Ciência da Computação Engenharia de Softwares II Prof. Rafael Stoffalette João

04/09/2020

### Sobre a última aula

#### **CMMi**

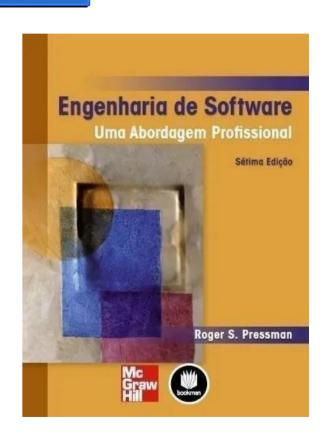
- → o que é?
- → Para que serve?
- → Por que é importante?

#### **EMENTA**:

- Métricas e Projeto e Implementação de Software CONCEITOS DE GERENCIAMENTO DE PROJETO
- Qualidade de Software
- Técnicas de Teste de Software
- Ferramentas de Teste de Software
- Manutenção de Software
- Engenharia Reversa e Reengenharia
- Ferramentas CASE e Técnicas UML

## Bibliografia

Capítulo 24



## Atenção aos prazos

Nossa avaliação será em:

24/09/2020

19h20min

### Projetos cansativos/ineficientes

→ resultado de gerenciamento de projeto fraco

# Gerenciamento efetivo de desenvolvimento de software foca nos 4Ps:

- Pessoas;
- Produto;
- Processo; e
- Projeto.

#### Gerenciamento efetivo de desenvolvimento de software foca nos 4Ps:

- Pessoas;
- Produto;
- Processo; e
- Projeto.

Falha de comunicação pode implicar em uma solução elegante para o problema errado.

#### SEI desenvolveu um CMMI para recursos humanos

- People-CMMI

```
formação de equipe
comunicação
ambiente de trabalho
treinamento
análise de competência, ...
```

#### Interessados (5 grupos):

Gerentes seniores: definem itens de negócio

Gerentes técnicos: organizar, controlar, motivar e planejar os programadores envolvidos

Programadores: habilidade técnica

Clientes: especificam os requisitos.

Usuários finais: interagem com o software em ambiente de produção

Líder da equipe:

Responsável pela integração das equipes descritas anteriormente.

Ótimos programadores são, geralmente, péssimos líderes.

#### Modelo MOI:

- Motivação: encorajar pessoal técnico.
- Organização: moldar processos/criar novos.
- Ideias/Inovação: incentivo à criatividade.

Um gerente de projeto:

Sabe diagnosticar e solucionar problemas;

- Organizar uma sistemática ou encorajar equipes a encontrar uma solução

Assume a responsabilidade do projeto.

- Assumir o controle e permitir instinto da equipe

Recompensa iniciativas para otimizar a produtividade da equipe

Lê pessoas. Identificar estado mental e sinais para prever crises.

Um gerente de projeto:

Sabe diagnosticar e solucionar problemas;

- Organizar uma sistemática ou encorajar equipes a encontrar uma solução

Assume a responsabilidade do projeto.

- Assumir o controle e permitir instinto da equipe

Recompensa iniciativas para otimizar a produtividade da equipe

Lê pessoas. Identificar estado mental e sinais para prever crises.

Solução de problemas

Identidade gerencial

Realizações

Formação de equipe e de influência

Equipe de software

A melhor estrutura depende do tipo de gerenciamento da organização – flexível; levando em conta:

- Grau de dificuldade do problema
- Tamanho dos programas gerados
- Tempo de cada equipe atuante
- Grau de modularização
- Qualidade e confiabilidade do sistema a ser produzido
- Rigidez de prazos
- Grau de sociabilidade requerido no projeto

Antes de tudo, deve-se ter um objetivo do produto e seu escopo;

considera soluções alternativas;

considerar restrições técnicas e administrativas.

Objetivos == metas gerais sem considerar "como" atingi-las.

Gerente de projeto deve estimar um plano.

- Informações sólidas nem sempre são disponíveis no início.

Análises levam semanas

- Requisitos podem ser informados ao decorrer do projeto

No mínimo, o escopo do produto deve ser estabelecido.

No mínimo, o escopo do produto deve ser estabelecido.

Escopo de software. Construído com as seguintes perguntas

- Contexto: restrições do contexto e integração externa
- Objetivo da informação: objetos de entrada/saída necessários
- Função e performance: o que o software faz e o desempenho deve ser considerado?

No escopo NÃO deve haver ambiguidade.

Deve ser claro:

- Número de usuários simultâneos;
- Tempo máximo de resposta;
- Consumo de memória;
- Qual a restrição de memória?

- ...

Decomposição do problema.

A medida em que os parâmetros do escopo evoluem, ocorre um particionamento natural.

- Equipe notificada;
- Cada nova funcionalidade é uma subfunção a ser implementada.

