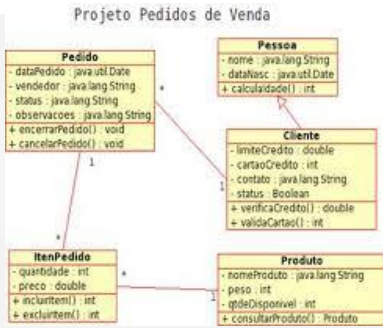


# PROCESSO DE TESTES



Plano de testes

Projeto de software



Codificação

Elicitação/  
Especificação  
de requisitos

Manutenção  
Evolução



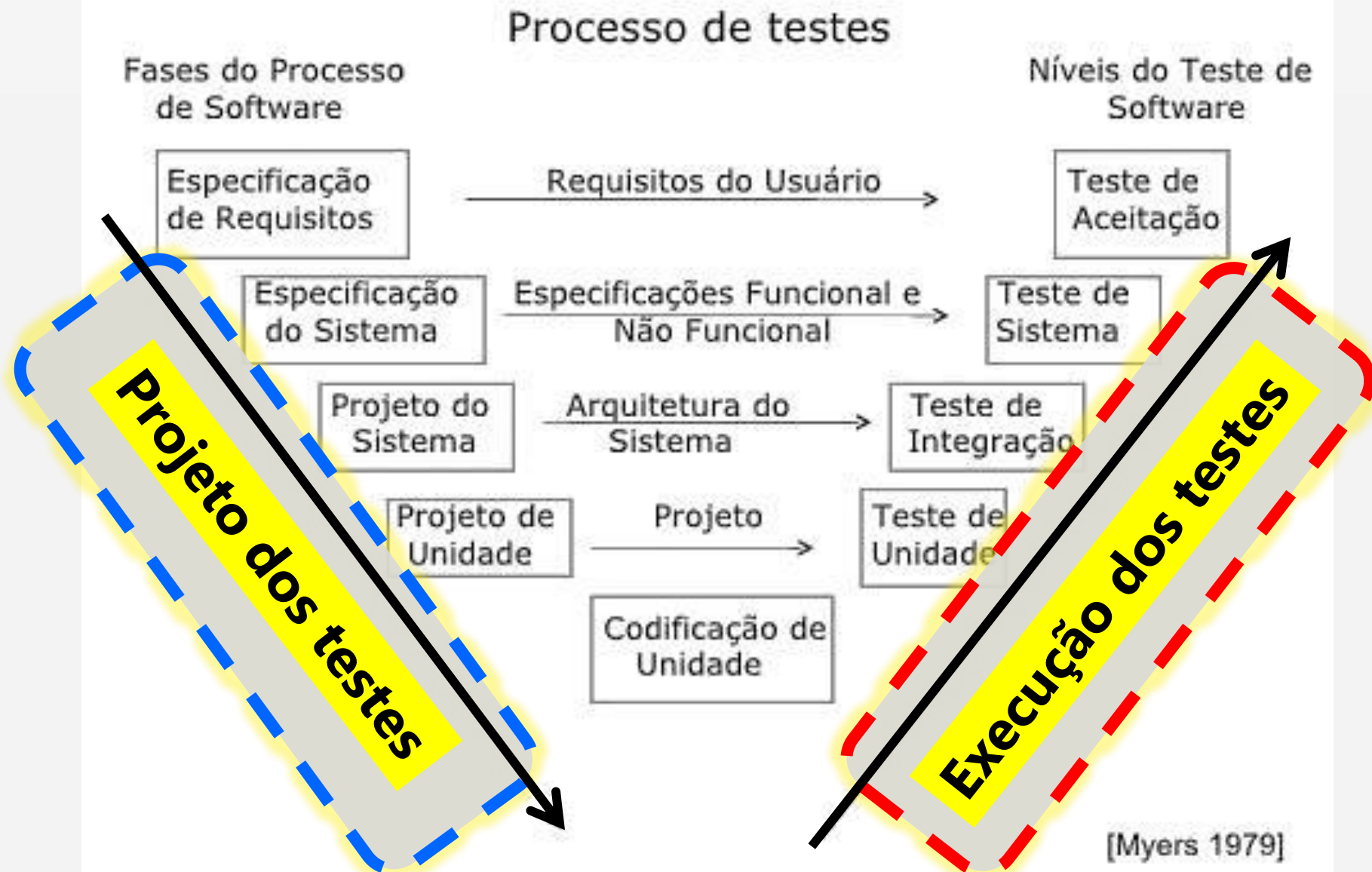
Implantação



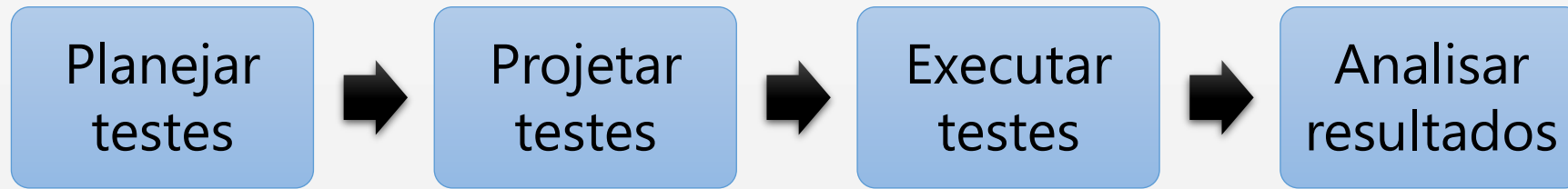
Testes de software

GERÊNCIA DE PROJETOS

# PLANO DE TESTES



# Processo de testes



Gerenciar  
defeitos

1º) **Planejamento:**

Elaboração do **plano de testes**

2º) **Especificação (projetar):**

Elaborar casos de testes

3º) **Execução:**

Executar os testes e registrar os resultados

4º) **Avaliação dos resultados:**

Avaliação da qualidade do software

# Processo de testes

## 1º) Planejamento:

definição dos objetivos dos testes, a identificação dos recursos necessários, a elaboração de planos de teste e a definição dos critérios de aceitação → Elaboração do **plano de testes**

## 2º) Especificação (projetar):

Com base no plano de testes, devem ser criados os casos de teste para cobrir os diferentes aspectos de uso do software → Elaborar **casos de testes**

## 3º) Execução:

Os casos de teste são executados e os resultados são registrados → Executar os **testes e registrar os resultados**

## 4º) Avaliação dos resultados:

Após a conclusão dos testes, um relatório é elaborado para documentar os resultados dos testes, incluindo os casos de teste executados, os defeitos encontrados e as métricas de qualidade → Avaliar a **qualidade do software**

# Elaborar casos de testes

---

## É atividade chave para um teste ser bem sucedido

Caso de teste é um conjunto de condições usadas para testar uma condição particular de um software

- Composto por valores de entrada, condições para a sua execução, e um resultado ou comportamento esperado.

Elaborado para:

- Identificar defeitos na estrutura interna do software (verificação)
- Garantir que os requisitos do software que foi construído sejam plenamente atendidos (validação)

