

Análise de Requisitos & Introdução à Garantia da Qualidade (QA)

Introdução [↗](#)

No desenvolvimento de software, a **análise de requisitos** é uma etapa fundamental para garantir que o produto atenda às expectativas do cliente e funcione conforme o esperado. Já a **Garantia da Qualidade (Quality Assurance – QA)** atua em todas as fases do ciclo de vida do software para evitar falhas, melhorar processos e entregar valor real.

Este artigo aborda, de forma técnica e explicativa, os conceitos-chave sobre análise de requisitos e como QA se relaciona com essa etapa crítica da engenharia de software.

O que são Requisitos? [↗](#)

Requisitos representam as **necessidades, funcionalidades e restrições** que um sistema deve satisfazer. Podem ser divididos em:

- **Requisitos Funcionais:** descrevem *o que* o sistema deve fazer (ex: "o sistema deve permitir login de usuários").
 - **Requisitos Não Funcionais:** descrevem *como* o sistema deve se comportar (ex: desempenho, segurança, usabilidade).
 - **Requisitos de Negócio:** alinham o sistema aos objetivos estratégicos da empresa.
 - **Requisitos Técnicos:** envolvem integrações, frameworks, plataformas, entre outros aspectos técnicos.
-

Etapas da Análise de Requisitos [↗](#)

1. **Elicitação:** levantamento das necessidades com stakeholders (entrevistas, workshops, questionários).
2. **Análise:** avaliação da viabilidade, inconsistências, conflitos e priorização dos requisitos.
3. **Documentação:** uso de documentos formais como SRS (Software Requirements Specification), histórias de usuário e critérios de aceitação.
4. **Validação e Verificação:** garantir que os requisitos estão corretos, completos e alinhados ao negócio.

Ferramentas e Técnicas para Análise de Requisitos [↗](#)

- **User Stories & BDD (Behavior-Driven Development)**
Ex: "Como [tipo de usuário], quero [ação] para [benefício]"
 - **Casos de Uso (UML)**
 - **Personas e Jornada do Usuário**
 - **Prototipagem com Figma, Balsamiq ou Adobe XD**
 - **Análise SWOT, 5W2H, Brainstorming, MoSCoW (Must/Should/Could/Won't)**
-

Introdução ao QA (Quality Assurance) [↗](#)

O que é QA? [↗](#)

Quality Assurance é o conjunto de processos focados em **prevenir defeitos** no produto de software, garantindo que ele seja entregue com o maior nível de qualidade possível.

QA **não é apenas testar** — é participar ativamente do desenvolvimento desde o planejamento, análise de requisitos, design, implementação até o deploy.

Papéis do QA no Ciclo de Vida do Software [↗](#)

Etapa	Participação do QA
Levantamento de Requisitos	Validação de critérios de aceitação, identificação de riscos
Planejamento	Criação de plano de testes, definição de estratégias
Desenvolvimento	Testes contínuos (unitários, de integração, e2e), suporte a devs
Homologação	Execução de testes funcionais, regressivos e exploratórios
Pós-produção	Testes em ambiente real, monitoramento, feedback

QA e Análise de Requisitos: Conexão Estratégica [↗](#)

O QA ajuda a **identificar falhas nos requisitos** antes mesmo de o código ser escrito. Por exemplo:

- Requisitos ambíguos geram testes incertos.
- Falta de critérios de aceitação impede validação automatizada.
- Cenários de exceção mal mapeados criam brechas para bugs.

Um bom analista de QA colabora com PO, devs e analistas para transformar requisitos em **cenários de teste claros, completos e rastreáveis**.

Técnicas de QA Relacionadas à Análise de Requisitos [↗](#)

- **Análise Estática:** revisão de requisitos antes do desenvolvimento.
- **Partição de Equivalência & Análise de Valor Limite**
- **Matriz de Rastreabilidade:** conecta requisitos aos casos de teste.
- **Técnicas de Teste Baseadas em Cenários e Heurísticas (como a Goldilocks)**
- **Mapas Mentais e Test Design Mind Maps**

Exemplos práticos [↗](#)

Exemplo 1 – História de usuário: [↗](#)

“Como cliente, quero poder cadastrar um produto com nome, preço e descrição, para gerenciar meu estoque.”

Critérios de aceitação:

- Não pode haver produtos com nomes duplicados.
- O preço deve ser maior que zero.
- O campo “descrição” é opcional.

Cenários de teste baseados na análise:

1. Cadastro com todos os campos válidos → Sucesso
2. Cadastro com nome duplicado → Erro 409

3. Cadastro com preço negativo → Erro 422
 4. Cadastro com descrição em branco → Sucesso
-

Conclusão

Dominar a **análise de requisitos** é essencial para quem atua com **QA**, pois garante que o processo de testes seja bem direcionado, eficaz e agregue valor ao negócio. Ao entender as necessidades do sistema e colaborar desde o início do projeto, o QA se torna uma peça-chave para prevenir erros e construir software de qualidade.

💡 *“Um bom QA não testa apenas o que foi construído. Ele ajuda a construir o que realmente deve ser testado.”*