

Teste de Software

# Estimativas, Medidas, Métricas e Indicadores em Gestão de Processo de Teste



**PUC Minas**

Instituto de Ciências Exatas  
e Informática

Prof. Lesandro Ponciano

Departamento de Engenharia de Software  
e Sistemas de Informação (DES)

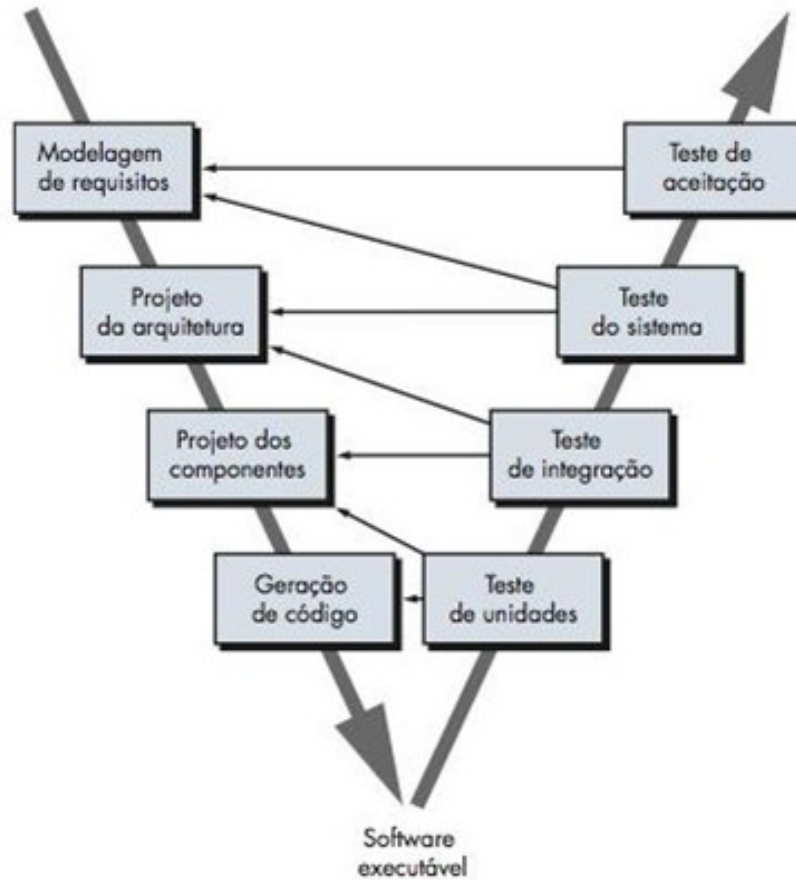
