

Métricas de software / Estimativas

Engenharia de Software I

Introdução

- Um cliente ao contratar um determinado projeto de software tem interesse de saber:
 - Qual o custo?
 - Qual o prazo de entrega?
- A empresa quer saber:
 - Qual o tamanho do produto?
 - Qual o esforço?
- As empresas necessitam **estimar** o tamanho dos produtos de *software* visando a realização de um melhor planejamento para a construção de produtos de *software* e, ainda, diminuir o risco da tomada de decisões errôneas.

Medição nas áreas de engenharia



Medição nas áreas de engenharia:

- A medição é algo comum no mundo da engenharia, existindo vários tipos de grandezas para medição: peso, temperatura, dimensões, tensão, entre outras.

Medição na área de engenharia de software

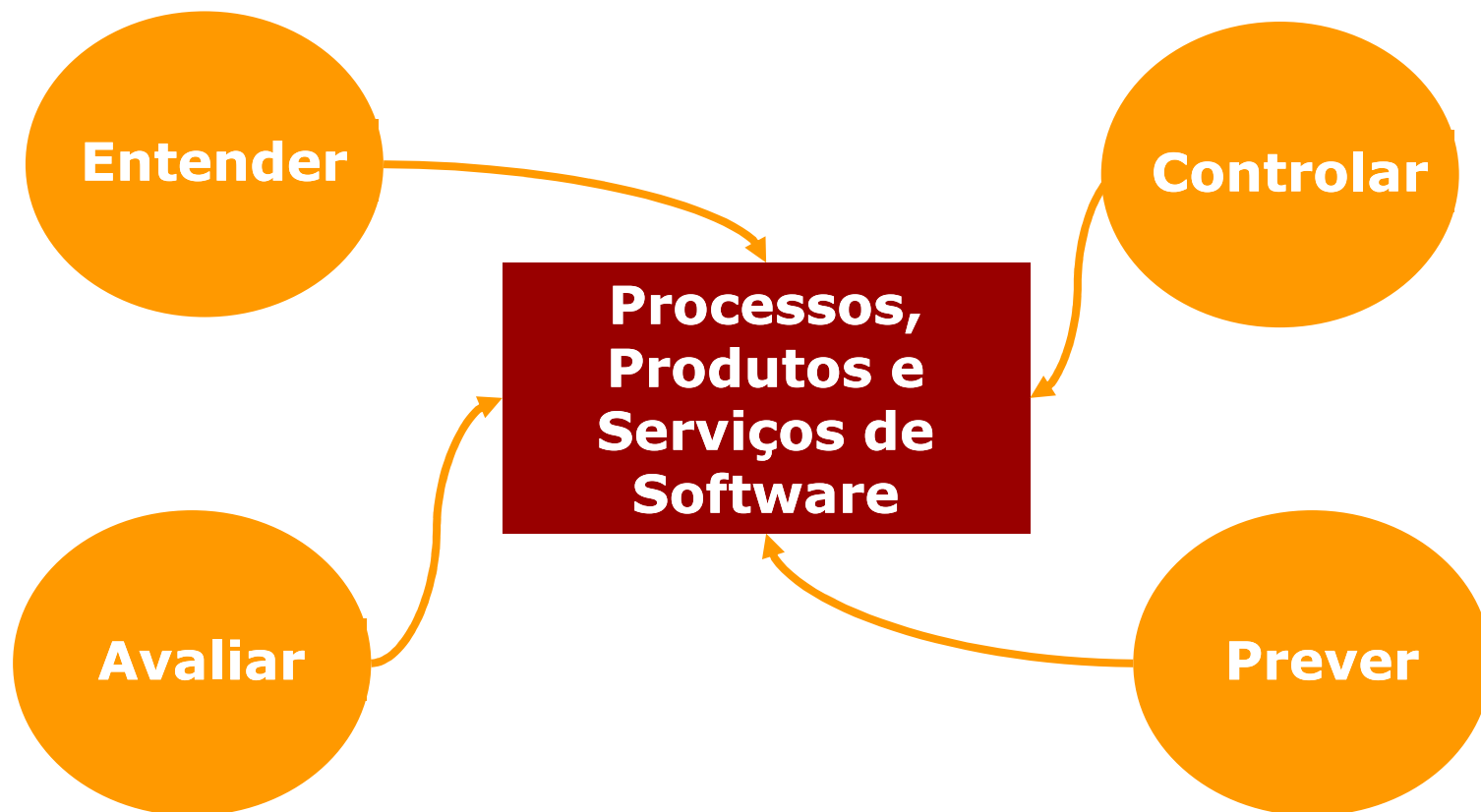
- Infelizmente, a engenharia de software está longe de ter uma medição padrão amplamente aceita e com resultados sem nenhum fator subjetivo.
- Temos dificuldade em concordar sobre o que medir e como avaliar o resultado das medições obtidas.
- **Métricas de softwares** nos possibilitam realizar uma das atividades mais fundamentais do processo de gerenciamento de projetos que é o **planejamento**.
- A partir deste, passamos a identificar a quantidade de esforço, o custo e as atividades que serão necessárias para a realização do projeto.