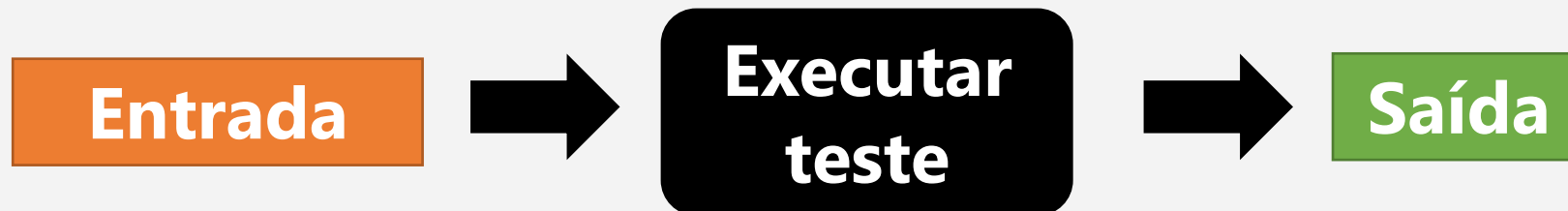
# Elaborar casos de testes

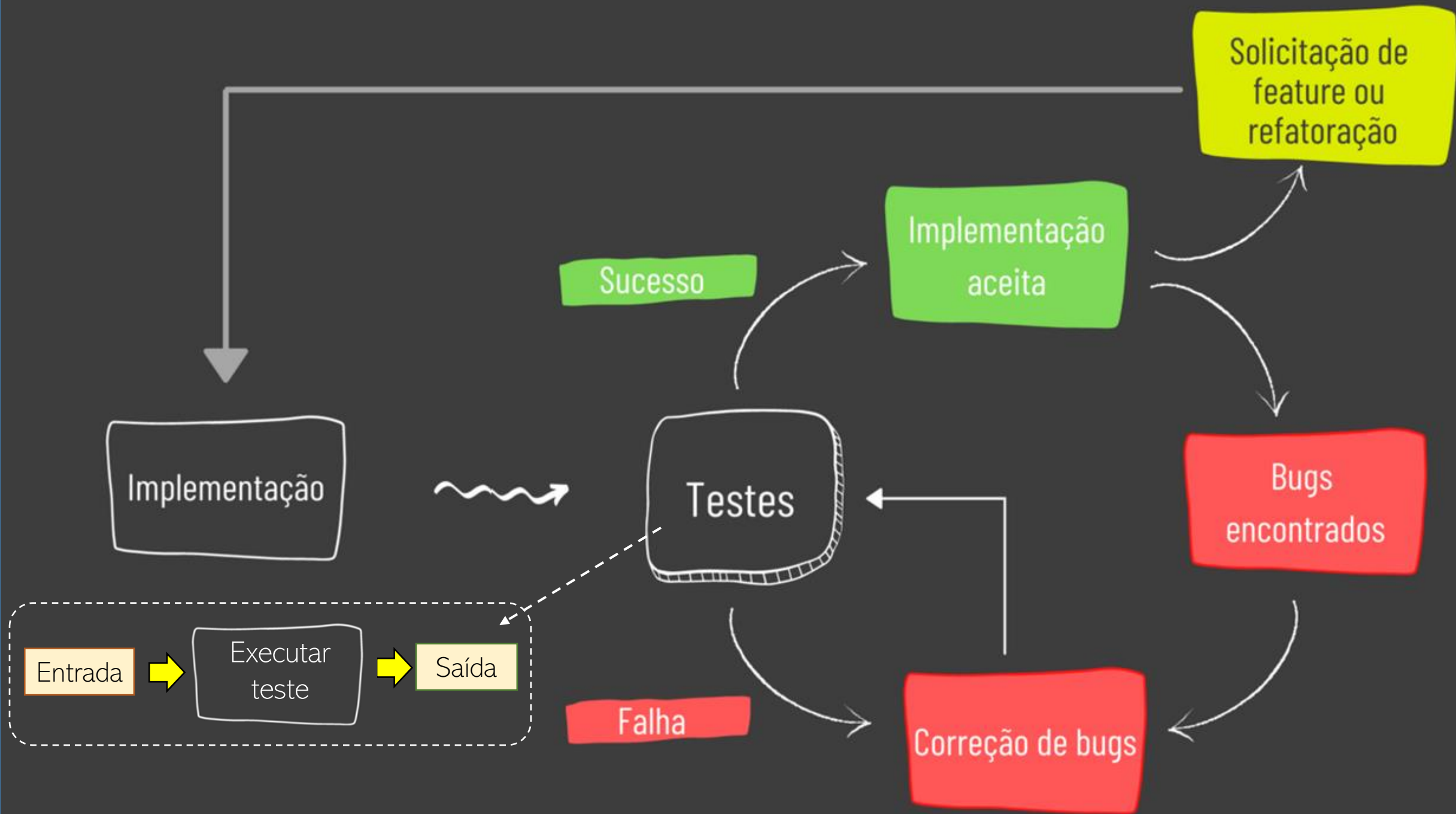
---

**1º) Qual o objetivo (porque testar? o que deve ser avaliado?)**

**2º) Qual ação de teste a ser executado?**

**3º) Qual o resultado ou comportamento esperado?**







---

***Bons casos de testes são aqueles que detectam defeito ainda não descoberto.***

# Estrutura dos casos de teste

---

## IDENTIFICAÇÃO DO CASO DE TESTES:

**ID do Caso de Teste:** Identificador único para o caso de teste (ex: CT-001)

**Título:** Descrição breve do que será testado (ex: "Teste de login com dados válidos e inválidos").

**Descrição:** Contém a descrição do caso de teste, descrevendo a finalidade ou o objetivo do teste e o escopo

**Pré-condições de execução:** Condições em que o sistema deve se encontrar antes do início do teste. Isso pode incluir dados de teste, estado do sistema, ou ações anteriores (ex: "Usuário deve estar registrado no sistema e a aplicação deve estar ativa").

## EXECUÇÃO – ETAPAS A SEREM EXECUTADAS:

**Cenário de teste:**

**Dados de entradas** (dados do teste): Dados que devem ser informados para realização dos testes

**Passos para execução:** Passos a serem seguidos

**Resultados esperados (Critério de aceitação):** Resultado/comportamento esperado pelo sistema após a realização do teste

# Exemplo – caso de testes

**ID do Caso de Teste:** CT-001

**Título:** Teste de login com dados válidos e inválidos

**Descrição do Caso de Teste:** Verificar se um usuário pode fazer login com credenciais válidas

**Pré-condições:**

1. Deve haver um usuário previamente cadastrado no sistema com as credenciais corretas