# Objetivos da Aula

---

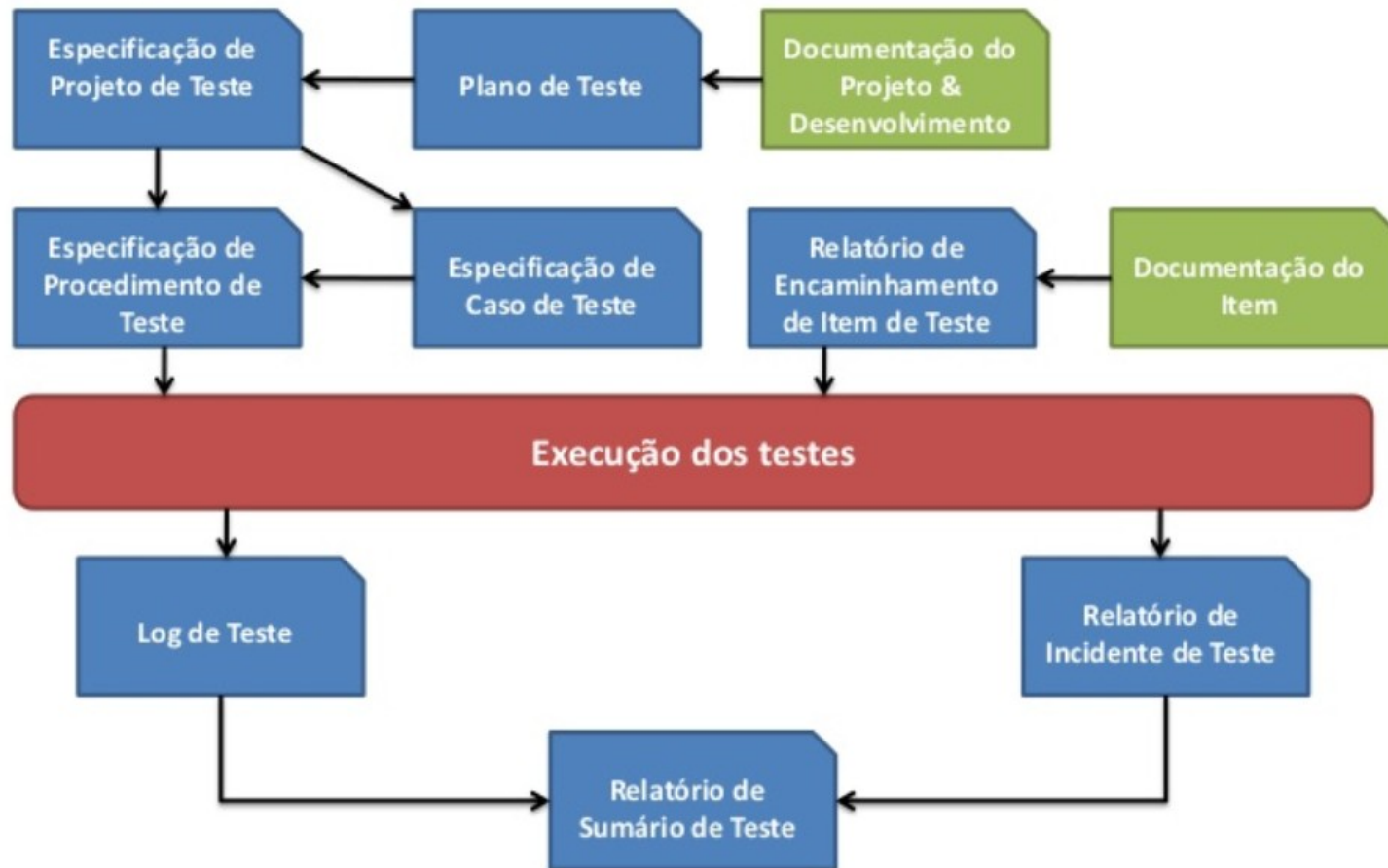
- **Contextualizar** estimativas em Gestão de Processo de Teste
- **Discutir** dimensões do processo, do software, do software de teste
- **Analisar** e contextualizar medidas, métricas e indicadores

