

Técnicas de revisão & Garantia de qualidade

“A qualidade de software não aparece simplesmente do nada. Ela é o resultado de um bom gerenciamento de projeto e uma prática consistente de engenharia de software.”

ENGENHARIA DE SOFTWARE I - Sistemas de Informação

Docente: Profa. Dra. Karliane Medeiros Ovidio Vale

Discente: Zaú Júlio A. Galvão

Revisões Técnicas Formais

Revisão técnica formal ou RTF é uma atividade de controle da qualidade de software realizada por engenheiros de software (e outros profissionais). Os objetivos de uma RTF são:

1. Descobrir erros na função, lógica ou implementação para qualquer representação do software;
2. Verificar se o software que está sendo revisado atende aos requisitos;
3. Garantir que o software foi representado de acordo com padrões predefinidos;
4. Obter software que seja desenvolvido de maneira uniforme;
5. Tornar os projetos mais gerenciáveis;





RTF

Além disso, a RTF serve como base de treinamento, possibilitando que engenheiros mais novos observem diferentes abordagens para análise, projeto e implementação de software.

A RTF também serve para promover backup e continuidade, pois muitas pessoas se familiarizam com partes do software que de outra maneira jamais teriam visto.

A RTF é, atualmente, uma classe de revisões que inclui walkthroughs e inspeções.

Cada RTF é realizada como uma reunião e apenas será bem-sucedida se for apropriadamente planejada, controlada e tiver a participação de todos os envolvidos.

Reunião de revisão

Independentemente do formato de RTF escolhido, cada reunião de revisão deve observar as seguintes restrições:

- Devem estar envolvidas de três a cinco pessoas em uma revisão (tipicamente).
- Deve ocorrer uma preparação antecipada, porém não deve tomar mais do que duas horas de trabalho de cada pessoa.
- A duração da reunião de revisão deve ser de menos de duas horas.




O foco da RTF é um artefato resultante



O indivíduo que desenvolveu o artefato informa ao líder de projeto do artefato está completo e que é necessário fazer uma revisão. O líder de projeto contata um líder de revisão, que avalia o artefato em termos de completude, gera cópias dos materiais resultantes e as distribui para dois ou três revisores para preparação prévia.

Espera-se que cada revisor revise o artefato, tomando notas e, de alguma forma, se familiarize com o trabalho realizado.



Uma reunião de revisão tem a participação de um líder de revisão, todos os revisores e o produtor. Um dos revisores assume o papel de registrador, isto é, o indivíduo que registra todas as questões importantes surgidas durante a revisão.

A RTF começa com uma introdução da agenda e uma breve introdução por parte do produtor, que continua na descrição do artefato resultante, explicando o material, enquanto os revisores levantam questões com base em sua preparação prévia.

No final da revisão, todos os participantes da RTF devem decidir se:

1. Aceitam o artefato sem as modificações adicionais;
2. Rejeitam o artefato devido a erros graves (uma vez corrigidos, deve ser realizada outra revisão);
3. Aceitam o artefato provisoriamente (foram encontrados erros secundários que devem ser corrigidos, mas não haverá nenhuma outra revisão);

Relatório de revisão e manutenção de registros



O relatório sintetizado de revisão deve responder a três questões:

1. O que foi revisado?
2. Quem o revisou?
3. Quais foram as descobertas e as conclusões?

A lista de problemas de revisão atende a dois propósitos:

1. Identificar áreas problemáticas no artefato
2. Servir como uma lista de verificação de itens de ação que orienta o produtor à medida que são feitas as correções.

Normalmente é anexada uma lista de problemas ao relatório sintetizado.

Deve-se estabelecer um procedimento de acompanhamento para garantir que itens contidos na lista de problemas tenham sido corrigidos apropriadamente.

