

[< VOLTAR](#)

CMMI - Áreas de Processos: Garantia da Qualidade; Medição e Análise

Detalhar a áreas de processos do CMMI-DEV nível 2 de Garantia da Qualidade de Processo e da área de processos Produto e de Medição e Análise.

NESTE TÓPICO

➤ Área de Processos de Garantia da Qualidade de Processo e Produto

➤ Qualidade de software

➤ Software Quality Assurance (SQA)

➤ Process and Product Quality Assurance (PPQA)



Área de Processos de Garantia da Qualidade de Processo e Produto

No modelo CMMI-DEV, essa área de processos, tem como o objetivo principal, assegurar que os compromissos assumidos pelo projeto estão sendo realizados de forma satisfatória. Os processos dessa área fornecem subsídios para que, toda a equipe e a gerência, tenha visibilidade suficiente sobre os processos e produtos de trabalho, visando condições exercer a garantia de qualidade (KOSCIANSKI e SOARES, 2007) (SOFTWARE ENGINEERING INSTITUTE, 2010).

A área de processo Garantia da Qualidade de Processo e Produto apoia a entrega de produtos e serviços de alta qualidade, fornecendo à equipe do projeto e aos gerentes de todos os níveis a visibilidade apropriada sobre os processos e produtos de trabalho associados, ao longo do ciclo de vida do projeto.

((SOFTWARE ENGINEERING INSTITUTE, 2010))



Controle de Qualidade

Vejamos a seguir alguns importantes conceitos sobre garantia da qualidade de software.

Qualidade de software

Segundo Pressman (2011), em um sentido geral, a qualidade de software pode ser definida como:

Uma gestão de qualidade efetiva aplicada de modo a criar um produto útil que forneça valor mensurável para aqueles que o produzem e para aqueles que o utilizam.

((PRESSMAN, 2011))

Esse “produto útil”, no caso o software, deve oferecer aos seus usuários finais todas as funções e recursos que esses necessitam (requisitos funcionais). Mas Pressman (2011) ainda lembra que, além disso, o software deve satisfazer a um conjunto de requisitos implícitos (por exemplo, facilidade de uso). O autor ainda destaca a importância que há no seguimento de padrões de desenvolvimento que sejam claramente documentados.

Alguns conceitos são indispensáveis para entender de forma ampla as definições sobre qualidade de software apresentadas pelo autor. São eles:

- **Requisitos de software:** é a base pela qual a qualidade do software é avaliada. Quando não há concordância com os requisitos, existe falta de qualidade.
- **Padrões de desenvolvimento:** conjunto de atividades, regras e convenções adotadas pela área de desenvolvimento que orientam a forma como um

software deve ser desenvolvido. A não adoção a esses critérios pode originar um produto de software com qualidade duvidosa.

- **Características implícitas:** um software é composto, de requisitos funcionais e não funcionais. Os requisitos funcionais são as características explícitas do software e os requisitos não funcionais referem-se às características implícitas, por exemplo, a usabilidade. A falta de conformidade com esses requisitos pode comprometer a qualidade do produto de software.

Software Quality Assurance (SQA)

Em português, essa sigla é traduzida como: Garantia de Qualidade do Software. Trata-se de um planejamento de ações que visam garantir que o produto de software possua a qualidade esperada. As ações executadas englobam:

- Aplicação de técnicas que auxiliam o desenvolvedor a conseguir especificações de requisitos com elevada qualidade, fato que permite que o projeto desenvolvido também possua qualidade.
- Avaliações técnicas periódicas, ou **Revisões Técnicas Formais (RTF)**, que têm por objetivo encontrar erros durante o processo antes que eles se tornem defeitos e avaliam a qualidade das especificações do projeto.
- Testes de software, que, quando executados desde os passos iniciais do projeto, identificam erros de entendimento ou de interpretação durante todo o ciclo de vida do desenvolvimento do software.
- Aplicação dos padrões de desenvolvimento baseados na engenharia de software.
- Controle de mudanças para permitir um controle efetivo sobre as alterações implementadas na configuração do software.



Process and Product Quality Assurance (PPQA)

A Garantia de Qualidade de Processo e Produto (PPQA – sigla em inglês) é uma evolução da SQA. É uma área responsável pela qualidade, tanto dos produtos de software que são desenvolvidos, quanto pelo processo que é seguido para o desenvolvimento desses produtos.

