

Requisitos Não Funcionais

- Requisitos não funcionais
 - Requisitos de produto
 - Requisitos de eficiência
 - Requisitos de desempenho
 - Requisitos de espaço
 - Requisitos de confiabilidade
 - Requisitos de portabilidade
 - Requisitos de facilidade de uso
 - Requisitos organizacionais
 - Requisitos de entrega
 - Requisitos de implementação
 - Requisitos de padrões
 - Requisitos externos
 - Requisitos de interoperabilidade
 - Requisitos éticos
 - Requisitos legais
 - Requisitos de privacidade
 - Requisitos de segurança

Requisitos não funcionais definem as *características e restrições* que moldam o comportamento do seu sistema, indo além das funcionalidades básicas. São como as regras do jogo que *garantem que tudo funcione como esperado*, desde a performance até a segurança.

1. Requisitos de Produto:

1.1. Eficiência:

- **Desempenho:** Seu sistema precisa ser veloz? Tempo de resposta instantâneo é crucial? Defina metas de performance para garantir uma experiência ágil e fluida.

Exemplo: O sistema de e-commerce precisa processar pagamentos em menos de 3 segundos para garantir uma compra rápida e sem frustrações.

- **Espaço:** Memória e armazenamento são limitados? Otimize o uso de recursos para garantir que o sistema funcione em dispositivos com configurações variadas.

Exemplo: O aplicativo móvel precisa ser leve e consumir menos de 50MB de memória para funcionar em smartphones básicos.

1.2. Confiabilidade:

- **Disponibilidade:** O sistema precisa estar sempre disponível? Defina o tempo de inatividade tolerável para garantir uptime máximo.

Exemplo: O sistema bancário online precisa estar disponível 24 horas por dia, 7 dias por semana, com no máximo 1 hora de inatividade por mês para evitar perdas financeiras.

- **Tolerância a falhas:** O sistema precisa lidar com falhas inesperadas? Implemente mecanismos de recuperação para garantir resiliência.

Exemplo: O sistema de controle de tráfego aéreo precisa ser tolerante a falhas de hardware e software para garantir a segurança das operações aéreas.

1.3. Portabilidade:

- **Adaptabilidade:** O sistema precisa funcionar em diferentes plataformas ou ambientes? Garanta a portabilidade para alcançar um público mais amplo.

Exemplo: O software de gestão empresarial precisa ser compatível com Windows, macOS e Linux para atender às necessidades de diferentes empresas.

1.4. Facilidade de Uso:

- **Usabilidade:** O sistema precisa ser intuitivo e fácil de usar? Simplifique a interface e a navegação para proporcionar uma experiência agradável.

Exemplo: O aplicativo de vendas precisa ter uma interface amigável e intuitiva para que qualquer pessoa possa realizar compras com facilidade.

2. Requisitos Organizacionais:

2.1. Entrega:

- **Processo de desenvolvimento:** Defina como o sistema será desenvolvido, desde a metodologia até as ferramentas e prazos.

Exemplo: O sistema de reservas de hotel será desenvolvido usando a metodologia Agile, com entregas em sprints de duas semanas.

2.2. Implementação:

- **Padrões de codificação:** Estabeleça regras para garantir código consistente e legível.

Exemplo: O sistema de gerenciamento de projetos seguirá o padrão PSR-12 para garantir a qualidade do código.

2.3. Padrões:

- **Conformidade:** Siga normas e regulamentações relevantes para garantir a segurança e a qualidade do sistema.

Exemplo: O sistema de saúde precisa estar em conformidade com a HIPAA para proteger os dados dos pacientes.

3. Requisitos Externos:

3.1. Interoperabilidade:

- **Integração:** O sistema precisa se integrar com outros sistemas ou APIs? Garanta a interoperabilidade para um fluxo de trabalho eficiente.

Exemplo: O sistema de contabilidade precisa se integrar com o sistema de CRM para automatizar a transferência de dados.

3.2. Éticos:

- **Responsabilidade social:** O sistema deve ser desenvolvido de forma ética e responsável, sem causar danos à sociedade.

Exemplo: O sistema de reconhecimento facial não deve ser usado para discriminar pessoas com base em raça, gênero ou religião.

3.3. Legais:

3.3.1. Privacidade:

- **Proteção de dados:** O sistema precisa proteger os dados pessoais dos usuários de acordo com as leis de privacidade.

Exemplo: O aplicativo de mensagens instantâneas precisa seguir a GDPR para proteger os dados dos usuários na Europa.

3.3.2. Segurança:

- **Proteção contra ataques:** O sistema precisa ser protegido contra ataques cibernéticos e violações de segurança.

Exemplo: O sistema bancário online precisa ter mecanismos de segurança robustos para proteger as informações financeiras dos clientes.