

Gerência de Projetos

Métricas de Software

O Processo de Gerência de Projetos

- Projeto de Software bem sucedido depende de uma eficaz **Gerência de Projeto**
- **Gerência de Projetos**
 - Abrange todo o processo de desenvolvimento.
 - Envolve:
 - o escopo do trabalho a ser feito
 - os riscos a que se incorre
 - os recursos exigidos
 - as tarefas a serem executadas
 - o esforço (custo) despendido
 - a programação a ser seguida

Elementos Chave da Gerência de Projetos

- Objetivos e Escopo do Projeto
- Medidas e Métricas
- Estimativa
- Análise de Riscos
- Determinação de Prazos
- Monitoração e Controle

Elementos Chave da Gerência de Projetos

- Objetivos e Escopo do Projeto
- Medidas e Métricas
- Estimativa
- Análise de Riscos
- Determinação de Prazos
- Monitoração e Controle

Objetivos e Escopo do Projeto



- estabelecer objetivos e escopo
- considerar soluções alternativas
- identificar restrições administrativas e técnicas



- Definir estimativas de custo razoáveis (e precisas)
- Divisão realística das tarefas de projeto
- Programação de projeto administrável que ofereça indícios significativos de progresso

Objetivos e Escopo do Projeto

- O desenvolvedor e o cliente reúnem-se para definir os objetivos e o escopo do projeto (processo de engenharia de sistemas)
- **Objetivos:**
 - identificam as metas globais do projeto sem levar em consideração como essas metas serão atingidas
- **Escopo:**
 - identifica as funções primárias que o software deve realizar e tenta *delimitar* essas funções de uma forma quantitativa

Elementos Chave da Gerência de Projetos

- Objetivos e Escopo do Projeto
- Medidas e Métricas
- Estimativa
- Análise de Riscos
- Determinação de Prazos
- Monitoração e Controle

Medidas e Métricas

- As **medições** e as **métricas** ajudam-nos a entender o processo técnico usado para se desenvolver um produto, como também o próprio produto.
- O **processo** é medido num esforço para melhorá-lo.
- O **produto** é medido num esforço para aumentar sua qualidade.

Elementos Chave da Gerência de Projetos

- Objetivos e Escopo do Projeto
- Medidas e Métricas
- **Estimativa**
- Análise de Riscos
- Determinação de Prazos
- Monitoração e Controle

Estimativa

- **Planejamento**: atividade fundamental do processo de gerenciamento de projetos,
- No Planejamento do Projeto de Software devem ser derivados:
 - **estimativa** do esforço humano exigido (pessoas-mês)
 - duração cronológica do projeto (em tempo de calendário)
 - custo (em dólares)

Como são feitas as estimativas?

Estimativa

- Muitas vezes as **Estimativas** são feitas usando-se a **experiência passada** como um único guia
 - se o projeto não for semelhante, a experiência passada poderá não ser suficiente
- Uma série de **técnicas de estimativa** foram disponibilizadas para o desenvolvimento de software
- Todas as técnicas têm os seguintes atributos em comum:

Estimativa

ATRIBUTOS EM COMUM

- O escopo do projeto deve ser estabelecido antecipadamente
- Métricas de software são utilizadas e o histórico de aferições passadas é usado como uma base a partir da qual as estimativas são feitas
- O projeto é dividido em pequenas partes que são estimadas individualmente

atributos em comum:

Elementos Chave da Gerência de Projetos

- Objetivos e Escopo do Projeto
- Medidas e Métricas
- Estimativa
- **Análise de Riscos**
- Determinação de Prazos
- Monitoração e Controle

Análise de Riscos

- Sempre que um software for construído haverá áreas de incerteza.
- A análise dos riscos é crucial para um bom gerenciamento de projeto de software

**Passos para
atacar os riscos**



- identificação dos riscos
- avaliação dos riscos
- disposição por ordem de prioridade
- estratégias de administração
- resolução dos riscos
- monitoração dos riscos