Os objetivos específicos dessa área são os seguintes:

- Avaliação da aderência do projeto aos processos, normas e procedimentos existentes na organização no que se refere ao desenvolvimento de sistemas.
- Resolução de problemas de não aderência, detectados no desenvolvimento de um projeto.

As ferramentas que são utilizadas para a avaliação do produto e a verificação dos processos em relação aos seus objetivos são as auditorias e a avaliação de objetivos, visando identificar:

- A utilização correta do processo de desenvolvimento de software.
- O rigor dos controles de qualidade.
- A eficiência dos controles de qualidade.
- A adequação do processo de desenvolvimento de software às necessidades e expectativas da organização.

Formalmente, o controle da qualidade do produto encontra-se entre as incumbências da área de processos. Assegurar a qualidade está nas ações que devem ser observadas segundo os critérios do CMMI.

Faz parte do controle da qualidade a execução dos testes, entretanto, controlar a qualidade, não se restringe somente à realização de testes. Trata-se de um processo que deve ter uma profundidade maior e mais abrangente.

A principal ferramenta do PPQA é a relação de atividades que serão alvo da auditoria. Essas atividades dizem respeito à sequência de ações que devem ser executadas para o desenvolvimento do sistema. Um relatório, originado dessa auditoria, é enviado a todos os envolvidos com o projeto apontando os pontos falhos, caso existam, no que se refere à falta de conformidade com o processo de desenvolvimento de sistemas da organização.

É importante salientar que os auditores respondem diretamente à alta administração da empresa, evitando que exista conflito de interesses, o que poderia ocorrer caso os auditores estejam vinculados a uma determinada área.

Após a apresentação do relatório, os desvios (itens que apresentaram falta de conformidade) devem ser corrigidos, podendo haver ou não prazo definido para que esse acerto seja efetuado e a “não conformidade” encerrada. A orientação é que os desvios sejam corrigidos o mais rápido possível e tenham a orientação do gerente de projeto.

Obviamente, todas essas atividades mencionadas devem ter como base um processo de desenvolvimento de software definido e maduro, sem o qual não será possível garantir a qualidade do produto de software.



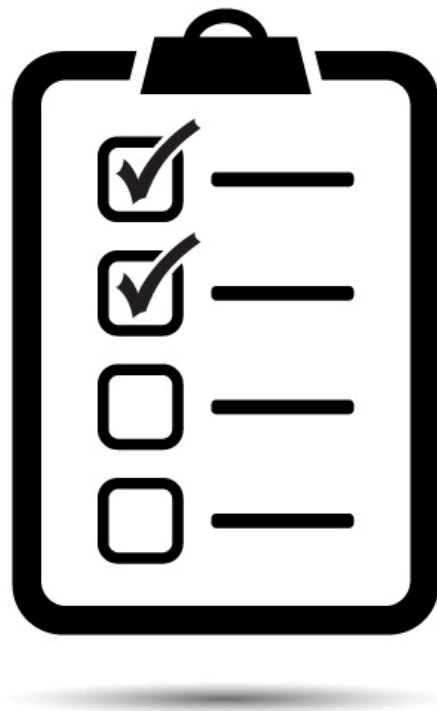
Aspectos importantes da garantia de qualidade

- A garantia de um produto de software está ligada à conformidade do produto aos requisitos explícitos e implícitos.
- A qualidade de um produto de software deve ser incorporada desde o início de seu desenvolvimento. É impossível incorporar a qualidade depois de terminado o processo de desenvolvimento.
- Um processo de desenvolvimento de software é composto por etapas muitas vezes complexas. A complexidade desse processo pode acarretar perda de qualidade do produto.
- Cada etapa do processo de desenvolvimento é um produto que afetará a qualidade do produto desenvolvido na próxima etapa, ou seja, se a etapa de requisitos for confeccionada com uma qualidade duvidosa, a etapa de

projeto, que é a seguinte, provavelmente gerará um produto com qualidade baixa, e assim sucessivamente.

Para apoiar a organização na implementação dessa área de processos, o modelo CMMI-DEV traz diretrizes, onde as principais atividades relacionadas a essa área são:

- Avaliar objetivamente os processos e produtos de trabalho e serviços selecionados em relação às descrições de processo, padrões e procedimentos aplicáveis.
- Fornecer visibilidade, que envolve principalmente comunicar e assegurar a solução de não conformidades.
- Estabelecer e manter registros das atividades de garantia da qualidade.



