



Verificação e Validação de Software

Caroline Lewandowski - 20102626

Turma 31 - 17 mar 2022

### **Questionário - Teste e Análise de Software:**

**1. Todo desenvolvimento de produto exige atividades de verificação para aferir a qualidade?**

Sim, a verificação indica o quanto o produto satisfaz o padrão de qualidade, passando por testes específicos dependendo das especificações do item.

**2. As atividades de verificação são sempre as mesmas para produtos desenvolvidos em série? Por quê?**

Não, apesar de existirem verificações padrões para itens não-críticos, em alguns casos certos produtos produzidos em série necessitam da verificação individual de componentes cruciais como, por exemplo, telas e placas de circuito.

Há também produtos produzidos em série que precisam ser verificados individualmente antes da entrega aos consumidores, exigindo projetos sofisticados e procedimentos avançados de verificação por possuírem processos complexos ou requisitos de qualidade rígidos.

**3. Qual a diferença entre a verificação de produtos desenvolvidos em série para produtos únicos?**

A verificação de produtos desenvolvidos em série consistem em repetir um conjunto predefinido de teste e análises que indicam se os produtos satisfazem os padrões de qualidade.

Em contraste, verificações de produtos únicos requerem a elaboração de um conjunto específico de testes e análises, além disso, a relação entre os resultados dos testes e análises e a qualidade deve ser avaliada para cada produto.

#### **4. Como se caracterizam os produtos de software neste contexto de teste e análise?**

Produtos de Software não são feitos em série, sendo eles parcialmente únicos em seu projeto e funcionalidade. Portanto, devem ser verificados individualmente, tanto durante como depois de sua produção, a fim de identificar e eliminar erros.

#### **5. Qual o percentual do custo da verificação de software frente ao custo total do produto?**

O custo da verificação de Software frequentemente corresponde a mais de 50% do custo total do desenvolvimento e manutenção.

#### **6. Técnicas avançadas de desenvolvimento impactam na verificação?**

Podem reduzir a frequência de algumas classes de erros, em muitos casos também podem induzir a novos tipos de erros, podendo esses serem mais difíceis de revelar e remover.

#### **7. Quando se começa e termina o processo de verificação e validação?**

A verificação e validação começa assim que é decidido construir um produto de Software, e, mesmo após completar o projeto o software continua evoluindo, necessitando dos processos de verificação e validação a cada mudança do sistema.

#### **8. Todo o produto de software passa, obrigatoriamente, pelas mesmas etapas de teste e análise?**

Não, técnicas de teste e análise variam de acordo com as decisões preliminares definidas no plano de construção inicial para o desenvolvimento.

#### **9. Quais técnicas devem ser aplicadas?**

A escolha das técnicas de teste e análise dependem das restrições de qualidade, custo, prazo e recurso de desenvolvimento do produto em particular.

#### **10. Como se pode verificar se um produto está pronto?**

Um produto deve ser entregue quando atinge um nível adequado de acordo com as diferentes métricas de dependabilidade apropriadas para o contexto. Sendo elas:

**Disponibilidade**, que define a qualidade de serviço em termos de tempo de serviço versus tempo fora. **Qualidade**, representando o tempo médio entre falhas e, por sua vez, **Confiabilidade**, para o percentual de operações completadas com sucesso.

### **11. O que vem a ser teste estatístico? Para que serve? Qual sua relação com perfis operacionais?**

Teste usado para medir confiabilidade do programa. Os perfis operacionais podem ser utilizados para gerar grandes conjuntos estatisticamente válidos de testes gerados randomicamente, podendo estimar a confiabilidade a partir da taxa de falha presente nas amostras geradas.

### **12. O que são testes alfa e beta? Qual sua relação com testes estatísticos?**

*Alfa Testes* são testes executados pelos usuários em um ambiente controlado, observados pela equipe de desenvolvimento.

*Beta Testes* consistem em execuções de tarefas reais pelo usuário em seu próprio ambiente, sem interferência ou monitoração própria.

Os testes acima se referem à segunda abordagem para medir a confiabilidade do programa, recorrido quando não é possível construir e validar um modelo com detalhes suficientes para obter estimativas de confiabilidade após a utilização de testes estatísticos.

### **13. Como se garante o sucesso de versões sucessivas?**

O autor propõe a construção de *point releases* e *patches*: Versões relativamente maiores são chamadas de *point releases*, onde o processo da qualidade completo é repetido em miniatura para cada versão pontual. Revisões a nível de *patch*, por sua vez, são revisões menores onde seu teste e análise são abreviados.

Antes da liberação, cada versão de ponta deve se submeter a um teste de regressão, já revisões menores podem prosseguir com a execução de um subconjunto de casos de teste. À medida que novos erros são descobertos estes são acrescentados a novos casos de teste de regressão.

#### **14. Como o processo de desenvolvimento pode ser melhorado?**

É sugerido um programa de melhoria de qualidade para rastrear e classificar erros, identificando falhas humanas e fraquezas no teste e análise que permitiram que o erro permanecesse indetectado. O processo de análise de erros e melhoria de processo compreende quatro fases:

1. Definir dados e implementação de procedimentos de coleta.
2. Analisar dados - onde ocorre a identificação de erros importantes.
3. Analisar erros.
4. Ajustar processo de qualidade e desenvolvimento.

Os dados coletados, como no exemplo, são integrados ao sistema e com o tempo são agregados e classificados a fim de identificar classes de falhas importantes por ocorrerem com frequência, por fim, as falhas são analisadas para compreender como são introduzidas e os motivos de escaparem da detecção.

#### **15. Quais as vantagens do uso de uma arquitetura em camadas frente às etapas de teste?**

A divisão do sistema em camadas facilita o trabalho da equipe de teste e outros desenvolvedores, permitindo também uma melhor alocação dos requisitos de qualidade, facilitando o projeto detalhado e o teste.