2. O sistema deve estar disponível

## EXECUÇÃO:

**Cenário 1:** Usuário válido pode acessar o sistema

**Dados de Entrada:**

Nome de usuário: user123

Senha: password123

**Passos para Execução:**

1. Navegar até a página de login
2. Inserir "user123" no campo de nome de usuário
3. Inserir "password123" no campo de senha
4. Clicar no botão "Login"

**Resultados Esperados:**

1. O usuário é redirecionado para a página inicial do sistema
2. Verificar a mensagem de boas-vindas apresenta o nome correto do usuário: "Olá, [Nome do Usuário]"
3. O sistema deve criar uma sessão de usuário

**Para projetar os casos de testes  
é necessário ter acesso  
aos artefatos de planejamento do software**

# Artefatos necessários a serem analisados para criar casos de testes

---

## **1 – Documentos de requisitos: Requisitos funcionais e não funcionais**

- Serve como referência para garantir que todos os requisitos do sistema são cobertos pelos casos de teste

## **2 - Diagrama de casos de uso**

- Descreve como os usuários interagem com o sistema para alcançar objetivos específicos
- Ajuda a criar casos de teste baseados nos diferentes fluxos de uso do sistema, incluindo fluxos alternativos e exceções

## **3 – Protótipo**

- Protótipos visuais que representam a interface do usuário
- Auxilia na criação de casos de teste de interface do usuário, verificando se os elementos de UI estão presentes e funcionam corretamente

## **4 - Modelo-ER ou Documento de Arquitetura de Software**

- Descreve a estrutura geral do sistema
- Ajuda a entender como diferentes partes do sistema se interconectam, permitindo a criação de casos de teste visualizando a integração entre componentes