Check List

Área de Processos de Medição e Análise

Uma boa gestão de projeto de software envolve o uso correto de informações. Quanto mais acurada e completa for a informação, melhores as condições para se analisar o desempenho do projeto quanto a sua qualidade, o atendimento a prazos e o efetivo controle de gastos. As métricas (medições) fornecem uma parte importante dos dados necessários para a administração do projeto de software (KOSCIANSKI e SOARES, 2007).

O objetivo da área de processo Medição e Análise do CMMI-DEV é fornecer subsídios para desenvolver e manter uma capacidade de medição utilizada para dar suporte às necessidades de informação para gestão (SOFTWARE

ENGINEERING INSTITUTE, 2010).

A medida é, por definição, a quantificação de uma característica. No caso da área de desenvolvimento de sistemas, não somente as características dos produtos finais, mas também dos processos envolvidos na concepção, construção e manutenção de software (VAZQUEZ, SIMÕES e ALBERT, 2013).

Para cada um dos objetivos que deseja acompanhar é possível estabelecer um conjunto de perguntas que verifique o seu cumprimento; para muitas dessas perguntas é possível identificar uma métrica que possa quantificar a resposta.

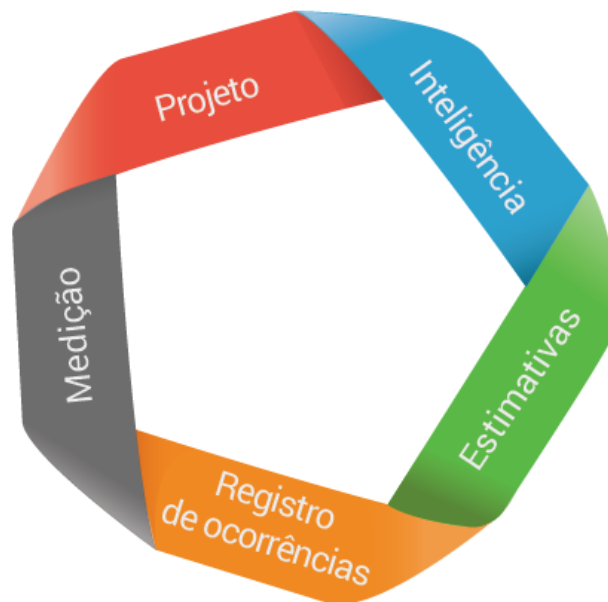
((BASILI, 1992))

Na sequência, você terá a oportunidade de estudar os importantes conceitos que envolvem os processos de medição e análise.

Estimativa

Como não é possível administrar o que não se pode medir, para que um projeto de desenvolvimento de software seja conduzido de forma controlada, respeitando prazos, custos e esforço previamente determinados, é necessário que uma estimativa seja feita sobre esses itens.

Estimar não significa determinar os valores exatos, mas também não é colocar valores aleatórios, sem qualquer base. Trata-se de um processo que utiliza experiência e medições efetuadas em projetos anteriores e que são continuamente refinadas. Estimar é, definindo de forma simples, planejar.



Ciclo de Regimento para Métricas

A figura acima mostra um ciclo de refinamento para as métricas de um processo. A contínua avaliação e adequação das métricas em cada projeto executado permite que as estimativas estejam cada vez mais próximas da realidade.

Vamos agora entender um pouco melhor o conceito, a definição, sobre métrica.

Métrica

Medir é um processo em que números ou símbolos são atribuídos a entidades do mundo real, descrevendo-as com regras previamente determinadas. A medição tem o propósito de estabelecer conclusões sobre uma determinada entidade e deve ter especificações claras para que seja precisa. Tais especificações podem ser entendidas através de três elementos básicos:

- **Domínio:** dimensões que devem ser consideradas (exemplo: na medição de um veículo, deverão ser consideradas as dimensões de largura, comprimento e altura).
- **Faixa:** padrão pelo qual as dimensões devem ser expressas (exemplo: metros, centímetros, milímetros, pés, etc.)
- **Mapeamento:** regras que determinam a abrangência de uma medição (exemplo: a altura do veículo deve ser medida a partir do chão ou a partir da lataria?).

