

Ruptura < Resumo - Curso Início Rápido em Teste e QA

Data [↗](#)

Mar 25, 2025 a Mar 27, 2025

Participantes [↗](#)

- @Adony Lagares
- @Cassia Yumi
- @Thais Nogueira
- @Raique Alfredo Pereira de Ramos
- @Luis Magris de Sousa

Objetivo do curso [↗](#)

- O curso descreve de maneira completa a profissão de Analista de Teste/QA e começando desde o fundamento e apresentando a profissão, seus pilares, conceitos chaves e exemplos de atuação.

História do Teste [↗](#)

@Luis Magris de Sousa

- Em meados do século XIX Charles Babbage criou a primeira máquina automática de somar.
- Ada Lovelace contribuiu com Charles Babbage, utilizando o sistema de tiras e furos desenvolvido por ele e por muitos é considerada a primeira programadora. Também foi ela que encontrou o primeiro defeito.
- Ainda no século XIX, Herman Hollerith, baseado na máquina de Babbage, desenvolveu uma máquina para realizar o censo demográfico do EUA, o que há época poupou bastante tempo. Ele também foi responsável pela criação dos cartões perfurados sendo a forma de representação de dados mais utilizada até os anos 70. Hollerith mais tarde seria um dos fundadores da IBM.
- Durante a Segunda Guerra Mundial, Alan Turing desenvolveu uma máquina eletromecânica responsável por quebrar a criptografia da Enigma, máquina utilizada para comunicação das forças alemãs.
- Em 1947, Grace Hopper descreveu o primeiro "bug", ao encontrar um inseto que estava causando um defeito na máquina.
- Em 1979, foi publicado por Glenford Myers, A Arte do Teste de Software, por muitos considerada a bíblia do teste de software. Trazendo conceitos importantes utilizados até hoje. Myers também culminou a regra de 10, em que projeta que um erro não corrigido pode custar 10 vezes mais após um período de tempo e continuar a trazer prejuízos exponenciais se negligenciado.

Conceitos fundamentais [↗](#)

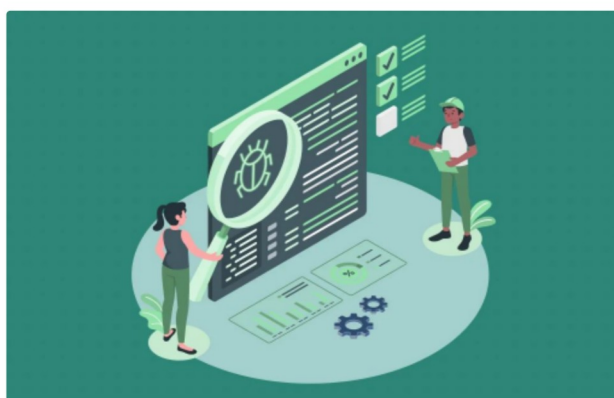
@Thais Nogueira

- O que é um teste de software e quais os pontos chaves na atuação de um testador/QA;
- Todos, dentro de seus próprios conhecimentos e perspectivas podem testar um software;
- Viés de confirmação;
- A importância do estudo na área da tecnologia (débito técnico);
- A importância do trabalho em equipe e o papel do QA na Squad;
- Diferença entre teste (voltado para produto) e QA (voltado para processo);
- Diferença entre erro (engano/encontrado pela própria pessoa), defeito (bugs/encontrado por outra pessoa) e falha (encontrado quando o código executado).

✓ Comportamento esperado de um QA/Testador [↗](#)

@Cassia Yumi

- Colaboração: fazer junto, saber ajudar, saber receber ajuda, dar feedback, receber feedback;
- Empatia: entender as necessidades, desejos, dores, restrições, preferências, hábitos e costumes de outra pessoa;
- Comunicar-se de maneira neutra: promover uma orientação e não criticar.
 - Ter cuidado com a comunicação verbal e não verbal.
- Responsabilidade compartilhada;
- Sempre buscar desenvolver habilidades pessoais e interpessoais (soft skills, hard skills);
- Separar tempo para se atualizar sobre novas tecnologias, tendências, etc;
- Ter autogerenciamento: identificar e classificar as tarefas, negar tarefas ou delegá-las, identificar a melhor sequência para execução das tarefas;
- O MVQ (Mínimo viável com Qualidade) deve ser sempre priorizado, mesmo com entregas rápidas;
- Em momentos críticos, a QArentena (pausa estratégica para revisão e validação antes da entrega final) é crucial;
- Mesmo sob pressão, um QA precisa manter o foco em garantir a excelência do produto entregue;
- Buscar acordos onde todos saem ganhando;



▶ Produtividade e fluxo contínuo [↗](#)

