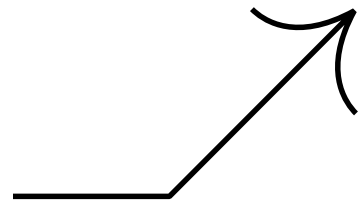


# Termos que todo QA precisa conhecer



---

*Já se sentiu perdido com tantas siglas e termos técnicos?  
Calma, todo QA já passou por isso.*

*Este guia foi feito pra te ajudar a entender os conceitos mais usados no dia a dia dos testes.*

***Por Karen.K***

***Porque entender o vocabulário é o primeiro passo para falar a língua da qualidade!***

## Fundamentos do QA

1. **Bug / Defeito** → Falha no sistema que faz o software se comportar de forma incorreta.
2. **Erro** → Engano humano que gera um defeito (geralmente de lógica ou digitação).
3. **Falha** → Quando o defeito realmente aparece durante a execução do software.
4. **Validação** → Garante que o produto faz o que o usuário espera.
5. **Verificação** → Garante que o produto foi construído corretamente de acordo com as especificações.
6. **Qualidade** → Não é ausência de defeitos, e sim **atendimento às expectativas do cliente**.
7. **Teste de Software** → Processo de avaliar e melhorar a qualidade de um sistema.

## Documentação e Planejamento

8. **Plano de Teste (Test Plan)** → Documento que descreve o escopo, estratégia e recursos do teste.
9. **Caso de Teste (Test Case)** → Passo a passo para verificar uma funcionalidade específica.
10. **Cenário de Teste (Test Scenario)** → Situação ou fluxo completo a ser testado.
11. **Roteiro de Teste (Test Script)** → Sequência detalhada de passos para executar um teste.

- 12. **Test Suite** → Conjunto de casos de teste agrupados por funcionalidade.
- 13. **Matriz de Rastreabilidade** → Relaciona requisitos com casos de teste, garantindo cobertura.
- 14. **Plano de Regressão** → Conjunto de testes executados após uma atualização para garantir estabilidade.

## *Ambiente e Execução*

- 15. **Ambiente de Teste** → Onde o QA realiza os testes, separado do ambiente de produção.
- 16. **Build** → Versão compilada do software que será testada.
- 17. **Deploy / Implantação** → Ato de disponibilizar uma versão do software em um ambiente.
- 18. **Smoke Test** → Teste rápido pra validar se o sistema está “respirando” após um deploy.
- 19. **Sanity Test** → Verifica se as funcionalidades específicas corrigidas continuam funcionando.
- 20. **Test Data (Dados de Teste)** → Informações usadas para executar os casos de teste.

## *Termos de API e Automação*

- 21. **API (Application Programming Interface)** → Conjunto de regras que permite sistemas se comunicarem.
- 22. **Endpoint** → URL que representa um serviço da API (ex: /user/login).
- 23. **Request (Requisição)** → O que o QA envia para a API.
- 24. **Response (Resposta)** → O que a API devolve.
- 25. **Status Code** → Código que indica o resultado da requisição (ex: 200 OK, 404 Not Found).
- 26. **JSON / XML** → Formatos de troca de dados usados em APIs.
- 27. **Automação de Testes** → Execução automática de testes repetitivos usando scripts.

## *Estratégia e Análise*

- 28. **Teste Exploratório** → Teste sem roteiro fixo, guiado pela curiosidade e experiência.
- 29. **Teste Baseado em Risco** → Prioriza áreas do sistema onde a falha seria mais crítica.
- 30. **Teste de Regressão** → Garante que algo que já funcionava continua ok após mudanças.
- 31. **Teste de Integração** → Verifica se diferentes partes do sistema funcionam bem juntas.
- 32. **Teste de Unidade** → Valida pequenas partes isoladas do código (geralmente feito por devs).
- 33. **Teste Funcional** → Valida o comportamento do sistema conforme os requisitos.

34. **Teste Não Funcional** → Mede desempenho, segurança, usabilidade, etc.

35. **Teste de Aceitação (UAT)** → Feito pelo cliente/usuário para validar se o produto atende às necessidades reais.

## **Classificação e Gestão de Defeitos**

36. **Severidade** → O quão grave é o defeito tecnicamente.

37. **Prioridade** → O quão urgente é corrigir o defeito.

38. **Bug Report** → Documento que registra um defeito com descrição, passos e evidências.

39. **Repro Steps** → Passos para reproduzir o defeito.

40. **Expected Result x Actual Result** → Resultado esperado x resultado obtido.

## **Gestão e Métricas**

41. **Cobertura de Testes (Test Coverage)** → Percentual do sistema validado por testes.

42. **Defect Density** → Quantidade de defeitos encontrados por módulo/tamanho do código.

43. **Release** → Entrega de uma nova versão do produto.

44. **Sprint** → Ciclo de desenvolvimento em metodologias ágeis.

45. **Backlog** → Lista de tarefas ou funcionalidades pendentes.

46. **Retrospectiva** → Reunião de melhoria contínua ao fim de uma sprint.

## **Comunicação e Mentalidade**

47. **QA Mindset** → Mentalidade de prevenção, clareza e questionamento constante.

48. **Shift Left** → Envolver QA desde o início do desenvolvimento.

49. **Feedback Loop** → Ciclo de aprendizado contínuo entre time de QA, Dev e PO.

50. **Quality Culture** → Cultura onde todos são responsáveis pela qualidade, não só o QA.

---

## **Boas Práticas e Dicas**

Como aprender todos esses termos com naturalidade:

- Participe das reuniões de refinamento e ouça os devs e POs.
- Monte um glossário pessoal com exemplos reais.
- Revise o vocabulário sempre que usar uma nova ferramenta.
- E lembre-se: **não saber é normal — perguntar faz parte da qualidade.**