Além desses três elementos, que fazem menção às quantidades, uma métrica deve também ter os seus elementos qualitativos engajados dentro de padrões críticos, como:

- **Precisão válida:** a medição deve quantificar, de forma exata ou dentro dos padrões de exatidão aceitos, o que se quer medir.
- **Confiabilidade:** deve produzir resultados idênticos se as mesmas condições forem novamente implementadas.
- **Praticidade:** deve ser facilmente interpretada a quem for dirigida.

No que se refere à área de Tecnologia da Informação (T.I.), as métricas são utilizadas para aprimorar processos, produtos e serviços através de quatro conceitos:

- **Entendimento:** auxiliam a analisar comportamentos e o funcionamento de processos, produtos e serviços.
- **Avaliação:** podem auxiliar no processo de tomada de decisões para a determinação de metas ou critérios de aceitação.
- **Controle:** podem proporcionar uma visão de como um processo ou serviço deve ser conduzido de forma eficiente.
- **Previsão:** o constante aprimoramento das métricas, através dos sucessivos refinamentos, permite prever, com relativa exatidão, o desempenho na execução de um projeto.

Vejamos na sequência as categorias as quais as métricas são divididas.



Categorias de métricas

As medições podem ser divididas em duas categorias:

- **Medidas diretas:** são medidas que possuem convenções pré-estabelecidas, objetivas e facilmente compreendidas. Geralmente expressam valores exatos. Custo, esforço e prazo são exemplos.
- **Medidas indiretas:** são medidas subjetivas, difíceis de avaliar ou que não possuem uma convenção específica. Confiabilidade, usabilidade e manutenibilidade podem ser consideradas medidas indiretas devido a suas características que carecem de exatidão.

É fundamental que se decida de forma eficiente o que deve ser medido no projeto. Medir o que não é necessário é perda de tempo e tira o foco do que é importante. Deixar de mensurar aquilo que é relevante para o projeto pode comprometer a qualidade do produto ou serviço. Segundo os autores Vazquez, Simões e Albert (2013) “a chave está na identificação dos objetivos estabelecidos tanto para o sucesso do desenvolvimento e manutenção de sistemas quanto para o produto gerado em si”. A seguir, será abordado um método voltado ao planejamento de medições no projeto.

Abordagem GQM

GQM é uma sigla, que em inglês significa Goal, Question, Metric. É um paradigma que tem uma abordagem para a identificação de métricas úteis e relevantes. Seu principal objetivo é focar em objetivos específicos.



Ele foi criado por Victor Basili para avaliar defeitos em projetos de voo no Centro Espacial Kennedy, da Nasa e, portanto, não é um produto de utilização exclusiva da área de T.I. Pode ser utilizado para apoiar a medição de qualquer tipo de produto ou processo. As três palavras que formam a sigla são utilizadas para identificar os pilares sobre os quais o GQM foi criado:

- **Goal:** determinar as metas/objetivos que devem ser alcançados.
- **Question:** identificar as questões que devem ser respondidas para verificar se os objetivos foram atingidos. As respostas obtidas com a medição devem trazer informações úteis para melhorar o produto.
- **Metric:** decidir o que deve ser medido e quais métricas serão utilizadas para que as questões possam ser respondidas.



GQM

Etapas do Método GQM

Como qualquer processo, o GQM possui uma sequência de tarefas a ser seguida:



- Identificar e listar os objetivos do processo de medição.
- Para cada objetivo, formular as questões que devem ser respondidas para averiguar se os objetivos foram atingidos.
- Definir o tipo de métrica que será utilizada e o que deve ser medido.

Os objetivos da medição são expressos em termos de:

- **Entidade:** objeto em que se deseja efetuar a medição (exemplo: sistema de conta corrente).
- **Propósito:** motivo pelo qual é necessário fazer a medição na entidade (exemplo: analisar a entidade).
- **Atributos de qualidade:** identifica o domínio do propósito (exemplo: confiabilidade do sistema no que se refere à segurança).
- **Ponto de vista:** identifica que o objeto deve ser medido para atender aos requisitos de algum protagonista (exemplo: cliente).
- **Ambiente:** indica onde a medição deverá ser realizada (exemplo: Banco Y).

Qualidade das métricas

A escolha das métricas a serem utilizadas é um fator que deve ser considerado com cuidado. Segundo Koscianski:

Todo trabalho de avaliação é colocado em risco se não for possível garantir a obtenção de dados confiáveis. Se o resultado de uma medida é utilizado para julgar o software, é preciso ter certeza de que ela está sendo feita da maneira correta.