- Produtividade é sobre criar hábitos que maximizem o rendimento do tempo, eliminando distrações e estabelecendo rotinas.
- Dez hábitos da produtividade:
 1. Viver o hoje : Focar no presente, valorizando cada momento e priorizando as ações do agora.
 2. Se exercitar: Saúde do corpo e mente.
 3. Cultivar rituais saudáveis: Alimentação balanceada e garantir um sono de qualidade.
 4. Afastar-se de pessoas negativas: Evitar influências que drenam energia e comprometem o foco.
 5. Conviver com pessoas do bem: Estar próximo de indivíduos que inspiram, apoiam e motivam.
 6. Aprender com os erros: Encarar os erros como oportunidades de crescimento.
 7. Formalizar seus sonhos: Definir metas e registrar objetivo.
 8. Praticar a gratidão: Valorizar as conquistas e reconhecer as pequenas vitórias.
 9. Curtir a jornada: Aproveitar o caminho, celebrando o processo e não apenas o resultado final.
 10. Foco na solução: Direcionar a energia para encontrar respostas e resolver problemas.
- Fluxo contínuo é focar em manter um ritmo constante de trabalho sem desgaste, permitindo trabalhar no seu nível de esforço ideal, com alta auto performance e engajamento.
- Quanto mais se pratica, maior a produtividade e o engajamento. O fluxo contínuo permite que se trabalhe menos, mas com maior rendimento.
- Evita falhas de planejamento que possam sobrecarregar , mantendo um fluxo de rotina aceitável e sustentável.

🏆 Importância do Teste na entrega do produto final [↗](#)

@Thais Nogueira

- Essencial para garantir a qualidade e evitar impactos negativos.
- Importante para preservar a receita e a imagem dos envolvidos.
- Principais danos ocasionados pelos bugs :
 - Empresas e organizações: Provocam atrasos na entrega, insatisfação dos clientes e quebra de confiança ,resultando em perdas de vendas, e reputação da empresa.
 - Pessoas: Acidentes e falhas críticas impactam diretamente a segurança dos usuários. Podem causar constrangimentos, supressão de direitos e até colocar vidas em risco.
 - Governos: Lidam com dados sensíveis. Falhas em sistemas governamentais podem levar ao roubo e manipulação de dados sensíveis, prejudicando a comunicação e a segurança nacional.
 - Meio Ambiente: Erros podem atrasar alertas e comprometer o gerenciamento adequado dos recursos. Isso pode resultar em desperdício, poluição e outros danos ambientais devido a falhas no funcionamento dos sistemas.

🔧 Fundamentos, Técnicas e Ferramentas de QA [↗](#)

@Raique Alfredo Pereira de Ramos

- Burndown e gestão de sprint.
- **7 Fundamentos do Testes (ISTQB).** Conceitos que determinam o que o teste é capaz ou incapaz de realizar:
 - 1º Fundamento: Testes demonstram a presença de defeitos, não sua ausência.
 - 2º Fundamento: Teste exaustivo é impossível.
 - 3º Fundamento: Testes antecipados economizam tempo e dinheiro
 - 4º Fundamento: Agrupamento de defeitos.
 - Poucos módulos concentram a maioria dos defeitos identificados nos testes antes da entrega, com bugs distribuídos de forma desigual.
 - 5º Fundamento: O Paradoxo do pesticida.
 - A repetição constante do mesmo teste reduz a detecção de defeitos, exigindo inovação e criação de novos testes.
 - 6º Fundamento: Os testes dependem do contexto.
 - 7º Fundamento: A ausência de erros é uma ilusão.
- **Técnica Pomodoro:** Dividir o tempo em blocos:
 - 25 minutos de trabalho focado;
 - 5 minutos de pausa;
 - A cada 4 pomodoros, uma pausa maior de 15 a 30 minutos;



Norma IEC/ISO 25010 [↗](#)



@Ádony Lagares

- Define um modelo de qualidade de software (características e subcaracterísticas).
- **Adequação:** Capacidade do sistema em fornecer funções que atendam corretamente necessidades específicas;
- **Usabilidade:** Facilidade de uso, aprendizado e atratividade do sistema para o usuário;
- **Compatibilidade:** Software funcionar em conjunto com outros sistemas e ambientes;
- **Confiança:** Capacidade do sistema de manter a tríade CID (confiabilidade, integridade e disponibilidade);
- **Eficiência no desempenho:** Maior velocidade das funções de um sistema com um menor uso de recursos;
- **Manutenibilidade:** Facilidade de modificar o software;
- **Portabilidade:** Capacidade de transferir o sistema para diferentes ambientes;
- **Segurança:** Proteção de acessos não autorizados, vazamentos de dados e falhas de segurança;

Tipos de Teste [↗](#)

@Cassia Yumi



Os testes automatizados possibilitam uma **redução de mais de 80%** na incidência de bugs



- **Testes Manuais:** testador executa as ações do sistema de forma manual, sem uso de ferramentas automáticas.
- **Testes Automatizados:** realizados por ferramentas e scripts de maneira automática.
 - Teste de regressão automatizado: tipo de teste que verifica se as mudanças feitas no código de um sistema não afetaram negativamente as funcionalidades já existentes.
- **Testes Tradicionais:** seguem uma abordagem sequencial e mais estruturada, com documentação extensa e planejamento detalhado.
- **Testes Ágeis:** são mais adaptativos e iterativos, integrados ao processo de desenvolvimento desde o início.