Elementos Chave da Gerência de Projetos

- Objetivos e Escopo do Projeto
- Medidas e Métricas
- Estimativa
- Análise de Riscos
- **Determinação de Prazos**
- Monitoração e Controle

Determinação de Prazos

- A programação do projeto de software não é diferente de qualquer projeto de engenharia
 - Um conjunto de tarefas de projeto é identificado
 - Interdependências entre as tarefas são estabelecidas
 - O esforço associado a cada tarefa é estimado
 - Pessoas e outros recursos são atribuídos
 - Uma rede de tarefas é criada
 - Um gráfico de Gant (*time-line*) é desenvolvido

Elementos Chave da Gerência de Projetos

- Objetivos e Escopo do Projeto
- Medidas e Métricas
- Estimativa
- Análise de Riscos
- Determinação de Prazos
- **Monitoração e Controle**

Monitoração e Controle

- Cada tarefa anotada na programação de desenvolvimento deve ser rastreada pelo gerente de projetos.
 - se a tarefa não acompanhar a programação, pode-se usar uma ferramenta de planejamento e controle de projetos para determinar o impacto do não cumprimento dos prazos sobre os marcos de referência intermediários do projeto e a data de entrega global

Elementos Chave da Gerência de Projetos

- Objetivos e Escopo do Projeto
- Medidas e Métricas
- Estimativa
- Análise de Riscos
- Determinação de Prazos
- Monitoração e Controle

Métricas

- Razões para se medir o software:
 - Indicar a qualidade do produto
 - Avaliar a produtividade dos que desenvolvem o produto
 - Determinar os benefícios derivados de novos métodos e ferramentas de engenharia de software
 - Formar uma base para as estimativas
 - Ajudar na justificativa de aquisição de novas ferramentas ou de treinamentos adicionais

Métricas

MEDIDAS DO SOFTWARE



MEDIDAS DIRETAS

- Custo
- Esforço
- Linhas de Código
- Velocidade de Execução
- Memória
- Nro de Erros