Medição de Software

- **Medição de software** é uma avaliação quantitativa de qualquer aspecto dos processos e produtos da Engenharia de Software, que permite seu melhor entendimento e, com isso, auxilia o planejamento, controle e melhoria do que se produz de como é produzido.
- O elemento básico da medição, que propicia a análise quantitativa, são as **medidas**. Elas caracterizam, em termos quantitativos, alguma propriedade de um objeto da Engenharia de Software.
- Provê informação útil para que as organizações tomem decisões que impactam em seus objetivos de negócios.
- A medição de software é um dos **principais pilares da melhoria de processos de software**.

Papéis da medição

- São quatro os principais papéis de Medições de Software:



Objetivos da Medição de Software e utilidade das métricas



- **Entender:** ajudam a entender o comportamento e o funcionamento de produtos de software.
- **Controlar:** utilizadas para controlar processos, produtos e serviços de software.
- **Prever:** utilizadas para prever valores de atributos.
- **Avaliar:** utilizadas para determinar padrões, metas e critérios de aceitação.

Métricas de software

- As métricas são importantes para **entender** o comportamento e o funcionamento do produto ou do processo, para **controlar** os processos e serviços, para **prever** valores de atributos e para **tomar decisões** a partir de padrões, métricas e critérios.

Métricas de software

- As análises baseadas em métricas são mais eficientes do que as que utilizam informações subjetivas.
- Dados históricos das métricas também são usados para gerar estimativas, a partir de dados de projetos anteriores, além da comparação de projetos, visando o aumento da qualidade e da produtividade do processo, que são exigências cada vez maiores nas organizações.

Métricas diretas

- Podemos considerar como de **métricas diretas**:
 - Custo;
 - Esforço aplicado para o desenvolvimento e manutenção;
 - Quantidade de linhas de código produzidas (LOC);
 - Velocidade de execução;
 - Tamanho de memória ocupada;
 - Número total de defeitos/erros registrados num determinado período de tempo;
 - Entre outros.

Métricas indiretas

- Porém existem critérios difíceis de serem avaliados, e só é possível medir de **forma indireta**, como por exemplo:
 - Funcionalidade;
 - Qualidade;
 - Complexidade;
 - Eficiência;
 - Confiabilidade;
 - Manutenibilidade,
 - Entre outros.

Por que medir software?

- Entender e aperfeiçoar o processo de desenvolvimento
- Melhorar a gerência de projetos e o relacionamento com clientes
- Gerenciar contratos de software
- Indicar a qualidade de um produto de software
- Avaliar a produtividade do processo
- Avaliar os benefícios (em termos de produtividade e qualidade) de novos métodos e ferramentas de engenharia de software
- Avaliar retorno de investimento

Por que medir software?

