Validação: O processo de validação representa a forma de avaliar a

qualidade de um produto ou componente de produto, garantindo o atendimento das

necessidades dos seus usuários, quando colocado em seu ambiente de uso.

O processo de validação em um projeto de software envolve algumas etapas, desde um

bom planejamento, onde serão definidos os produtos que serão validados, os métodos e

as técnicas que serão utilizados, além da definição dos ambientes necessários,

ferramentas e recursos que serão utilizados na validação.

(Softex, 2007a)

Selecionar Produtos para validação ==> Determinar o Ambiente de Validação ==>

Determinar Procedimentos e Critérios de Validação ==> Realizar a Validação.

**Selecionar produtos para validação**: Nesta atividade, os produtos e componentes

dos produtos a serem validados e os métodos de validação que serão utilizados são

selecionados (CMMI PRODUCT TEAM, 2006).

**Determinar o ambiente de validação**: Nesta atividade, o ambiente necessário para

suportar a validação será determinado e mantido (CMMI PRODUCT TEAM, 2006).

**Determinar procedimentos e critérios de validação**: Nesta atividade, os

procedimentos e critérios para validação são determinados e mantidos. Os

procedimentos e critérios de validação incluem a avaliação de manutenção, treinamento

e serviços de apoio (CMMI PRODUCT TEAM, 2006).

**Realizar a validação**: Nesta atividade, a validação é realizada nos produtos e

componentes de produtos selecionados. Atividades de validação são realizadas e os

resultados são coletados de acordo com métodos, procedimentos e critérios

estabelecidos (CMMI PRODUCT TEAM, 2006).

**Verificação**: O processo de verificação trata da forma de como avaliar produtos de trabalho e serviços, garantindo

que atendam a seus requisitos. Assim como o processo de validação, para realizar uma

verificação é necessário um planejamento para identificar os itens a serem verificados,

a forma de verificação de cada um desses itens e a execução da verificação conforme

planejado ao longo do desenvolvimento do produto. (Sofitex 2007a p. 37)

Selecionar Produtos de Trabalho para Verificação ==> Determinar o Ambiente de Verificação ==>

Determinar Procedimentos e Critérios de Verificação ==> Realizar a Verificação.

**Selecionar produtos de trabalho para verificação**: Nesta atividade, os produtos de

trabalho a serem verificados e os métodos de verificação que serão utilizados são

selecionados (CMMI PRODUCT TEAM, 2006).

**Determinar o ambiente de verificação**: Nesta atividade, o ambiente necessário

para suportar a verificação será determinado e mantido (CMMI PRODUCT TEAM, 2006).

Isso envolve hardware, softwaree outros recursos.

**Determinar procedimentos e critérios de verificação**: Nesta atividade, os

procedimentos e critérios para verificação dos produtos de trabalho selecionados são

determinados e mantidos. Critérios de verificação são definidos para medir o quanto os

produtos de trabalho estão aderentes aos seus requisitos (CMMI PRODUCT TEAM,

2006).

**Realizar a verificação**: Nesta atividade, a verificação é realizada nos produtos e

componentes de produtos selecionados, de acordo com os métodos, procedimentos e

critérios disponíveis (CMMI PRODUCT TEAM, 2006).

Diferença entre esses conceitos:

**Verificação**: Um conjunto de atividades que garanta

que um software implemente corretamente uma função específica. O objetivo da

verificação é assegurar que o software, ou determinada função dele, esteja sendo

implementada corretamente. Verifica-se, inclusive, se os métodos e processos de

desenvolvimento foram adequadamente aplicados (ROCHA, MALDONADO & WEBER,

2001)

**Validação**: Um conjunto de atividades diferentes

que garante que o software construído corresponde aos requisitos do cliente. O objetivo

da validação é assegurar que o software que está sendo desenvolvido é o software

correto de acordo com os requisitos do usuário (ROCHA, MALDONADO & WEBER, 2001).

Resumo, Boehm (apud PRESSMAN, 2006).

Técnicas "inspeção" e a "revisão técnica".

**Inspeção**: Define inspeção como "um método formal de revisão por pares, onde um grupo de 'pares'", incluindo o autor, se

reúne para examinar um determinado produto de trabalho. Esta revisão é realizada por

um grupo de pessoas que possuem o mesmo perfil do autor do produto a ser revisado. (apudVASCONCELOS, MACIEL & ROUILLER, 2005)

• **Planejamento**: O objetivo do planejamento da inspeção é: coletar o material para

revisão; selecionar a equipe e definir os papeis dos integrantes da equipe de

revisão; planejar o cronograma de execução; e determinar se é necessária a reunião

de apresentação.

• **Reunião de Kikoff**: Nesta reunião o moderador apresenta os participantes e os

papeis de cada um para a equipe de revisão e informa o produto e os objetivos da

inspeção.

• **Preparação Individual**: Os revisores avaliam o material e elencam possíveis

dúvidas a serem esclarecidas pelo autor na reunião de inspeção.

• **Reunião de Inspeção**: O objetivo da reunião de inspeção é revisar o produto

para classificar e registrar os defeitos encontrados sem a preocupação de solucionálos. Vale ressaltar que o que é avaliado é o produto, não o seu autor; o autor

participa apenas para esclarecer, não para se justificar. O autor esclarece possíveis

dúvidas dos revisores durante a reunião.

• **Registro de Métricas**: O moderador registra, por exemplo, o tamanho dos

produtos e a quantidade de defeitos encontrados para a avaliação da inspeção.

**Revisão Técnica**: É o filtro mais efetivo do ponto de vista de garantia de qualidade. Conduzida

por engenheiros de software (e outros) para engenheiros de software, a FTR é um meio

efetivo de descobrir erros e aperfeiçoar a qualidade do software.(Formal Technical Review – FTR)

Uma atividade de garantia da qualidade de software realizada por engenheiros de software (e outros)", que tem como

objetivos:

• Descobrir erros na função, na lógica ou na implementação, para qualquer

representação do software.

• Verificar se o software sob revisão satisfaz seus requisitos.

• Garantir que o software tenha sido representado de acordo com padrões

predefinidos.

• Conseguir software que seja desenvolvido de modo uniforme.

• Tornar os projetos mais administráveis.