# Artefatos necessários para serem analisados ao criar casos de testes

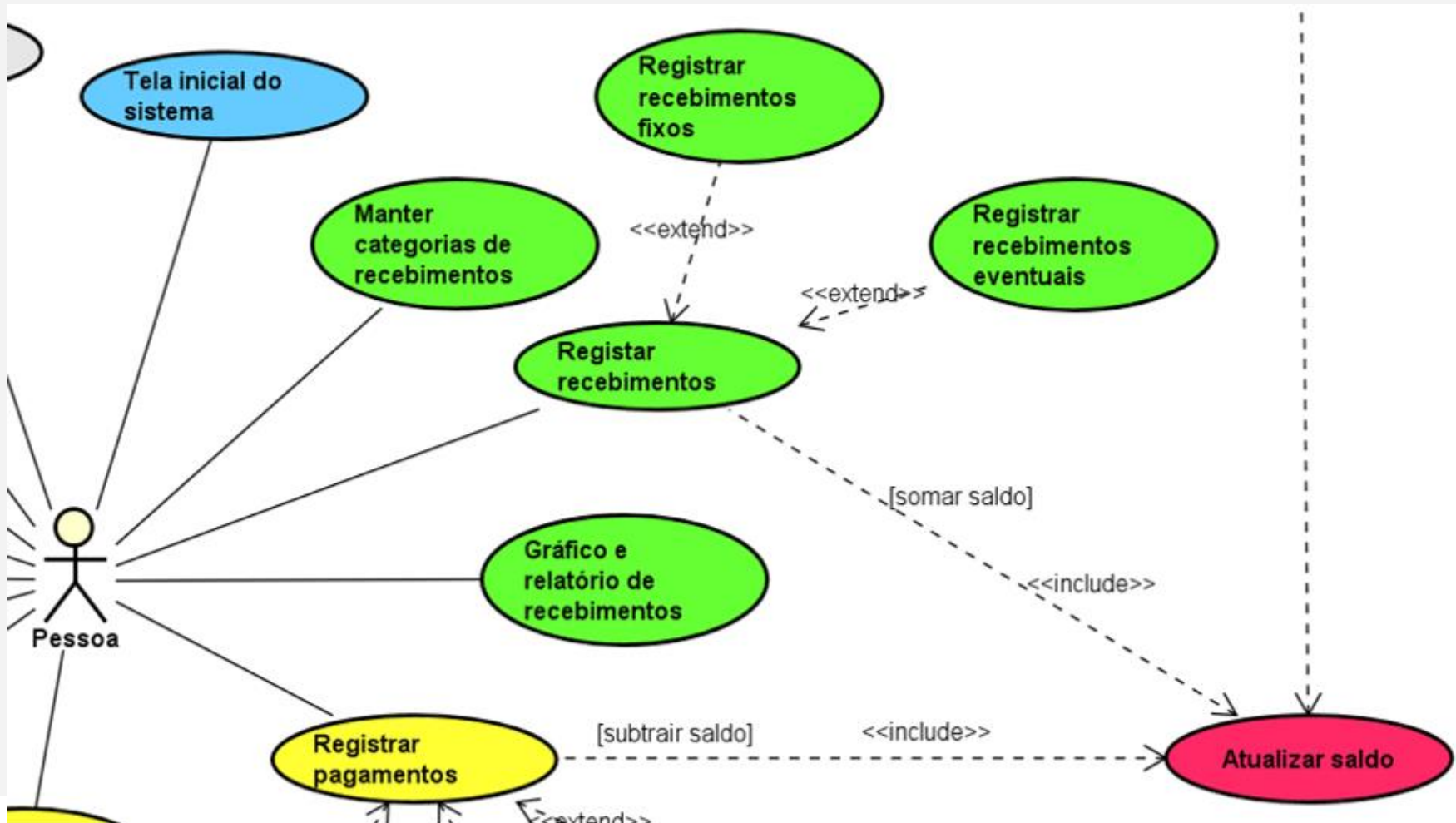
---

## 1 - Documentos de requisitos: Requisitos funcionais e não funcionais

| Req  | Tipo usuário | Requisito              | Descrição   |
|------|--------------|------------------------|---|
| RF05 | Cliente      | Registrar recebimentos | Registrar recebimentos de várias fontes de rendas e classificar por categoria (salário, bolsa de estudos, <u>estágios</u> , etc.) e somar o valor ao saldo da conta. Deve ter opção de registrar se é um recebimento fixo mensal ou extraordinário. |