# Teste de Software

---



# Padrão IEEE 829-2008

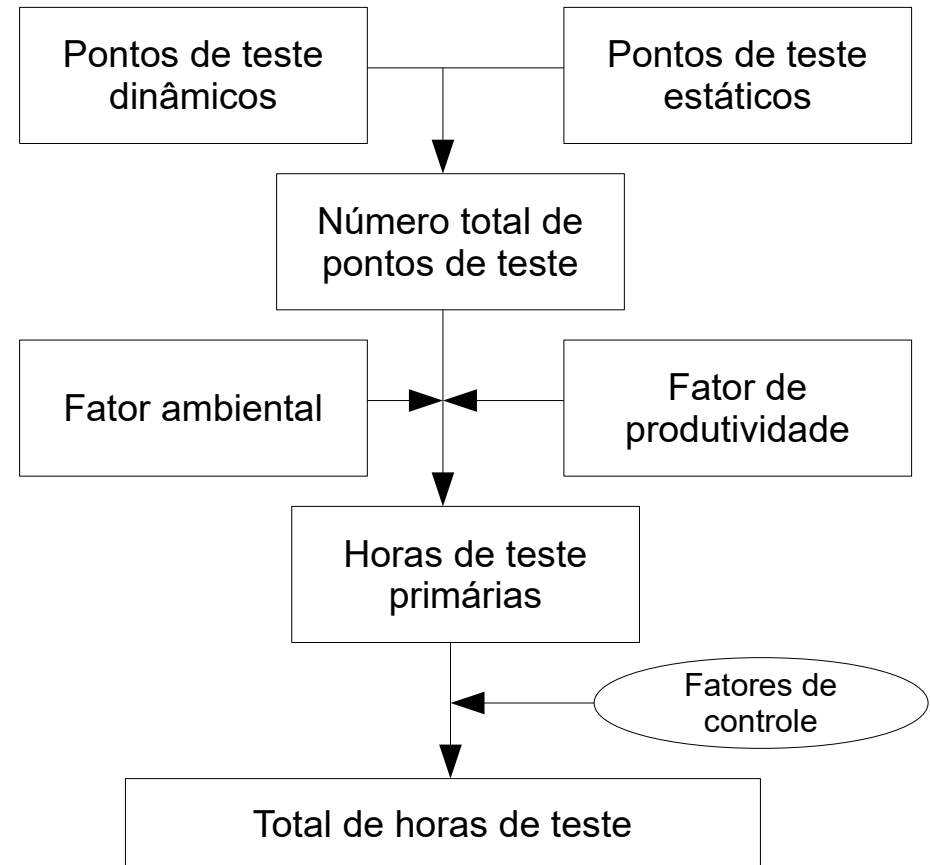
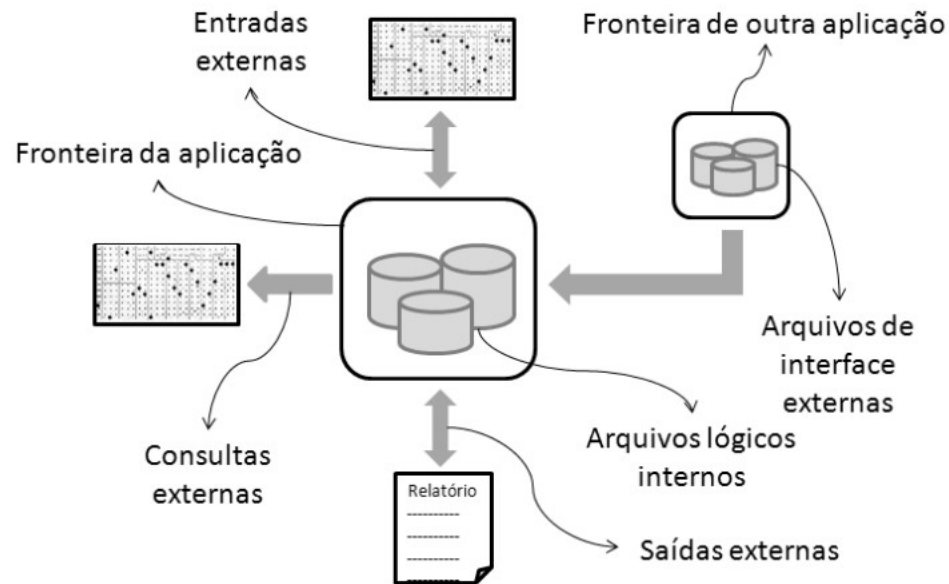


# Estimativas

---

- Uma estimativa é um parecer sobre uma situação baseado nas evidências existentes
- Para obter **evidências**
  - Histórico de medições da organização
  - Modelos matemáticos e/ou empíricos
- Estimativas não são exatas, há **incertezas**
  - Ex.: *Estimamos que seria necessário implementar 16 casos de teste, mas foram necessário 40 casos de teste*
  - Subestimar versus superestimar