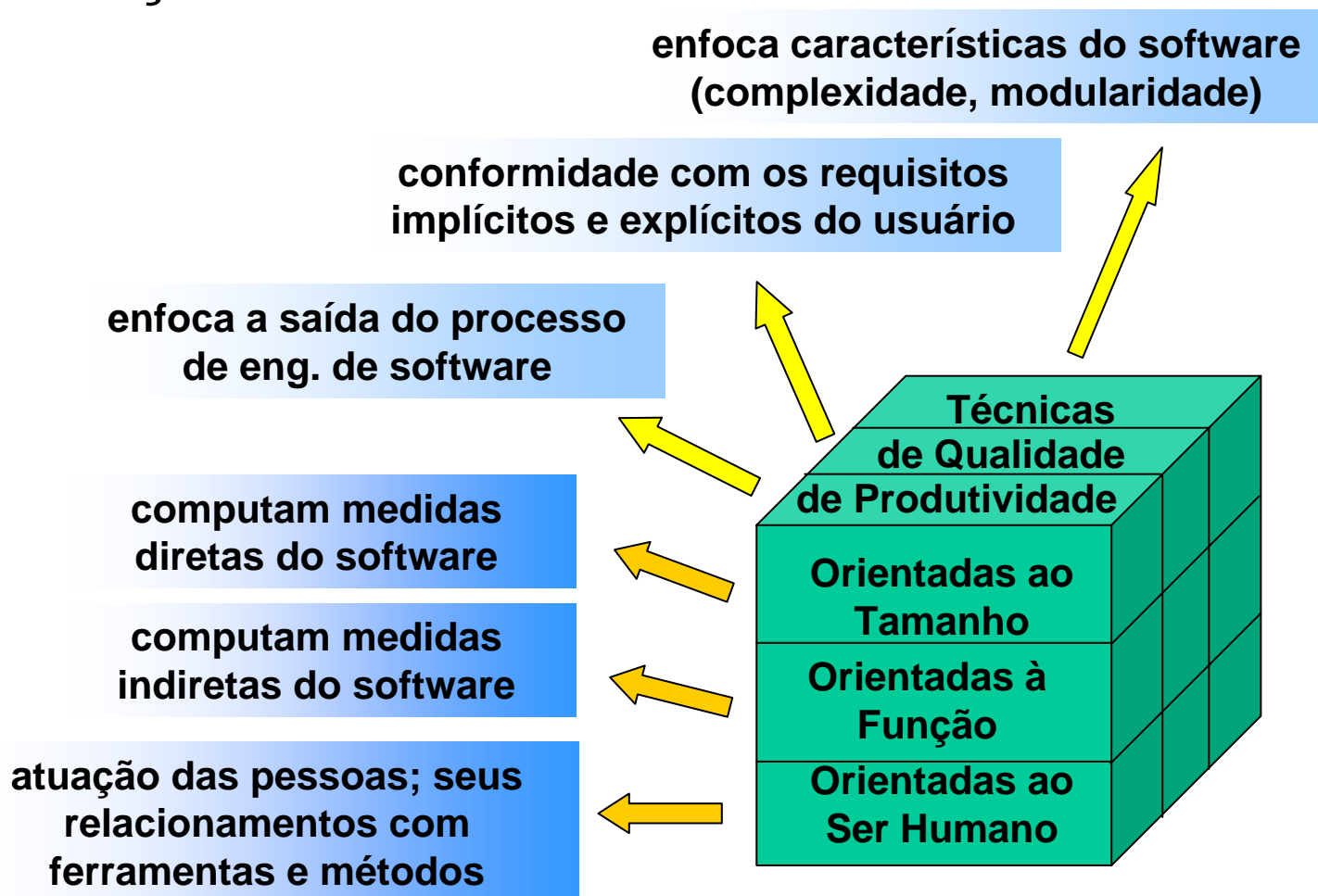


MEDIDAS INDIRETAS

- Funcionalidade
- Qualidade
- Complexidade
- Eficiência
- Confiabilidade
- Manutenibilidade

Métricas

• Classificação das Métricas



Métricas

MÉTRICAS ORIENTADAS AO TAMANHO

São derivadas de medidas diretas do software e do processo através do qual ele é desenvolvido

Exemplos: **LOC** - Lines of Code

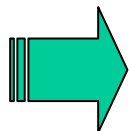
KLOC - Thousand Lines of Code

Métricas

MÉTRICAS ORIENTADAS AO TAMANHO LOC/KLOC

projeto	esforço	\$	KLOC	pags.docum.	erros	pessoas
projA-01	24	168	12.1	365	29	3
projB-04	60	440	27.2	1224	86	5
projC-03	48	314	20.2	1050	64	6

MÉTRICAS DERIVADAS



PRODUTIVIDADE = $\text{KLOC} / \text{pessoas-mês}$

QUALIDADE = $\text{erros} / \text{KLOC}$

CUSTO = $\text{\$/LOC}$

DOCUMENTAÇÃO = $\text{pags.docum.} / \text{KLOC}$

Métricas

MÉTRICAS ORIENTADAS AO TAMANHO

VANTAGENS:

- Fáceis de serem obtidas
- Vários modelos de estimativa baseados em LOC ou KLOC

DESVANTAGENS:

- LOC depende da linguagem de programação
- Penalizam programas bem projetados, mas pequenos
- Não se adaptam às linguagens não procedimentais
- Difícil de obter em fase de planejamento

Métricas

MÉTRICAS ORIENTADAS À FUNÇÃO

São derivadas de medidas indiretas do software e do processo através do qual ele é desenvolvido

Exemplo: **PF** - Pontos de Função

(Albrecht 1979)

Métricas

MÉTRICA ORIENTADA À FUNÇÃO - PF

Concentra-se na **funcionalidade** ou **utilidade** do software

Os PFs são derivados usando uma relação empírica baseada em medidas do **domínio de informação** e da **complexidade** do software

Métricas

MÉTRICA ORIENTADA À FUNÇÃO - PF

PONTOS DE FUNÇÃO É APLICADO ATRAVÉS DE 3 PASSOS:

1) Completar a seguinte tabela:

