



# Revisão e Inspeção

## Introdução

**N**esta aula vamos tratar do gerenciamento da qualidade do software, estão diretamente ligados com os requisitos, pois temos que garantir que o aplicativo esteja adequado aos seus propósitos e que atendam as necessidade e metas dos usuários, sejam confiáveis e seus projetos fiquem dentro do orçamento e no prazo correto.


Para isso, vamos ver as técnicas de revisão e inspeção que tem se mostrado eficientes no auxílio do gerenciamento da qualidade do software.

## Revisão Requisitos

A atividade de revisão ajuda na garantia da qualidade dos requisitos, no primeiro momento a revisão é utilizada para que o próprio autor dos requisitos revise o seu trabalho. Isso evita que os requisitos avancem com defeitos.

A revisão ajuda na avaliação pessoal do trabalho, e a revisão dos requisitos tem os objetivos de revisar os requisitos, identificar defeitos antes dos testes e se estruturado e medido pode-se tirar métricas de desempenho.

A revisão deve ser realizada em todos os artefatos gerados, na elicitação com a lista de requisitos, o modelo de domínio, na análise revisar os critérios definidos e os atributos aplicados aos requisitos, e na especificação onde teremos os detalhes e acertos de requisitos e na validação o contrato.

A fase de Análise, quando se discute as necessidades e me  dos usuários, os cenários vão sendo mapeados e descobertos com o passar do tempo, e isso pode deixar muitas brechas, a revisão auxilia no fechamento assertivo.

Um ciclo de atividades para realizar a revisão nos requisitos: a) a aderência entre os NMs dos usuários e os requisitos gerados; b) formatação e padronização estabelecidos; c) se o nível de detalhamento está adequado aos modelos; e d) criação de pontos de verificação.

## Revisão Código

Nesta disciplina estamos tratando de forma intensa a análise dos requisitos, neste item “revisão de código” tem dois propósitos, a saber: definir os conceitos defeito, erro e falha; e identificar os locais chaves de defeitos.

Um defeito equivale as imperfeições encontradas no aplicativo ou nos processos de levantamento de requisitos, as vezes um defeito fica por um bom tempo antes de ser encontrado.

Um erro equivale a uma imperfeição encontrada no código decorrente de um defeito não resolvido na fase de análise. Em alguns casos estão conectados com regras de negócio e regras complexas de cálculos.

Uma falha equivale a ocorrência do erro para o usuário, sendo a pior situação pois tem um erro no código e ele se materializa para o usuário.

Onde realizar a revisão para mitigar a possibilidade de defeitos, observe a tabela:



Defeito	Local
De Conceitos	Requisitos
De Design	Interpretação dos requisitos
De Lógica de programação	Erros em algoritmos
De não conformidade	Produzido comparado com o especificado



A partir dessa tabela procure identificar os artefatos e realizar a revisão, vai auxiliar na redução de defeitos e consequentemente os reduzir os erros.

## Revisão Testes

Na fase de Análise um dos artefatos que pode ser gerado é o plano de testes gerado a partir dos requisitos descobertos. A revisão do plano de testes auxilia o engenheiro de software a validar os requisitos, pois lacunas e interpretações podem ser identificadas e desta forma serem sanadas a tempo.

Diversos tipos de testes devem ser realizados para garantirmos um mínimo de qualidade do sistema, podemos listar alguns: teste funcional, teste de entrada e saída de dados, teste de fluxo de atividades, teste de integração e teste do sistema.

## Inspeção Requisitos

A inspeção são revisões realizadas em pares com a colaboração do time com o objetivo de encontrar os defeitos e erros nos artefatos e nos códigos.

A inspeção de requisitos visa os artefatos produzidos na fase de Análise que compreende as fases da engenharia de requisitos, ao longo desta disciplina você teve contato com eles, que são: lista de requisitos; o glossário; o modelo de domínio; os casos de uso ou as histórias dos usuários, e agora com os planos de testes.

A criação de checklists é uma boa prática para garantir que todas as atividades foram realizadas. Um checklist para inspeção dos requisitos contempla: a completude dos requisitos, se existem requisitos contraditórios, se existem requisitos ambíguos e se os requisitos estão claros e bem definidos.

## Inspeção Código

Assim como na revisão de código, a inspeção do código feito em pares reduz o retrabalho, a inspeção deverá ser feita para antecipar os erros que poderão existir no código da aplicação.

Como o artefato de requisitos passam por validações dos clientes, o foco da inspeção do código será nos artefatos de design que estão diretamente ligados com a interpretação dos requisitos e definição da arquitetura do software. Inspeccionar se os modelos foram adequadamente entendidos é fundamental para qualidade do software.

E complementamos o foco da inspeção do código com os artefatos da lógica de programação, com a finalidade de identificar defeitos e erros em algoritmos pensados na redução de defeitos e erros que venham a afetar a computação desejada.

## Inspeção Testes

Um plano de testes envolve diversas possibilidades de testes do sistema, a inspeção dos testes visa inspecionar todos os planos que foram criados, como: para os testes funcionais; para os testes de entrada e saída de dados; para os testes de fluxo de atividade; para os testes de integração e para os testes do sistema.

Como os planos de testes refletem os requisitos a realização da inspeção reforça a qualidade dos requisitos e auxilia na remoção de defeitos dos requisitos.

Para a inspeção dos testes, um checklist adequado seria: ins<sup>o</sup>onar se o plano de testes está completo; se o plano cobre todos os requisitos; se cobrem todos os aspectos a serem testados como a funcionalidade, o desempenho, a resposta, ao volume de dados e recuperação e reinício do sistema.

### Atividade Extra

Leitura:

Artigo

<https://www.devmedia.com.br/artigo-engenharia-de-software-introducao-a-inspecao-de-software/8037>

### Referência Bibliográfica

- REINEHR, S. Engenharia de Requisitos – Editora Grupo A. 1ª Edição. 2020.
- SOMMERVILLE, I. Engenharia de Software – Editora Pearson. 10a Edição. 2019.

**Ir para questão**

