



**UNIPAR – Universidade Paranaense**

**Testes de Software**

Prof.: Paulo H. P. Santos

Aluno: Bruno Manoel Toebe de Almeida RA: 233397-1

Aluno: Diogo Massayuki Kumizaki RA: 238909-3

### ● **Teste de Integração:**

O teste de integração tem como objetivo verificar se a comunicação entre diferentes módulos do sistema ocorre de maneira correta. Enquanto o teste unitário foca em testar módulos de forma isolada, o teste de integração agrupa esses módulos e avalia a interação entre eles para identificar possíveis falhas nas interfaces. Durante o processo, inicialmente são realizados testes individuais nos módulos. Posteriormente, os módulos são integrados em grupos, permitindo a identificação de erros de interface e a medição do desempenho e confiabilidade dessas interações. Apenas após testar todos os grupos, o sistema como um todo é avaliado no teste de sistema final.

O teste de integração são essenciais para:

- **Desenvolvimento de uma aplicação web:** Quando diferentes partes do sistema, como o front-end e o back-end, precisam interagir para funcionar corretamente. O teste de integração verifica se a interface entre essas partes está funcionando corretamente.
- **Sistemas distribuídos ou microserviços:** Em um ambiente onde diferentes serviços independentes devem se comunicar, o teste de integração assegura que as APIs estejam se conectando de forma confiável e trocando dados corretamente.
- **Integração com banco de dados:** Testa-se se os módulos de negócio estão interagindo corretamente com o banco de dados, verificando se as operações de leitura e escrita são realizadas conforme o esperado.

- **Teste de Validação:**

O teste de validação é um processo de garantia de qualidade que visa garantir que o software desenvolvido atende às necessidades e expectativas dos usuários finais, conforme os requisitos definidos no início do projeto. O foco da validação é assegurar que o produto entregue é o *produto certo*, ou seja, que ele resolve o problema que se propôs a resolver, oferecendo valor para o usuário. Isso inclui verificar se as funcionalidades estão em conformidade com os requisitos e se a usabilidade atende às expectativas dos usuários. Enquanto o teste de verificação examina se o software foi construído corretamente (de acordo com as especificações técnicas), o teste de validação foca em garantir que o software atende ao propósito esperado.

#### Aplicação em um Projeto Real

Em um sistema de e-commerce, por exemplo, o teste de validação poderia ser aplicado para garantir que todo o fluxo de compra, desde a seleção de produtos até o pagamento, funciona corretamente do ponto de vista do usuário. Um cenário típico seria validar que o cliente consegue adicionar produtos ao carrinho, aplicar cupons de desconto, calcular corretamente os valores de frete e finalizar o pagamento sem falhas. Além disso, é possível validar a experiência do usuário final, certificando-se de que a interface é intuitiva e fácil de usar, atendendo ao público-alvo da plataforma.

Dessa forma, o teste de validação garante que o software não apenas funciona tecnicamente, mas também atende ao que o cliente realmente precisa.

- **Teste de Sistema:**

O teste de sistema é realizado após todas as integrações estarem concluídas, quando o sistema está completo e pronto para ser avaliado em um ambiente que simula o uso final. Esse tipo de teste é focado em identificar falhas que o usuário final pode encontrar, avaliando o software de uma perspectiva externa, sem a necessidade de entender a lógica interna ou o código. O objetivo principal é verificar a funcionalidade geral do sistema, incluindo aspectos como a qualidade da interface gráfica, o funcionamento correto das funções e se as precondições para as operações do sistema estão sendo atendidas. Esse teste garante que o

software atenda aos requisitos funcionais e de qualidade estabelecidos para a entrega final.

Cenários onde o teste de sistema é utilizado:

- Lançamento de um novo software de gestão empresarial: Um teste de sistema é crucial para garantir que todas as funcionalidades, como emissão de relatórios e controle de estoque, estejam funcionando corretamente e de maneira integrada, simulando o uso real por empresas.
- Aplicativo de mobile banking: Antes de ser liberado para os usuários finais, o sistema é testado para garantir que todas as operações financeiras, como transferências e pagamentos, sejam realizadas de forma precisa e sem falhas, verificando tanto o back-end quanto a interface de usuário.
- Sistemas embarcados em veículos: O teste de sistema avalia o comportamento completo de um software que gerencia funcionalidades críticas, como freios ABS e controle de tração, assegurando que todas as interações entre os subsistemas funcionem corretamente em situações reais.

## ● Teste de Recuperação:

O teste de recuperação é um tipo de teste que avalia a habilidade de um sistema de restaurar seu funcionamento normal após sofrer uma falha inesperada, como interrupções de energia, falhas de hardware ou problemas de rede. O objetivo é garantir que, após uma falha, o sistema consiga retornar a um estado operacional estável, sem perda de dados ou com perda mínima. Esse tipo de teste é essencial em sistemas críticos, como bancos de dados financeiros ou serviços de saúde, onde a disponibilidade e a integridade dos dados são vitais.

Durante o teste de recuperação, falhas são deliberadamente induzidas para avaliar a rapidez com que o sistema consegue se recuperar e retomar suas operações. Além disso, verifica-se se os mecanismos de backup e restauração de dados funcionam adequadamente.

Exemplos de Aplicação em Sistemas Críticos

1. Sistemas de Hospitais: Um sistema de gerenciamento de informações médicas precisa garantir que, após uma falha, os dados dos pacientes não sejam

perdidos e que o sistema volte a funcionar em um curto período de tempo, para que o atendimento não seja interrompido.

2. Bancos: Sistemas bancários precisam garantir que, após uma falha, como uma queda de energia ou problema de rede, as transações em andamento não sejam perdidas e que os dados sejam recuperados com precisão.

Esse teste é importante em qualquer sistema onde a continuidade dos serviços e a integridade dos dados sejam essenciais para o negócio.

## ● **Teste de Aceitação:**

O teste de aceitação é uma etapa no processo de testes de software, cujo objetivo é garantir que o sistema atenda às necessidades e expectativas dos usuários finais. Ele é semelhante ao teste de sistema, mas o foco está na experiência e na satisfação do usuário com o software. Segundo Graham et al. (2007) e a IBM, esse teste verifica se o público final aceita o sistema para as funções que ele deve realizar. O teste de aceitação é muitas vezes realizado com grupos específicos de usuários, como em versões beta, onde os feedbacks dos usuários ajudam a ajustar o software antes do lançamento final.

Exemplos de critérios de aceitação:

1. Funcionalidade: O software deve executar todas as tarefas conforme especificado.
2. Usabilidade: A interface deve ser intuitiva e fácil de usar.
3. Desempenho: O tempo de resposta do sistema deve ser aceitável.
4. Segurança: O sistema deve proteger os dados dos usuários adequadamente.
5. Conformidade com regras de negócio: O sistema deve respeitar as políticas estabelecidas.

## Referências

FÉLIX, Rafael (org.). **Teste de software**. São Paulo: Pearson, 2016. *E-book*. Disponível em: <https://plataforma.bvirtual.com.br>. Acesso em: 18 set. 2024.

SOMMERVILLE, Ian. **Engenharia de software**. 10. ed. São Paulo, SP: Pearson, 2018. *E-book*. Disponível em: <https://plataforma.bvirtual.com.br>. Acesso em: 18 set. 2024.

POLO, Rodrigo Cantú. **Validação e teste de software**. 1. ed. São Paulo: Contentus, 2020. *E-book*. Disponível em: <https://plataforma.bvirtual.com.br>. Acesso em: 18 set. 2024.