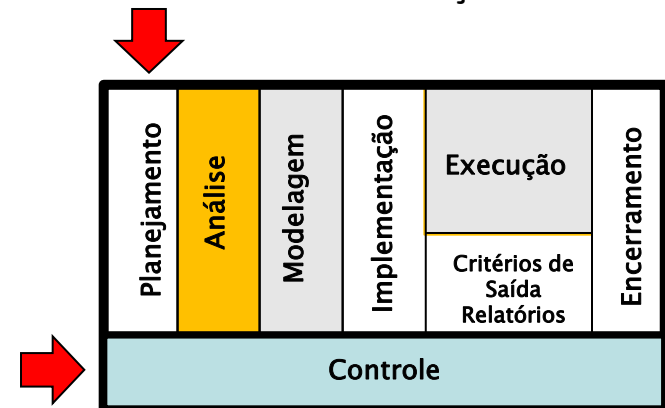


Observação: Essas etapas podem se **sobrepôr**, serem **concorrentes** e/ou **interagir** !



Planejamento e Controle do Teste

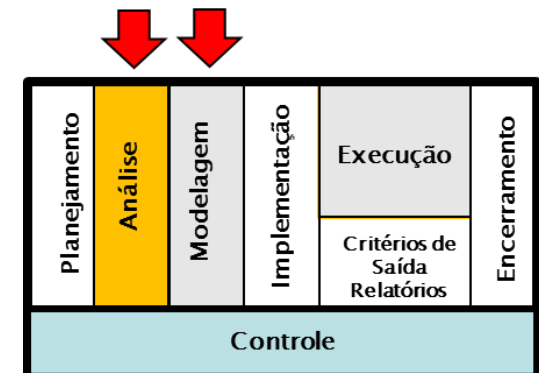
- ✓ O **Planejamento** envolve as atividades que definem os **propósitos** e as abordagens do teste para atender aos objetivos do teste, por exemplo, especificar técnicas e tarefas de teste, cronograma de teste, recursos a serem alocados, custo, etc.
- ✓ O **monitoramento** do Teste é a atividade constante que **compara** o progresso real contra o que foi planejado, comunicando o status e os desvios do plano;
- ✓ O **controle** do Teste envolve as **ações** necessárias para alcançar a missão e objetivos do plano de testes.
- ✓ Para um **controle** efetivo, o teste deverá ser **monitorado** durante todo o projeto;
- ✓ O planejamento de Teste leva em consideração o **retorno** de informações das atividades de Monitoração e Controle.





Análise e Modelagem do Teste

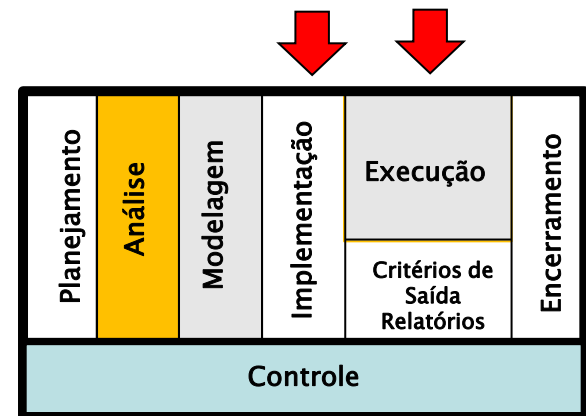
- ✓ Atividades onde a **base de teste** é analisada para se **identificar os recursos testáveis** !
- ✓ Na Análise se responde à questão: **O que deve ser testado?**
- ✓ Compreendem as seguintes atividades:
 - Analisar a base de teste (especificação de requisitos, casos de uso, diagramas UML, especificações de interface, código, modelagem do banco de dados);
 - Avaliar a testabilidade dos requisitos e do sistema (ambiguidades, omissões, inconsistências, imprecisões, contradições, etc);
- ✓ Na modelagem de teste, os casos de teste são projetados e priorizados. O ambiente de teste é modelado.
- ✓ Na Modelagem se responde à questão: **Como testar?**





Implementação e Execução de Teste

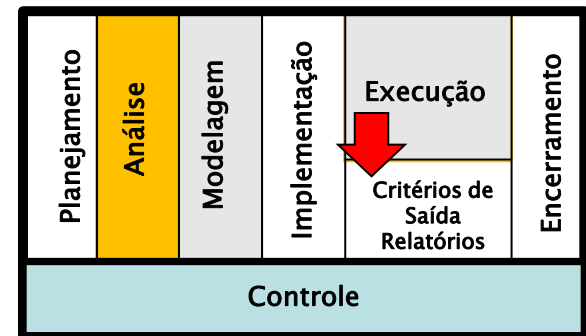
- ✓ Atividade onde os procedimentos ou os scripts de teste são especificados pela combinação dos casos de teste em uma ordem particular, incluindo todas as outras informações necessárias para a execução do teste, o ambiente é preparado e os testes são executados;
- ✓ Na **implementação**, responde-se à pergunta: “**Temos agora tudo que é necessário para iniciar a execução dos Testes ?**”
- ✓ Principais **atividades** da Execução dos Testes:
 - Executar testes (manualmente ou com ferramentas);
 - Comparar resultados;
 - Analisar anomalias para estabelecer prováveis causas;
 - Comunicar defeitos encontrados;
 - Registrar o resultado da execução do teste;
 - Repetir testes como resultado de ações corretivas;