# Artefatos necessários para serem analisados ao criar casos de testes

## 2 - Diagrama de casos de uso




# Artefatos necessários para serem analisados ao criar casos de testes

3 - Protótipo

4 - Modelo-ER ou Documento de Arquitetura de Software


Cadastro de recebimentos

Conta do usuário: <Nome do usuário> 

### Cadastro de Recebimentos

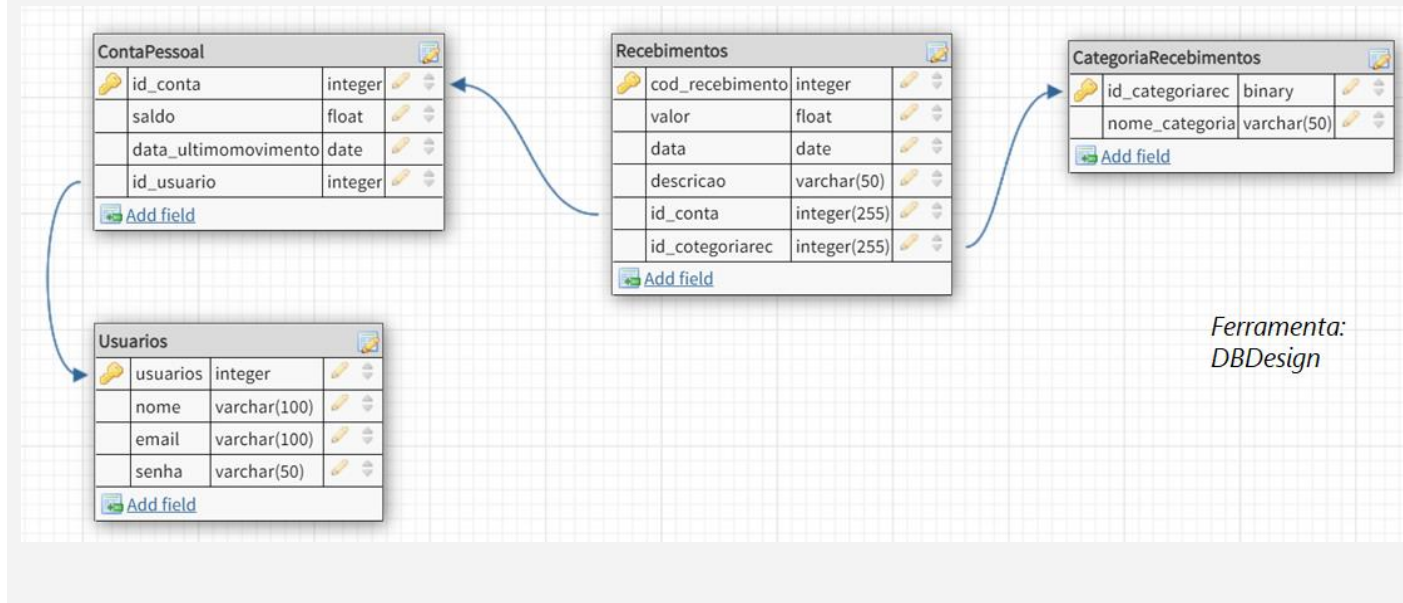
Descrição:

Valor:

Data:  

Categoria recebimento: 

Selecione a categoria:   
Salário  
Auxílio financeiro UFFS  
Auxílio financeiro familiar  
Free Lance





# Critérios para casos de testes funcionais

---

## Identificar bons casos de teste dependem de bons critérios a serem utilizados nos casos de testes

- **Particionamento em classes de equivalência:**

Dividir o conjunto de dados de entrada em classes de equivalência válidas e inválidas, permitindo que sejam escolhidos dados de teste representativos de cada classe.

- **Análise do valor limite (teste de limite inferior ou superior):**

Testa os limites das classes de equivalência, incluindo os valores nos limites inferiores e superiores

- **Gráfico de causa e efeito:**

Utiliza um diagrama de causa e efeito para identificar todas as possíveis causas de um problema específico, o que ajuda a garantir que todas as áreas relevantes sejam testadas

- **Transição de estado :**

Analisa o comportamento do sistema em diferentes estados, garantindo que todas as transições de estado sejam testadas

# Critérios para casos de testes funcionais

## Exemplos

---

- **Particionamento em classes de equivalência:**

- Exemplos: Venda de ingresso no cinema de acordo com a faixa etária, Acesso ao sistema por tipo de usuário

- **Análise do valor limite (teste de limite inferior ou superior):**

- Exemplos: Limite de cartão de crédito, Saldo do estoque

- **Gráfico de causa e efeito:**

- Exemplos: Aprovação ou reprovação do aluno numa disciplina  
Projeto dentro do prazo ou fora do prazo

- **Transição de estado :**

- Exemplo: Etapas de um pedido no comércio eletrônico: em análise, aprovado, separação, despacho, etc.

# Ambiente de testes

---

## **Ambiente de Desenvolvimento:**

Utilizado pelos desenvolvedores para escrever e testar o código antes de ser promovido para um ambiente de teste

## **Ambiente de Teste/QA:**

Um ambiente dedicado onde os testadores executam casos de teste  
Deve espelhar o ambiente de produção o mais próximo possível

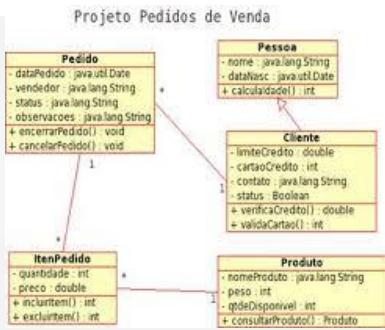
## **Ambiente de Pré-produção:**

Um ambiente que replica exatamente o ambiente de produção e é utilizado para testes finais antes do entrega final