Diretrizes de revisão:



Conjunto mínimo de diretrizes para revisões técnicas formais

1. Revisar o produto, não o produtor;
2. Estabelecer uma agenda e mantê-la;
3. Limitar debates e refutação;
4. Enunciar as áreas do problema mas não tentar resolver todo o problema registrado;
5. Tomar notas;
6. Limitar o número de participantes e insistir na preparação antecipada;
7. Desenvolver uma lista de verificação para cada artefato que provavelmente será revisado;
8. Alocar os recursos e programar o tempo para as RTFs;
9. Realizar treinamento significativo para todos os revisores;
10. Revisar revisões iniciais;

Revisões por amostragem



Em um ambiente ideal, todo artefato de engenharia de software deveria passar por uma revisão técnica formal. No mundo real dos projetos de software, os recursos são limitados e o tempo é escasso.

O processo de revisão por amostragem define que amostras de todos os artefatos da engenharia de software sejam inspecionadas para determinar quais são mais suscetíveis a erro.

Para ser eficaz, o processo de revisão por amostragem deve tentar quantificar aqueles produtos de trabalho que são alvos primários para as RTFs completas.

Garantia de Qualidade


“A Qualidade é a Tarefa n.º 1”



A garantia da qualidade de software (SQA) engloba:

1. Um processo de SQA;
2. Tarefas específicas de garantia da qualidade e controle da qualidade;
3. Prática efetiva de engenharia de software (métodos e ferramentas);
4. Controle de todos os artefatos de software e as mudanças feitas nesses produtos;
5. Um procedimento para garantir a conformidade com os padrões de desenvolvimento;
6. Mecanismos de medição e de relatórios;

Elementos de Garantia da Qualidade de Software



A garantia da qualidade de software engloba um amplo espectro de preocupações e atividades, podemos sintetizá-las em:

- **Padrões.** O IEEE, a ISO, et al.
- **Revisões e auditorias.** As revisões técnicas são uma atividade de controle de qualidade com o intuito de revelar erros e garantir o cumprimento das diretrizes.
- **Testes.** Essencialmente, descobrir erros através do planejamento de procedimentos com maior probabilidade para encontrá-los.
- **Coleta e análise dos erros.** Estudo e coleta de métricas a partir dos erros para avaliação do desempenho.
- **Gerenciamento de mudanças.** Instrução de práticas adequadas para gerenciamento de mudanças no projeto de software.
- **Educação.** Liderança no processo de aperfeiçoamento de software.
- **Gerência dos fornecedores.** Controle de qualidade na obtenção de software externo.
- **Administração de Segurança.** Gerência do emprego dos processos e tecnologias necessárias para garantir a segurança desejada.
- **Proteção.** Avaliação do impacto das falhas de software.
- **Administração de riscos.** Garantia do controle de risco e criação de planos de contingência.



Tarefas da SQA

A prerrogativa do grupo de SQA é ajudar a equipe de software a obter um produto final de alta qualidade.

O SEI (Software Engineering Institute) recomenda um conjunto de ações de SQA que tratam do planejamento, da supervisão, da manutenção de registros, da análise e de relatórios relativos à garantia da qualidade.

O objetivo dessas ações é atingir um conjunto de métricas pragmáticas.

O seguinte conjunto de ações:

- Prepara um plano de SQA para um projeto.
- Participar no desenvolvimento da descrição da gestão de qualidade do projeto.
- Revisar as atividades de engenharia de software para verificar sua conformidade com a gestão de qualidade definida.
- Auditar produtos de software resultantes designados para verificar sua conformidade com aqueles definidos como parte da gestão de qualidade.
- Garantir que os desvios no trabalho de software e produtos resultantes sejam documentados e tratados de acordo com um procedimento documentado.
- Registrar qualquer não aderência e relata ao gerenciamento superior.