((KOSCIANSKI, 2007))

Além da necessidade de obter resultados precisos, os custos podem também influenciar na escolha das métricas para avaliação do produto. Medições complexas podem exigir mais recursos, ocasionando um custo maior do que outros tipos de medições. Para que essa escolha seja feita com base em critérios sólidos, pode-se utilizar o modelo SQuaRE (Software Quality Requirements and Evaluation), um conjunto de normas que tem por objetivo a especificação e a avaliação das normas de qualidade, e que apresentam as seguintes premissas, no que se refere a métricas:

- Métricas devem ter significância. As informações obtidas devem ser úteis para avaliar a qualidade.
- Os custos das métricas devem estar de acordo com a avaliação a ser realizada, ou seja, deve-se avaliar o custo/benefício ao se aplicar determinado tipo de métrica.
- Métricas devem ser padronizadas para que possam ser reutilizadas e os mesmos resultados devem ser obtidos se as mesmas condições forem impostas.
- O resultado da métrica deve ser objetivo, ou seja, desprovido de interpretação dúbia.
- Métricas devem ter imparcialidade. A escolha correta dos parâmetros a serem verificados impede que um produto seja favorecido em relação a outro.



Atividades de Medição e Análise no CMMI-DEV

A área de processos de medição e análise traz as diretrizes para se desenvolver e manter uma capacidade de medição, que é utilizada para dar suporte às necessidades de informação para gestão.

As atividades principais do CMMI-DEV nessa área de processos envolvem:

- **Alinhar Atividades de Medição e Análise**
 - Estabelecer Objetivos de Medição
 - Especificar Medidas
 - Especificar Procedimentos de Coleta e Armazenamento de Dados
 - Especificar Procedimento de Análise

- **Fornecer Resultados de Medição**
 - Coletar Dados Resultantes de Medição
 - Analisar Dados Resultantes de Medição
 - Armazenar Dados e Resultados
 - Comunicar Resultados

Juntamente com essas diretrizes, os conceitos e técnicas vistos nesse tópico, são uma base sólida para serem aplicados na implementação dessa área de processos em uma organização.

Quiz

Exercício Final

CMMI - Áreas de Processos: Garantia da Qualidade; Medição e Análise



INICIAR ➤

Referências

BARTIÉ, A. *Garantia da qualidade de software*. Rio de Janeiro: Campus, 2002.

BASILI, V. R. Software modeling and measurement: the Goal/Question/Metric paradigm. *University of Maryland*, CS-TR-2956, Maryland, set. 1992. 92-96.

KOSCIANSKI, A.; SOARES, M. D. S. *Qualidade de software*: aprenda as metodologias e técnicas mais modernas para o desenvolvimento de software. 2. ed. São Paulo: Novatec, 2007.

PAULA FILHO, W. D. P. *Engenharia de Software*: fundamentos, métodos e padrões. Rio de Janeiro: LTC, 2003.

PFLEEGER, S. L. *Engenharia de Software*: teoria e prática. 2. ed. São Paulo: Pearson, 2004.

PRESSMAN, R. S. *Engenharia de Software*: Uma Abordagem Profissional. 7. ed. Porto Alegre: McGraw Hill, 2011.

SOFTWARE ENGINEERING INSTITUTE. *CMMI for Services Version 1.3*. Bedford: Carnegie Mellon, 2010. Disponível em: <www.sei.cmu.edu/reports/10tr033.pdf>. Acesso em: 21 set. 2016.

SOMMERVILLE, I. *Engenharia de Software*. 8. ed. São Paulo: Pearson Addison-Wesley, 2007.

VAZQUEZ, C. E.; SIMÕES, G. S.; ALBERT, R. M. *Análise de Pontos de Função*: Medição, Estimativas e Gerenciamento de Projetos de Software. 13. ed. São Paulo: Érica, 2013.



Avalie este tópico



ANTERIOR

CMMI - Áreas de Processos: Gestão de Fornecedores, Gestão da Configuração

Biblioteca

(https://www.uninove.br/conheca-

a-

uninove/biblioteca/sobre-

a-

biblioteca/apresentacao/)

Portal Uninove

(http://www.uninove.br)

Mapa do Site



Índice

Ajuda?

PRÓXIMO
(https://ava.un
idCurso=)

CMMI-DEV - Níveis de C

© Todos os direitos reservados