Elicitação/  
Especificação  
de requisitos

Manutenção  
Evolução



Plano de testes

Projeto de software



Codificação



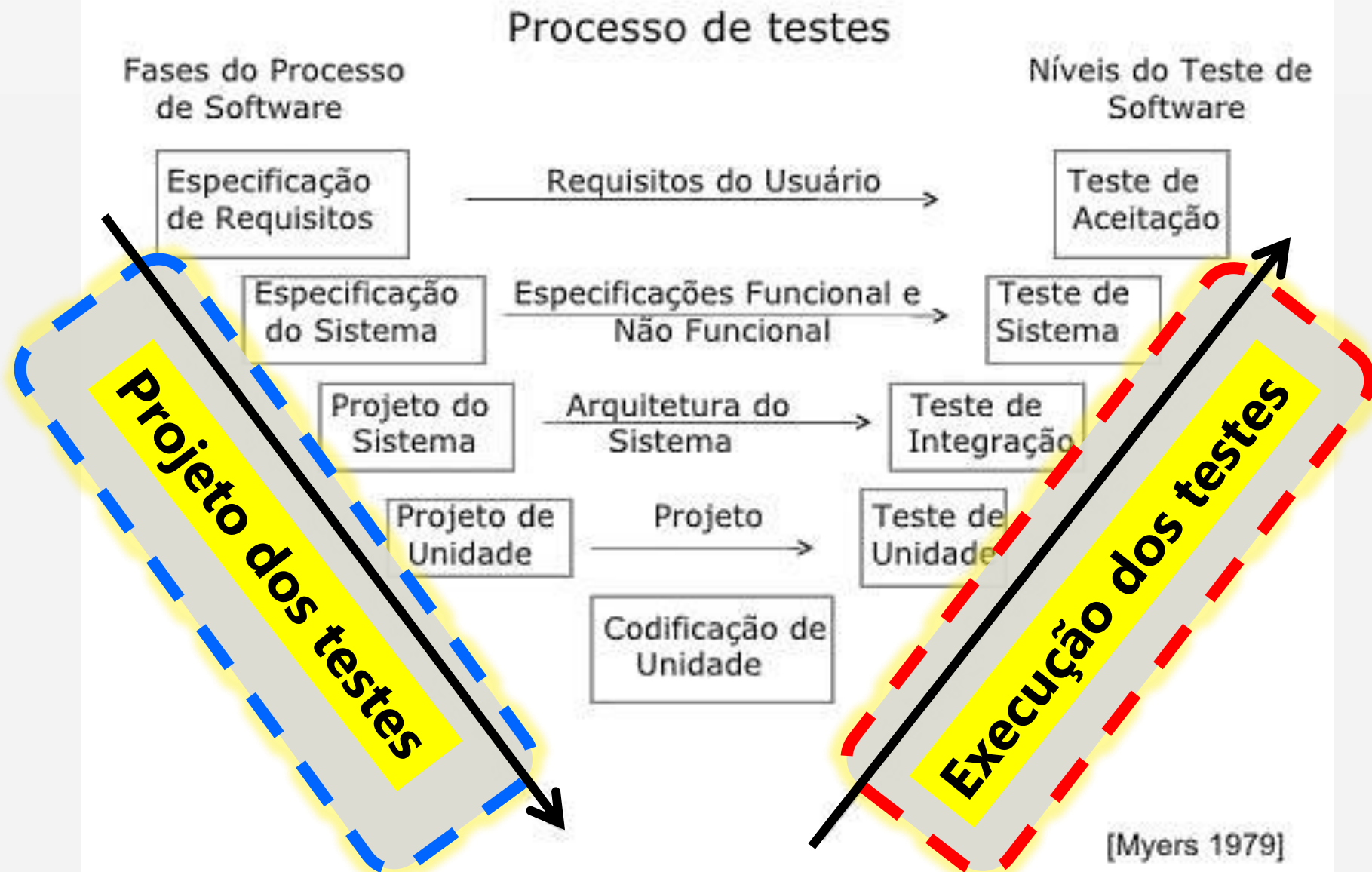
Implantação



Testes de software

GERÊNCIA DE PROJETOS

# PLANO DE TESTES



# Equipe de testes - (QA - Quality Assurance Garantia da Qualidade)

---

## **Testadores:**

Profissionais responsáveis pela execução dos casos de teste

Média Salarial: R\$ 3.000 a R\$ 6.000

## **Analistas de Teste:**

Especialistas que escrevem casos de teste com base nos requisitos e especificações.