Parâmetro	Contagem		fator de ponderação			
			Simples	Médio	Complexo	
nro de entradas do usuário	<input type="text"/>	x	3	4	6	<input type="text"/>
nro de saídas do usuário	<input type="text"/>	x	4	5	7	<input type="text"/>
nro de consultas do usuário	<input type="text"/>	x	3	4	6	<input type="text"/>
nro de arquivos	<input type="text"/>	x	7	10	15	<input type="text"/>
nro de interfaces externas	<input type="text"/>	x	5	7	10	<input type="text"/>
Contagem-Total	→					<input type="text"/>

PONTOS POR FUNÇÃO É APLICADO ATRAVÉS DE 3 PASSOS:

Parâmetro	Consideração					
	Simple	Complexo				
nro de entradas do usuário		6				
nro de saídas do usuário		7				
nro de consultas do usuário	<input type="text"/>	x	3	4	6	<input type="text"/>
nro de arquivos	<input type="text"/>	x	7	10	15	<input type="text"/>
nro de interfaces externas	<input type="text"/>	x	5	7	10	<input type="text"/>
Contagem-Total						

Métricas

MÉTRICA ORIENTADA À FUNÇÃO - PF

PONTOS POR FUNÇÃO É APLICADO ATRAVÉS DE 3 PASSOS:

1) Completar a seguinte tabela:

Parâmetro	Contagem	fator de ponderação				
		Simple	Médio	Complexo		
nro de entradas do usuário	saídas de usuário que forneçam informações orientadas a aplicações (relatórios, telas, mensagens de erro)			6	<input type="text"/>	
nro de saídas do usuário				7	<input type="text"/>	
nro de consultas do usuário				6	<input type="text"/>	
nro de arquivos				15	<input type="text"/>	
nro de interfaces externas	<input type="text"/>	x	5	7	10	<input type="text"/>
Contagem-Total	<div></div>					<input type="text"/>

Métricas

MÉTRICA ORIENTADA À FUNÇÃO - PF

PONTOS POR FUNÇÃO É APLICADO ATRAVÉS DE 3 PASSOS:

1) Completar a seguinte tabela:

Parâmetro	Contagem		fator de ponderação			
			Simple	Médio	Complexo	
nro de entradas do usuário	<input type="text"/>	x	3	4	6	<input type="text"/>
nro de saídas do usuário	<input type="text"/>	x	4	5	7	<input type="text"/>
nro de consultas do usuário	entrada <i>on-line</i> que resulte em saída <i>on-line</i>				6	<input type="text"/>
nro de arquivos	<input type="text"/>	x	7	10	15	<input type="text"/>
nro de interfaces externas	<input type="text"/>	x	5	7	10	<input type="text"/>
Contagem-Total						<input type="text"/>

Métricas

MÉTRICA ORIENTADA À FUNÇÃO - PF

PONTOS POR FUNÇÃO É APLICADO ATRAVÉS DE 3 PASSOS:

1) Completar a seguinte tabela:

Parâmetro	Contagem		fator de ponderação			
			Simple	Médio	Complexo	
nro de entradas do usuário	<input type="text"/>	x	3	4	6	<input type="text"/>
nro de saídas do usuário	<input type="text"/>	x	4	5	7	<input type="text"/>
nro de consultas do usuário	<input type="text"/>	x	3	4	6	<input type="text"/>
nro de arquivos	cada arquivo lógico				15	<input type="text"/>
nro de interfaces externas	<input type="text"/>	x	5	7	10	<input type="text"/>
Contagem-Total						<input type="text"/>

Métricas

MÉTRICA ORIENTADA À FUNÇÃO - PF

PONTOS POR FUNÇÃO É APLICADO ATRAVÉS DE 3 PASSOS:

1) Completar a seguinte tabela:

Parâmetro	Contagem		fator de ponderação			
			Simple	Médio	Complexo	
nro de entradas do usuário	<input type="text"/>	x	3	4	6	<input type="text"/>
nro de saídas do usuário	<input type="text"/>	x	4	5	7	<input type="text"/>
nro de consultas do usuário	todas as interfaces legíveis por máquina, usadas para transmitir informação para outro sistema				6	<input type="text"/>
nro de arquivos					15	<input type="text"/>
nro de interfaces externas					10	<input type="text"/>
Contagem-Total						<input type="text"/>

Métricas

MÉTRICA ORIENTADA À FUNÇÃO - PF

2) Responder as questões 1-14, considerando a escala de 0 a 5:



- | | |
|---|---|
| <ol style="list-style-type: none"> 1. O sistema exige backup e recuperação confiáveis? 2. É requerida comunicação de dados? 3. Existem funções de processamento distribuído? 4. O desempenho é crítico? 5. O sistema funcionará num sistema operacional existente e intensamente utilizado? 6. São requeridas entrada de dados <i>on-line</i>? 7. As entradas <i>on-line</i> requerem que as transações de entrada sejam construídas com várias telas e operações? | <ol style="list-style-type: none"> 8. Os arquivos são atualizados <i>on-line</i>? 9. Entradas, saídas, arquivos e consultas são complexos? 10. O processamento interno é complexo? 11. O código é projetado para ser reusável? 12. A conversão e a instalação estão incluídas no projeto? 13. O sistema é projetado para múltiplas instalações em diferentes organizações? 14. A aplicação é projetada de forma a facilitar mudanças e o uso pelo usuário? |
|---|---|

Métricas

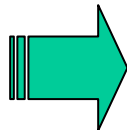
MÉTRICA ORIENTADA À FUNÇÃO - PF

3) Ajustar os Pontos por Função de acordo com a complexidade do sistema, através da seguinte fórmula:

$$PF = \text{Contagem-Total} \times \left(0,65 + 0,01 \times \sum_{i=1}^{14} (F_i) \right)$$

F_i = valores de ajuste da complexidade das perguntas 1-14

**MÉTRICAS
DERIVADAS**



PRODUTIVIDADE = PF / pessoas-mês

QUALIDADE = erros / PF

CUSTO = \$ / PF

DOCUMENTAÇÃO = pags.docum. / PF

Métricas

MÉTRICAS ORIENTADAS À FUNÇÃO

VANTAGENS:

- Independentes da linguagem
- Ideal para aplicações que usam linguagem não procedimental
- Baseados em dados mais fáceis de serem conhecidos durante a evolução do projeto

DESVANTAGENS:

- Cálculo baseado em dados subjetivos
- Não é uma medida direta; é apenas um número

Reconciliação de Diferentes Abordagens de Métricas

- A relação entre **linhas de código** e **pontos de função** depende da linguagem de programação

Linguagem de Programação	LOC/PF (Média)
Assembly	300
COBOL	100
FORTRAN	100
Pascal	90
Ada	70
Linguagens Orientadas a Objeto	30
Linguagens de Quarta Geração	20
Geradores de Código	15

Métricas

DE QUALIDADE

- **corretitude** - grau em que o software executa a função que lhe é exigida
- **manutenibilidade** - grau de facilidade com que o software pode ser corrigido, adaptado ou ampliado
- **integridade** - capacidade que um software tem de suportar *ataques* (acidentais ou intencionais) à sua integridade
- **usabilidade** - tenta quantificar a característica de *user friendliness* do software

Métricas

DE QUALIDADE

- **corretitude** - grau em que o software executa a função que lhe é exigida

ERROS / KLOC é a medida mais comum

os defeitos são registrados pelo usuário depois que o software foi liberado para uso, e são contados ao longo de um período de tempo padrão

Métricas

DE QUALIDADE

- **manutenibilidade** - grau de facilidade com que o software pode ser corrigido, adaptado ou ampliado

Tempo médio para mudança

corresponde ao tempo que demora para analisar um pedido de mudança, projetar a modificação adequada, implementar a mudança, testá-la e distribuir para todos os usuários