- Identificar as melhores práticas de desenvolvimento de software
- Embasar solicitações de novas ferramentas e treinamento
- Avaliar o impacto da variação de um ou mais atributos do produto ou do processo na qualidade e/ou produtividade
- Melhorar a exatidão das estimativas
- Oferecer dados qualitativos e quantitativos ao gerenciamento de desenvolvimento de software, de forma a realizar melhorias em todo o processo de desenvolvimento de software

Por que é tão difícil estimar?



- Desenvolvimento é um processo gradual de refinamento:
 - Incerteza da natureza do produto contribui para a incerteza da estimativa
 - Requisitos e escopo mudam
 - Defeitos são encontrados e demandam retrabalho
 - Produtividade varia

1. Estimar o tamanho do produto
2. Estimar o esforço
3. Estimar o prazo
4. Fornecer estimativas dentro de uma faixa permitida e refinar essa faixa à medida que o projeto progride

Principais Barreiras

- Falta de comprometimento da alta gerência
- Medir custa caro
- Os maiores benefícios vêm a longo prazo
- Má utilização das métricas
- Grande mudança cultural necessária
- Dificuldade de estabelecer medições apropriadas e úteis
- Interpretações dos dados realizadas de forma incorreta
- Obter o comprometimento de todos os envolvidos e impactados
- Estabelecer um programa de medições é fácil, o difícil é manter!!

Mas podemos contornar ...



- Foco desde os estágios iniciais da melhoria de processo
- Medição faz parte do TODO
- Começar Pequeno
- Selecionar um conjunto coerente
- É importante definir cada detalhe da métrica
- Descartar o que não estiver sendo útil
- Fornecer as informações corretas, para as pessoas certas
- “Agregar valor”, ao invés de gerar apenas dados
- Incentivar a equipe de desenvolvimento a fazer uso das métricas
- Estabelecer as expectativas
- Educação e Treinamento
- Compreender que a Adoção leva Tempo

Classificação das métricas

Ponto de vista da medição

- **As métricas de software, do ponto de vista de medição, podem ser divididas em duas categorias:**
 - Diretas
 - Indiretas
- As medições diretas são de obtenção relativamente simples, desde que estabelecidas as convenções específicas para isto
 - (Ex. custo e número de defeitos)
- As medições indiretas são difíceis de quantificar
 - (Ex. funcionalidade, complexidade e eficiência).

- É uma medida da taxa na qual os engenheiros individuais envolvidos no desenvolvimento de software produzem o software e sua documentação associada.
- Não é orientada à qualidade, embora a garantia de qualidade seja um fator da avaliação de produtividade.
- Essencialmente, queremos medir a funcionalidade útil produzida por unidade de tempo.

Medidas de produtividade

- **Medidas relacionadas a tamanho** são baseadas em alguma saída do processo de software. Podem ser linhas de código fonte entregues, instruções de código objeto, etc.
- **Medidas relacionadas a funções** são baseadas na estimativa da funcionalidade do software entregue. Pontos de função é a mais conhecida desse tipo de medida.

Métricas orientada ao tamanho

- São medidas diretas do tamanho dos artefatos de software associados ao processo por meio do qual o software é desenvolvido.
- Algumas métricas orientadas a tamanho:
 - LOC (Lines Of Code - Linhas de Código)
 - COMOCO

Medidas relacionadas a tamanho LOC - Linhas de Código



- As primeiras tentativas de se medir o tamanho de um sistema levou em consideração as LOCs (Lines Of Code - Linhas de Código).
- É baseada na ideia de que **“um sistema que possuir um número maior de LOCs que um outro, portanto, é maior e mais complexo”**.

Vantagem:

- Fácil de calcular;

Desvantagens:

- Está fortemente ligado à linguagem de programação utilizada e ao estilo de código escrito;
- Impossibilidade de utilização de dados históricos para projetos que não utilizam a mesma linguagem/tecnologia;
- Métrica pouco significativa para o cliente;
- A medida de linhas de código não deveria contar linhas de comentário e linhas em branco, pois não afeta a sua funcionalidade.