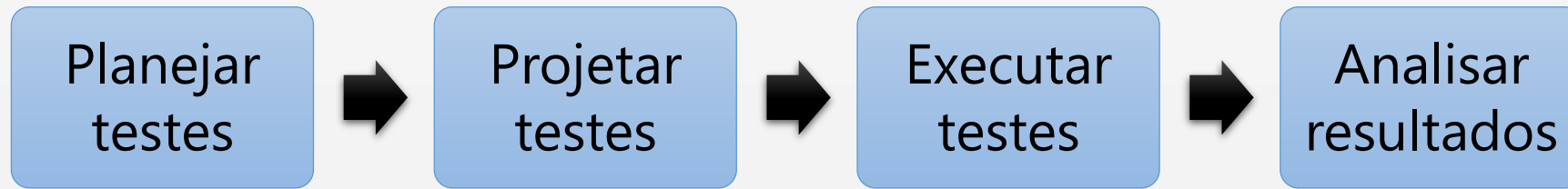
Média Salarial: R\$ 5.000 a R\$ 9.000

## **Líder de Teste/QA:**

Responsável por coordenar as atividades de teste, gerenciar a equipe e garantir a qualidade do processo de teste

Média Salarial: R\$ 8.000 a R\$ 15.000

# Processo de testes



Gerenciar  
defeitos

1º) **Planejamento:**

Elaboração do **plano de testes**

2º) **Especificação (projetar):**

Elaborar casos de testes

3º) **Execução:**

Executar os testes e registrar os resultados

4º) **Avaliação dos resultados:**

Avaliação da qualidade do software

# Relatórios e Análise

---

## 4º) Avaliação dos resultados:

### **Relatórios de Teste:**

Documentos que detalham os resultados dos testes, incluindo casos de teste executados, defeitos encontrados, status de correção e as métricas de qualidade

### **Análise de Teste:**

Revisões e análises dos resultados de teste para identificar padrões, tendências e áreas problemáticas.

**→ APRENDA COM OS TESTES**



# Engenharia de Software I

- Métricas de software

# Medição nas áreas de engenharia

---

## **Medição nas áreas de engenharia:**

- A medição é algo comum no mundo da engenharia, existindo vários tipos de grandezas para medição:
  - peso, temperatura, dimensões, tensão, entre outras.

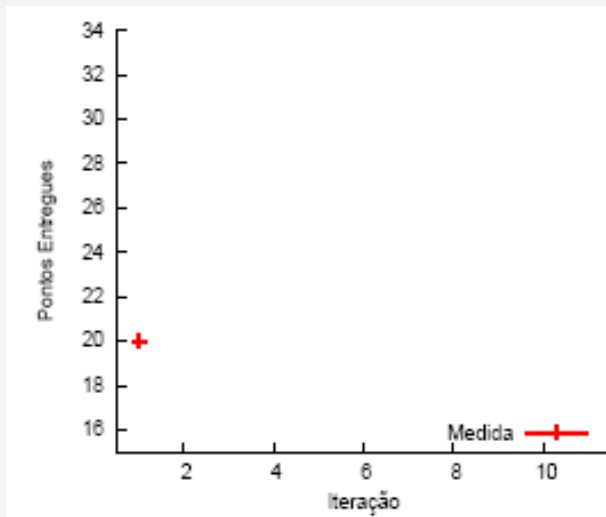
# Medição nas áreas de Engenharia de Software

---

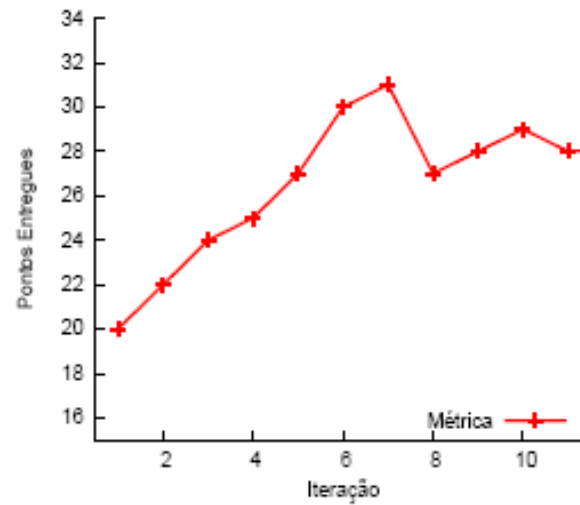
- Infelizmente a engenharia de software está longe de ter uma medição padrão amplamente aceita e sem nenhum fator subjetivo.
- Temos **dificuldade em concordar sobre o que medir e como avaliar o resultado** das medições obtidas.
- **Métricas de softwares** possibilitam realizar uma das atividades mais importantes do gerenciamento de projetos: o **planejamento**.

# Conceitos iniciais:

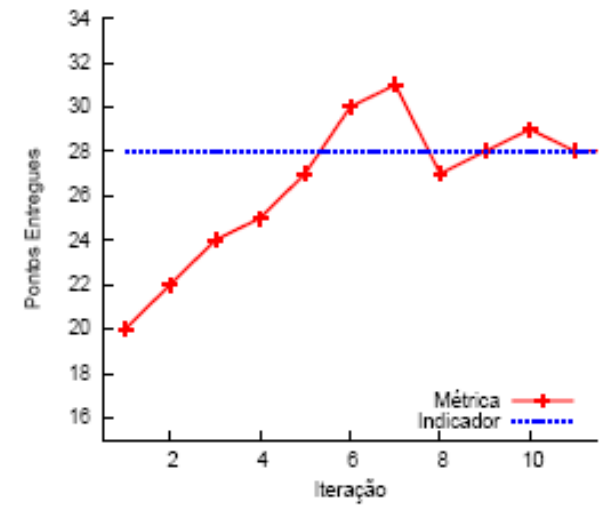
## Medida, métrica, medição e indicador



(a) Medida



(b) Métrica



(c) Indicador

# Conceitos iniciais

---

- **Medida:** fornece uma indicação quantitativa da extensão, quantidade, dimensão, capacidade ou tamanho de algum atributo de um produto ou processo.
- **Medição:** ato de determinação de uma medida.
- **Métrica:** medida quantitativa do grau em que um sistema se encontra em relação a um determinado atributo
- **Indicadores:** métrica ou combinação de métricas que fornece uma compreensão de um processo, projeto, ou produto.