Metodologia: qual a melhor forma de desenvolver (estratégia);

pensar em GQS: Garantia de Qualidade de Software e GCM: Gerenciamento de Configuração e de Medições.

Todo desenvolvimento deve seguir um modelo de processo (cascata/evolutivo/espiral/...)

A EQUIPE decide qual modelo mais apropriado; levando em conta:

- As pessoas envolvidas (cliente e desenvolvedores);
- Características do produto. Fácil modularização?

Após a definição a equipe define o planejamento inicial, depois o particionamento...

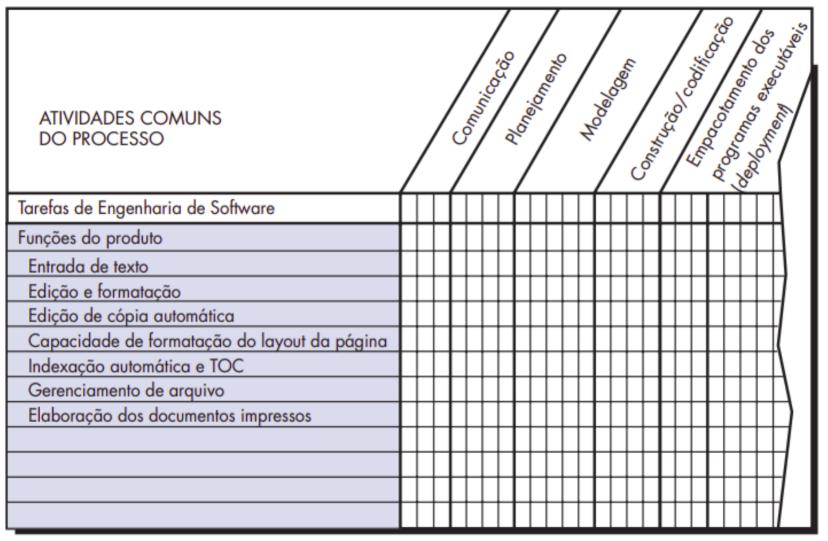
O projeto começa pela integração do produto ao processo

Construção de uma matriz de integração

Linhas: tarefas a serem realizadas (produto)

Colunas: atividades do processo

Trabalho do gerente: estimar necessidades para cada célula da matriz, recursos necessários, data de início e fim.



Colunas: Atividades

Linhas:

**Tarefas** 

Um projeto pequeno e relativamente simples pode requerer as seguintes tarefas para as atividades de **comunicação**:

- 1. Desenvolver uma lista de itens para esclarecimentos.
- 2. Reunir-se com interessados para esclarecer os itens pendentes.
- 3. Desenvolver conjuntamente uma base documentada do escopo.
- 4. Revisar a base considerando todos os envolvidos.
- 5. Alterar a base conforme necessário.

Agora, considere um projeto mais complexo, com um escopo mais amplo e um impacto comercial mais significativo. Tal projeto pode vir a requerer as seguintes tarefas para comunicação:

- 1. Revisão da solicitação do cliente.
- 2. Planejamento e agendamento de reuniões viabilizadas e formais com todos os envolvidos.
- 3. Realização de uma pesquisa para especificar a solução proposta e as abordagens existentes.
- 4. Preparação de um documento de trabalho e de uma agenda para a reunião formal.
- 5. Realização de reunião.
- 6. Desenvolvimento conjuntamente de mini especificações que reflitam os dados, a funcionalidade e os fatores comportamentais do software. De forma alternativa, desenvolvimento de

casos de uso que descrevam o software sob o ponto de vista do usuário.

- 7. Revisão de cada mini especificações ou caso de uso para realizar correções, consistências e eliminação de ambiguidades.
- 8. Reunião de mini especificações em um documento de escopo (bases).
- 9. Revisão do documento de escopo (bases) ou a coletânea de casos de uso com todos os envolvidos.
- 10. Alteração do documento de escopo ou de casos de uso conforme o necessário

Ambos os projetos constituem as atividades comunicação, entretanto, a primeira equipe executa metade das tarefas de trabalho de engenharia.

### Decomposição de processo

A equipe de software deve ter um grau de flexibilidade ao escolher o modelo de processo de software mais adequado ao projeto e às tarefas de engenharia de software que fazem parte do modelo selecionado.

Na tarefa de comunicação do projeto pequeno, cada item pode ser uma subtarefa com prazo - decomposição

Única maneira de administrar a complexidade;

estudo de 1998-2004: 25 de 250 grandes projetos conseguiram cumprir o cronograma, custos e objetivos.