# Pontos de Função | Pontos de Teste



# Métrica, Medida e Indicador

- Medida
  - Quantificação de dados em relação a um padrão
  - Ex.: tempo, distância
- Métrica
  - Composição de uma mais mais medidas que é atribuída a um atributo
  - Ex.: *mutation score*
- Indicador
  - Variável que interpreta uma métrica
  - Ex: *mutation score* > x pode indicar que o software é de qualidade

# Goal, Question, Metric (GQM)

- Metas (Objetivo), questão é métrica
- É uma forma sistematizada de se definir métricas
- Exemplo:
  - G: Desenvolver software com qualidade.
  - Q: O software está bem testado?
  - M:
    - M1: Cobertura de caminhos dos testes
    - M2: *Mutation score*



# Diversos Objetivos

---

- 1) Aumentar a capacidade de encontrar defeitos...
- 2) Melhorar a qualidade do código do sistema ...
- 3) Melhorar a qualidade do código de teste do sistema ...
- 4) Monitorar a automação de teste de software ...
- 5) Aumentar a detecção de defeitos gerados por mudança...
- 6) Aumentar a extensão de testes no sistema...
- 7) Diminuir o esforço necessário para criação de teste no sistema...

# Métricas e Indicadores: Software

- Métricas de coesão
  - fatia de dados (*data slices*)
  - Aglutinação (*stickness*)
- Métricas de acoplamento
  - Número de módulos chamados (*fan-out*)
  - Número de módulos que chamam o módulo em consideração (*fan-in*)
- Métricas de complexidade
  - Complexidade ciclomática

# Métricas e Indicadores: Testes

- Abrangência
  - Cobertura (comandos, decisões, condições, caminhos)
  - Fator de Teste =  $N^{\circ} \text{ Linha dos testes} / N^{\circ} \text{ Linhas do sistema}$
  - *Mutation score*
- Necessidade de refatorar a suíte de testes
  - Número de asserções por método
  - Replicação do código dos testes
- Tempo de execução dos testes

# **Métricas e Indicadores: Processo**

---

- Total de Defeitos Detectados (DD)
  - defeitos encontrados durante as fases de teste
- Total de Defeitos Removidos (DR)
  - defeitos efetivamente consertados pela equipe
- Total de Falhas Encontradas pelo Usuário (FU)
  - falhas detectadas pelo usuário durante o uso
- Eficácia na Detecção de Defeitos
  - $DD / (DD + FU) \times 100$

# Indicadores de Defeitos

---

- Úteis ao planejamento e avaliação da estratégia de testes
- Origem dos defeitos
  - Requisitos
  - Projeto
  - Codificação
  - Implantação
  - Documentação
  - Manutenção

Análise de métricas em todas as atividades do processo podem gerar indicadores de defeitos

# **Atividade de Fixação**

---

- Qual a importância de métricas em Teste de Software?
- Dê exemplo de como avaliar se um processo de teste está tendo sucesso?
- Como estimar o esforço que será necessário para testar o sistema a ser desenvolvido?

# Referências

---

- Pressman, Roger, and Bruce Maxim. Engenharia de Software-8ª Edição. McGraw Hill Brasil, 2016.
- Paulo Cheque. Métricas para Testes Automatizados.  
<http://ccsl.ime.usp.br/agilcoop/files/TestesAutomatizados-5-2-Metricas-AgilCoop-Verao2009.pdf>
- IEEE Standard for Software and System Test Documentation (Padrão IEEE 829-2008). Disponível em: <https://standards.ieee.org/standard/829-2008.html>  
Acesso em: 08 jan. 2020
- A Comparative Study of Three Test Effort Estimation Methods.  
<https://pdfs.semanticscholar.org/1a62/5819fbc6ed029b194cbc91be9f16033c309e.pdf>
- Van Veenendaal, E. P. W. M., and Ton Dekkers. "Test point analysis: a method for test estimation." Project control for software quality: proceedings of the combined 10th European Software Control and Metrics conference and the 2nd SCOPE conference on software product evaluation, April 27-29, 1999, Herstmonceux, England. Shaker-Verlag, 1999.