# Medição de Software

---

- Provê informação útil para que as organizações tomem decisões que impactam em seus objetivos de negócios.
- A medição de software é um dos **principais pilares da melhoria de processos de software.**

# Métricas de software

---

- As análises baseadas em métricas são mais eficientes do que as que utilizam informações subjetivas.
- Utilizar dados históricos das métricas.

# Exemplos da definição formal métricas

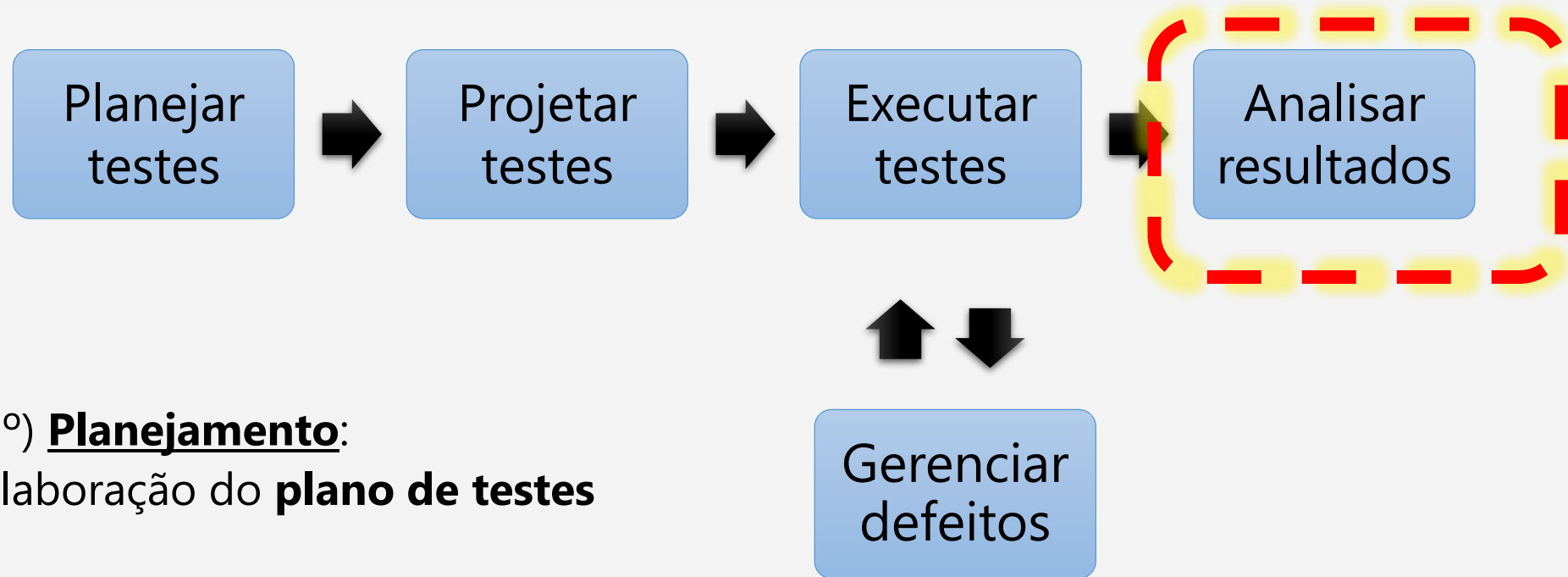
---

## **Lembrando da definição de Métrica:**

- É uma medida quantitativa (fórmula de cálculo) do grau em que um sistema se encontra em relação a um determinado atributo.
- Ou seja, é uma comparação de medias/valores em períodos de tempo diferentes.



# Processo de testes



1º) **Planejamento:**

Elaboração do **plano de testes**

2º) **Especificação (projetar):**

Elaborar casos de testes

3º) **Execução:**

Executar os testes e registrar os resultados

4º) **Avaliação dos resultados:**

Avaliação da qualidade do software

# Análise dos resultados → Métricas de testes

---

Exemplo de métricas relacionadas a testes:

- Cobertura de testes automatizados
- Defeitos em aberto (não solucionados)
- Defeitos encontrados na fase de testes
- Severidade média dos defeitos
- Defeitos encontrados pelo cliente
- Densidade dos defeitos (proporção de número de defeitos em relação ao tamanho do projeto)
- Índice de satisfação dos clientes
- NPS – Net Promoter Score
- Taxa de dívida técnica
- Retrabalho