Métricas

DE QUALIDADE

- **integridade** - capacidade que um software tem de suportar *ataques* (acidentais ou intencionais) à sua integridade

$$\text{Integridade} = \sum (1 - \text{ameaça} \times (1 - \text{segurança}))$$

ameaça - probabilidade de que um ataque de um tipo específico ocorrerá dentro de determinado tempo

segurança - probabilidade de que o ataque de um tipo específico será repellido

Métricas

DE QUALIDADE

- **usabilidade** - tenta quantificar a característica de *user friendliness* do software

Pode ser medida através de 4 características:

1. habilidade física/intelectual para se aprender o sistema
2. tempo exigido para se tornar moderadamente eficiente no uso
3. aumento de produtividade por alguém que seja moderadamente eficiente
4. avaliação subjetiva (questionário)

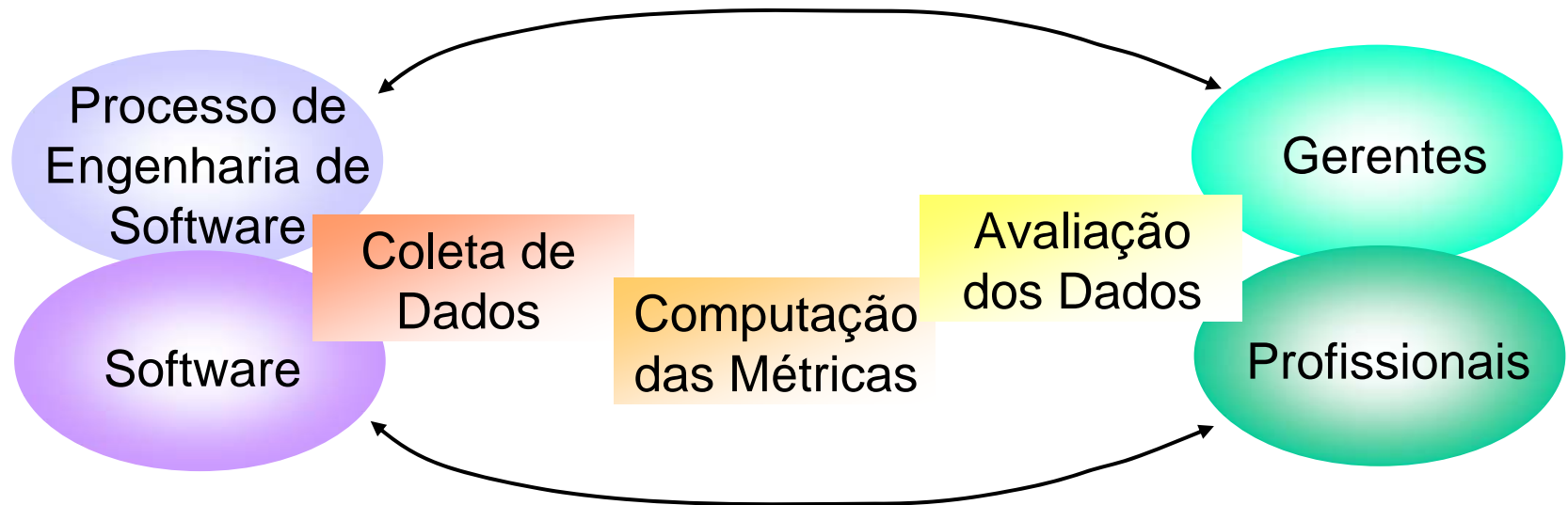
Métricas

BASELINE - DADOS HISTÓRICOS

- Atributos dos Dados Históricos:
 - Ajudam a reduzir o risco das estimativas
 - Devem ser precisos ou próximos de um valor real
 - Coletados do maior número de projetos possível
 - As medidas devem ser interpretadas da mesma maneira durante todo o projeto
 - As aplicações devem ser similares às do trabalho que se quer estudar
- *Existe um modelo de planilha para coleta e cálculo de dados históricos do software*

Métricas

COLETA, COMPUTAÇÃO E AVALIAÇÃO DAS MÉTRICAS



BASELINE - DADOS HISTÓRICOS

Gerência de Projetos

Métricas de Software