

# Perguntas e Respostas

## 1. Por que Qualidade é um conceito relativo?

Porque o que constitui “bom” depende de quem avalia (usuário, cliente, desenvolvedor), do contexto de uso, dos requisitos implicados e das expectativas, logo o valor de qualidade varia segundo perspectiva e finalidade.

## 2. Cite e exemplifique 3 dimensões de qualidade.

- **Funcionalidade:** o software faz o que se espera (ex: um sistema bancário permite transferir valores).
- **Confiabilidade:** o software mantém funcionamento estável sob condições previstas (ex: não falha durante picos de uso).
- **Usabilidade:** o software é fácil de usar, aprender e operar (ex: interface intuitiva para novos usuários).

## 3. Por que é útil ter uma forma de mensurar qualidade?

Porque sem métricas não se sabe se os objetivos de qualidade foram atingidos, não se identifica onde melhorar, não se compara opções ou produtos, e não se toma decisões informadas para gerenciar riscos.

## 4. Por que é necessária uma certificação de qualidade?

Porque atesta que processos ou produtos seguem padrões reconhecidos, passa confiança a clientes interessados, garante consistência, transparência e redução de erros ou falhas, além de favorecer concorrência e conformidade.

## 5. Diferencie Produto de Processo.

- **Produto:** o resultado final ou item entregue (ex: software, módulo funcional).
- **Processo:** o conjunto de atividades, métodos e recursos usados para produzir o produto (ex: ciclo de desenvolvimento, revisão de código, testes).

**6. Como a ISO/IEC 9126 se subdivide?**

Ela se divide em quatro partes:

- Parte 1: Modelo de qualidade (características e subcaracterísticas)
- Parte 2: Métricas externas
- Parte 3: Métricas internas
- Parte 4: Métricas de qualidade em uso (“quality in use”)

**7. Considerando a ISO/IEC 9126, diferencie qualidade interna e qualidade externa.**

- **Qualidade interna:** refere-se a atributos do software em estado não executável ou sob análise estática (ex: estrutura do código-fonte).
- **Qualidade externa:** refere-se ao comportamento do software quando em execução, em seu ambiente de sistema (ex: resposta, falhas, interoperabilidade).

**8. Diferencie Qualidade de Software de Qualidade de processo de software.**

- **Qualidade de software:** refere-se ao produto de software — suas funcionalidades, desempenho, confiabilidade, manutenibilidade etc.
- **Qualidade de processo de software:** refere-se à forma como o software foi desenvolvido — organização, métodos, controle, repetibilidade — e como isso influencia no produto final.

**9. Cite e explique sucintamente dois “mandamentos do software imaturo” que você já utilizou sem saber durante o curso ou**

**no seu trabalho.**

Dois exemplos:

- “Não utilizar uma equipe de teste independente”: assume-se que os programadores testam seu próprio código, o que reduz esforço de QA mas pode comprometer imparcialidade e cobertura.
- “Afastar o cliente do processo de desenvolvimento”: o cliente não participa ativamente após os requisitos iniciais, o que pode levar a resultados desalinhados com suas reais necessidades.

**10. Considerando os níveis de capacitação SPICE, diferencie “in-completo” de “executado”.**

No modelo ISO/IEC 15504 (ou SPICE) temos:

- Nível 0 (“Incompleto”): o processo não é implementado ou falha em atingir seu propósito.
- Nível 1 (“Executado” ou “Performed”): o processo é implementado e atinge seu propósito, mas de forma não sistemática ou padronizada.

**11. O que é o PDCA?**

O PDCA (Plan-Do-Check-Act) é um ciclo de melhoria contínua:

- **Plan:** planejar.
- **Do:** executar.
- **Check:** verificar/checar os resultados.
- **Act:** agir/corrigir para promover melhorias no processo ou produto.