

# Questionário sobre ISO/IEC 25010

1. O que significa compatibilidade em software segundo a ISO/IEC 25010?
2. O que são containers? O que seu uso (por exemplo, Docker) pode favorecer a portabilidade de aplicações?
3. Como práticas como profiling e otimização de código contribuem para a eficiência de desempenho?
4. Quais técnicas podem ser aplicadas no desenvolvimento para aumentar a confiabilidade de um sistema?
5. Sistemas que armazenam dados sensíveis, como informações de saúde, devem atender a altos requisitos de segurança. Quais práticas você indicaria para garantir isso?
6. Cite e explique duas situações onde a falta de compatibilidade pode comprometer um sistema.
7. Qual é a diferença entre comportamento temporal e utilização de recursos?
8. Defina portabilidade em software e explique sua importância no contexto atual de desenvolvimento multiplataforma.
9. Um software desenvolvido para funcionar apenas no Windows 10 falha ao ser executado no Windows 11. Qual subcaracterística de qualidade foi afetada? Justifique.
10. Qual a diferença entre maturidade e tolerância a falhas na subcaracterística de confiabilidade?
11. Explique a diferença entre instalabilidade e substituíbilidade no contexto de portabilidade.
12. Quais são as três principais propriedades da segurança da informação que impactam a qualidade de software? Explique cada uma.
13. Como práticas de desenvolvimento podem favorecer a compatibilidade entre diferentes sistemas? Dê dois exemplos.
14. Imagine um sistema bancário que frequentemente fica indisponível em dias de pico. Qual subcaracterística está comprometida?
15. O que significa eficiência de desempenho no contexto da ISO/IEC 25010?
16. Por que a ausência de segurança é considerada uma das falhas mais críticas em sistemas? Dê um exemplo real ou hipotético.

17. Defina confiabilidade no contexto da qualidade de software.
18. Explique a importância de validar entradas do usuário como uma prática de segurança no desenvolvimento de software.
19. O que significa adaptabilidade dentro da portabilidade? Dê um exemplo.
20. Cite dois exemplos de sistemas que exigem altíssima eficiência de desempenho e explique por quê.

## Estudo de caso: Caso 737 MAX

Entre 2018 e 2019, dois acidentes fatais com o Boeing 737 MAX causaram a morte de 346 pessoas. O sistema automatizado MCAS (Maneuvering Characteristics Augmentation System) foi considerado um dos principais responsáveis. Ele atuava automaticamente para corrigir o ângulo de ataque da aeronave, mas baseava-se na leitura de apenas um sensor.

Esse sensor, ao falhar, levou o sistema a realizar correções indevidas, colocando a aeronave em risco. Além disso, os pilotos não foram devidamente informados sobre a existência do MCAS e sobre como desativá-lo. O software foi projetado sem redundância suficiente, com testes limitados e sob forte pressão para apressar a certificação, visando competir com a Airbus.

As falhas deste caso se relacionam com diversas subcaracterísticas da ISO/IEC 25010:

- Confiabilidade: o sistema falhou ao não tolerar erros de sensor.
- Segurança: ausência de práticas robustas para evitar falhas críticas.
- Compatibilidade: integração insuficiente entre novos sistemas e procedimentos existentes.
- Usabilidade: falta de treinamento e informações adequadas aos pilotos.
- Manutenibilidade: dificuldades para alterar rapidamente o sistema diante de problemas identificados.

O caso resultou em paralisação global das aeronaves Boeing 737 MAX, prejuízos financeiros bilionários e danos severos à reputação da empresa.

### Perguntas sobre o Estudo de Caso:

1. Quais critérios da ISO/IEC 25010 foram mais afetadas no caso do Boeing 737 MAX?
2. De que maneira a falta de redundância contribuiu para o desastre?
3. O que poderia ter sido feito no projeto do MCAS para aumentar a confiabilidade do sistema?

4. Considerando as pressões comerciais, como o equilíbrio entre agilidade e qualidade poderia ter sido melhor gerenciado?
5. Quais lições esse caso traz para desenvolvedores que atuam em sistemas críticos?