Meta	Atributo	Métrica
Qualidade das necessidades	Ambiguidade	Número de modificadores ambíguos (por exemplo, muitos, grande, amigável)
	Completude	Número de TBA, TBD
	Compreensibilidade	Número de seções/subseções
	Volatilidade	Número de mudanças por requisito
		Tempo (por atividade) quando é solicitada a mudança
		Número de requisitos não atribuíveis ao projeto/código
		Clareza do modelo
	Facilidade de atribuição	Número de modelos UML
	Clareza do modelo	Número de páginas descritivas por modelo
		Número de erros UML

Metas, atributos e métricas

Meta	Atributo	Métrica
Qualidade do projeto	Integridade da arquitetura	Existência do modelo da arquitetura Número de componentes que se atribui ao modelo da arquitetura
	Compleitude dos componentes	Complexidade do projeto procedural Número médio de cliques para chegar a uma função ou conteúdo típico
	Complexidade da interface	Apropriabilidade do layout
	Padrões	Número de padrões usados

Metas, atributos e métricas

Meta	Atributo	Métrica
Qualidade do código	Complexidade Facilidade de manutenção Compreensibilidade Reusabilidade Documentação	Complexidade ciclométrica Fatores de projeto Porcentagem de comentários internos Convenções de atribuição de variáveis Porcentagem de componentes reutilizados Índice de legibilidade
Eficiência do controle de qualidade	Alocação de recursos Taxa de completude Eficácia da revisão Eficácia dos testes	Porcentagem de horas de pessoal por atividade Tempo de finalização real versus previsto Número de erros encontrados e criticalidade Esforço exigido para corrigir um erro Origem do erro

Metas, atributos e métricas



Estatística da Garantia da Qualidade

A estatística da garantia de qualidade reflete uma tendência crescente em toda a indústria de software para tornar mais quantitativa a análise da qualidade.

Para software, a estatística da garantia da qualidade implica as seguintes etapas:

1. Informações sobre erros e defeitos de software são coletadas e classificadas.
2. É feita uma tentativa de associar cada erro e defeito a sua causa subjacente.
3. Usando o princípio de Pareto (80% dos defeitos podem ser associados a 20% de todas as possíveis causas), são isoladas os 20%.
4. Assim que as poucas causas vitais tiverem sido identificadas, prossegue-se para a correção dos problemas que provocaram os erros e defeitos.



Confiabilidade

A confiabilidade de software, diferentemente de outros fatores de qualidade, pode ser medida diretamente e estimada usando-se dados históricos e de desenvolvimento. Podemos definir a confiabilidade de software em termos estatísticos como:

“a probabilidade de operação sem falhas de um programa de computador em um dado ambiente por um determinado tempo”.

As falhas podem ser apenas problemáticas ou catastróficas. Uma determinada falha pode ser corrigida em segundos, enquanto outras necessitarão de semanas ou até mesmo meses para serem corrigidas.

O Plano de SQA

O plano de SQA fornece um roteiro para instituir a garantia da qualidade de software. Desenvolvido pelo grupo de SQA (ou pela equipe de software, caso não exista um grupo de SQA), o plano serve como um gabarito para atividades de SQA que são instituídas para cada projeto de software.





Plano de SQA: IEEE

O padrão publicado pela IEEE recomenda uma estrutura que identifique:

1. O propósito e o escopo do plano.
2. Uma descrição de todos os artefatos resultantes de engenharia de software que caem dentro do âmbito da SQA.
3. Todos os padrões e práticas que são aplicados durante a gestão de qualidade.
4. As ações e tarefas da SQA (incluindo revisões e auditorias) e sua aplicação na gestão de qualidade.
5. As ferramentas e os métodos que dão suporte às ações e tarefas da SQA.
6. Procedimentos para administração de configurações de software.
7. Métodos para montagem, salvaguarda e manutenção de todos os registros relativos à SQA.
8. Papéis e responsabilidades dentro da organização relacionados com a qualidade do produto.



REFERÊNCIA

Pressman, Roger, and Bruce Maxim. *Engenharia de Software-8ª Edição*. McGraw Hill Brasil, 2016.