Avaliação de critério de saída e Relatórios

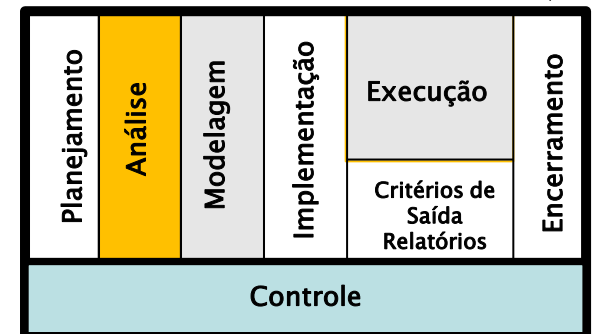
- ✓ Atividade onde a execução do teste é **avaliada** mediante **objetivos definidos**;
- ✓ Fornece a base para se tomar a **decisão** de **parada** do teste;
- ✓ Composta pelas seguintes atividades principais:
 - Checar registros de teste (**logs**);
 - Avaliar se são necessários testes **adicionais** ou se critério de saída de teste deve ser alterado;
 - Elaborar **relatório de teste** resumido para os interessados (**stakeholders**).





Atividades de Encerramento de Teste

- ✓ São **coletados** os dados de todas as atividades para consolidar a experiência, **testware**, fatos e números.
- ✓ São compostas pelas seguintes atividades principais:
 - **Checar** quais entregáveis foram realmente entregues;
 - Fechar relatórios de incidentes;
 - Documentar aceite do sistema;
 - Entregar o **testware** para manutenção da organização;
 - Analisar as lições aprendidas;
 - Utilizar informações coletadas para melhorar a maturidade de teste.





A Psicologia do Teste

- ✓ Formas **diferentes** de pensar:

Desenvolvedor quando há defeitos...



Testador fica feliz quando encontra defeitos...





Formas diferentes de pensar

- ✓ Desenvolvedores são aptos a testarem seus **próprios códigos e projetos**.
- ✓ Testadores têm uma **visão independente**, profissional e treinada sobre os recursos de teste;
- ✓ A separação da responsabilidade de testar ajuda a focar o esforço e provê benefícios adicionais;
- ✓ É difícil testar o que se constrói.





Independência de Teste

- ✓ Certo grau de independência (sem a influência do autor) muitas vezes representa uma forma eficiente de se encontrar defeitos e falhas;
- ✓ Independência não significa simplesmente uma substituição: desenvolvedores podem encontrar defeitos de maneira eficiente. (Cuidado para não virar alibi...)



Quem faz não checka ...

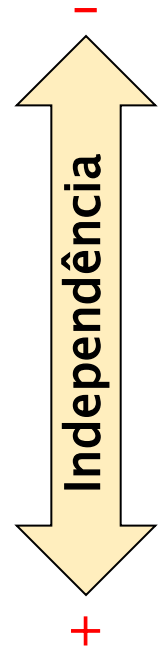


Quem checka não faz ...



Níveis de Independência

- ✓ Teste elaborado por quem escreveu o software (baixo nível de independência);
- ✓ Teste elaborado por outras pessoas da mesma equipe de desenvolvimento;
- ✓ Teste elaborado por pessoas de um grupo organizacional diferente (equipe independente de teste);
- ✓ Teste elaborado por pessoas de diferentes empresas (terceirizada ou certificada por um órgão externo, alto nível de independência);





Testar requer

- ✓ Curiosidade;
- ✓ Pessimismo profissional;
- ✓ Olha crítico;
- ✓ Atenção aos detalhes;
- ✓ Comunicação Eficiente com os Profissionais de Desenvolvimento;
- ✓ Experiência para encontrar Erros;
- ✓ Comunicar bugs de forma construtiva para evitar constrangimentos entre as equipes;





Eficiência de Teste

- ✓ Deve-se preterir testes em áreas de risco;
- ✓ **Engenheiros de Teste efetivos e eficientes**: Têm um talento para consumir tempo onde estão os bugs. Podem fazer uma completa isolamento do bug rapidamente;
- ✓ **Engenheiros de Teste Não efetivos e Ineficientes**: Escrevem testes para procurar bugs improváveis e de baixo impacto. Gastam horas pesquisando bugs triviais;
- ✓ Permanecer focado nos objetivos do projeto de teste.

Efetivo: alcançar uma meta

Eficiente: atingir resultado com pouco esforço





Habilidades do Testador

- ✓ **Leitura.** Especificações, e-mails, casos de teste;
- ✓ **Escrita.** Casos de teste, relatórios de bugs, documentação de teste, etc;
- ✓ **Estatística** e outras operações matemáticas;
- ✓ **Habilidades** pertinentes à Tecnologia, Projeto e Teste:
 - *Linguagens de Programação, Sistemas Operacionais, redes, Web, etc*
 - *Domínio do Negócio das aplicações;*
 - *Produção de Scripts, Automatização, Modelagem de Desempenho;*