Gerente deve estar atento ao que pode sair errado,

- Ações planejadas para evitar tais problemas.
- 10 sinais indicadores de que um projeto está em perigo:
- 1. O pessoal de software não compreende as necessidades de seus clientes.
- 2. O escopo do produto está parcialmente definido.
- 3. As alterações são mal gerenciadas/administradas.
- 4. A tecnologia escolhida muda.
- 5. As necessidades de negócio mudam (ou são mal definidas).
- 6. Os prazos estão fora da realidade.
- 7. Os usuários mostram-se resistentes.
- 8. O patrocínio é perdido (ou nunca foi propriamente obtido).
- 9. Faltam profissionais à equipe ou esta não possui pessoal com habilidades adequadas.
- 10. Gerentes e desenvolvedores evitam práticas e lições aprimoradas e aprendidas.

Regra 90-90 - projetos particularmente difíceis.

- Os primeiros 90% de um sistema absorvem 90% dos esforços e tempos alocados.
- Os 10% restantes consomem outros 90% de esforço e tempo alocados

- Comece com o pé-direito.
  - estabelecer expectativas e objetivos realísticos para todos envolvidos no projeto
  - concedendo-lhe autonomia, autoridade e tecnologia necessárias para
  - realizar o trabalho.

- Mantenha a velocidade (ímpeto).
  - Muitos projetos começam bem e depois desintegram lentamente.
  - Fornecer incentivos para que a rotatividade de pessoal deve ser mínima.
  - A equipe deve dar ênfase à qualidade em todas as tarefas que realiza
  - Gerente sênior posicionar-se fora do caminho da equipe

- Rastreie o andamento.
  - Mapeamento de como os artefatos (códigos-fonte, conjunto dos pacotes de testes) são produzidos e aprovados
  - Métricas de projeto podem ser coletadas e utilizados para avaliar o progresso

- Decisões com agilidade
  - "manter a simplicidade". Sempre que possível;
  - Utilização de software comercial
  - Atenção aos riscos óbvios.

- análise post-mortem.
  - Aprender com cada projeto.
  - Como aprender com erros?
  - Avalie os cronogramas planejados e os realizados,
  - as métricas de projetos de software coletadas e analisadas,
  - Obtenha feedback dos membros da equipe e dos clientes e
  - Documente TUDO

### Qualidade de software

Definição:

"Qualidade de software é a conformidade a requisitos funcionais e de desempenho que foram explicitamente declarados, a padrões de desenvolvimento claramente documentados, e a características implícitas que são esperadas de todo software desenvolvido por profissionais" [Pressman, 1994].

### Qualidade de software

Definição:

"Qualidade é a totalidade de características e critérios de um produto ou serviço que exercem sua habilidades para satisfazer às necessidades declaradas ou envolvidas" [ISO9126, 1994].

### Incorporação de Qualidade

Raramente incorporada ao produto final,

 após o processo de desenvolvimento ter terminado é muito difícil "incluir" qualidade

Produtos intermediários tem certos atributos de qualidade que afetam a qualidade do seguinte (próxima fase) e assim, afetam a qualidade do produto final.

### Incorporação de Qualidade



- A qualidade n\u00e3o pode ser incorporada ao produto depois de pronto
- Para que a qualidade possa ser efetivamente incorporada ao produto, ela deve ser um objetivo constante do processo de desenvolvimento.
- Processo de garantia de qualidade do software atividades de garantia da qualidade de software (GQS)

### Métricas de qualidade de software

"Não se pode gerenciar o que não se pode medir" – Tom de Marco

"Se você não sabe para onde quer ir, qualquer caminho você pode seguir. Se você não sabe onde está, um mapa não vai ajudar!" — Pressman

Uma métrica é a medição de um atributo (propriedades ou características) de uma determinada entidade (produto, processo)

### Métricas de qualidade de software

#### Mas medir o que???????

Processo → medir "produtividade"

Produto → medir "qualidade"

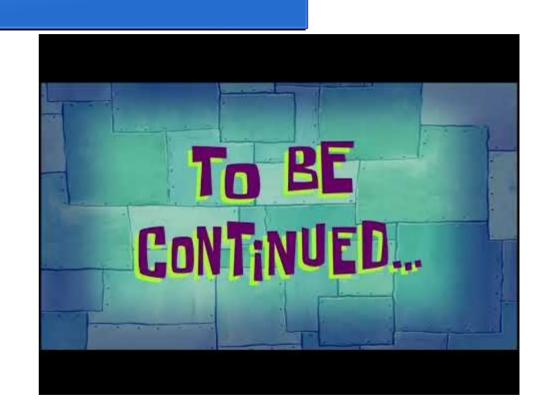
LOC: Lines of code

**KLOC: K-lines** 

**CLOC**: commented

Inputs

. . .



## Em prática

Durante a copa do mundo de futebol você decide chamar uns amigos para curtir a semifinal. Como você realizaria essa tarefa?

#### Lembre-se

- Lugar, convidados e agregados, comida, ...
- Como garantir que esse evento tenha qualidade?
- Como garantir que a final seja tão boa (ou melhor) que esse evento?