Medidas relacionadas a função - Análise de Pontos de Função



- A técnica de Análise de Pontos de Função - APF (Function Point Analysis) mede uma aplicação através das funções desempenhadas para/e por solicitação do usuário final.
- Baseada na **visão do usuário**, a APF é independente de tecnologia e pode ser utilizada para estimativas.
- A técnica de Análise de Pontos de Função mede “o que” é o sistema e não “como” será, ou foi, desenvolvido.
- Um dos principais conceitos relativos a APF é que as funções devem ser contadas a partir da perspectiva do usuário e não do analista ou programador.

Características:

- Baseada na visão de negócio do usuário;
- É independente da linguagem utilizada e de qualquer tecnologia em geral;
- Ela não permite calcular o esforço de desenvolvimento, mas gera uma variável que pode permitir seu cálculo;
- Auxilia o usuário final a melhorar o exame e avaliação de projetos.

Seus objetivos são:

- Medir as funcionalidades do sistema requisitadas e recebidas pelo usuário;
- Prover uma métrica de medição para apoiar a análise de produtividade e qualidade;
- Prover uma forma de estimar o tamanho do software;
- Prover um fator de normalização para comparação de software.

- O grupo responsável pela padronização denomina-se IFPUG (International Function Point Users Group), o qual tem procurado difundir esta técnica e padronizar os conceitos inerentes a ela.
- Recentemente a ISO (International Organization for Standardization) e a IEC (International Electrotechnical Commission) criaram um grupo para normalizar o processo de mensuração de software, cuja proposição inicial está baseada nesta técnica.

Contratação de serviços pelo governo federal

- Desde 2009 o governo federal praticamente extinguiu a aferição de esforço por meio de métrica Homem/hora, o qual foi feito através de uma instrução normativa nº 4 expedida pela Secretaria de Logística e Tecnologia de janeiro de 2009, e posteriormente pela Portaria SLTI/MP nº 31, de 29 de novembro de 2010.

SECRETARIA DE LOGÍSTICA E TECNOLOGIA DA INFORMAÇÃO

PORTARIA SLTI/MP Nº 31, DE 29 NOVEMBRO DE 2010

Dispõe sobre recomendações técnicas para a utilização da métrica Análise de Ponto de Função no âmbito da Administração Pública Federal direta, autárquica e fundacional e dá outras providências.

A SECRETARIA DE LOGÍSTICA E TECNOLOGIA DA INFORMAÇÃO DO MINISTÉRIO DO PLANEJAMENTO, ORÇAMENTO E GESTÃO, no uso de suas atribuições que lhe conferem o Decreto nº 7.063, de 13 de janeiro de 2010, o Decreto nº 1.048, de 21 de janeiro de 1994, e o Decreto nº 1.094, de 23 de março de 1994, resolve:

Art. 1º A métrica de Pontos de Função foi concebida como uma medida de tamanho funcional para projetos de desenvolvimento e de melhoria (manutenção evolutiva) de software.

§ 1º A métrica Ponto de Função é definida pelo organismo International Function Point Users Group (IFPUG).

§ 2º O manual de práticas de contagem de Pontos de Função publicado pelo IFPUG define as regras básicas orientativas de contagem de Pontos de Função para projetos de desenvolvimento e melhoria de soluções de software.

§ 3º Por permitir a medição objetiva de serviços de desenvolvimento de soluções de software, sua utilização é uma boa prática na contratação de serviços e está aderente ao estabelecido na Instrução Normativa SLTI nº 4 de 12 de novembro de 2010.

Art. 2º O Roteiro de Métricas de Software do SISP é um documento técnico complementar que visa esclarecer questões técnicas, harmonizar entendimento e abordar assuntos relativos à contratação de soluções de software não contempladas pelo manual de contagem do IFPUG.

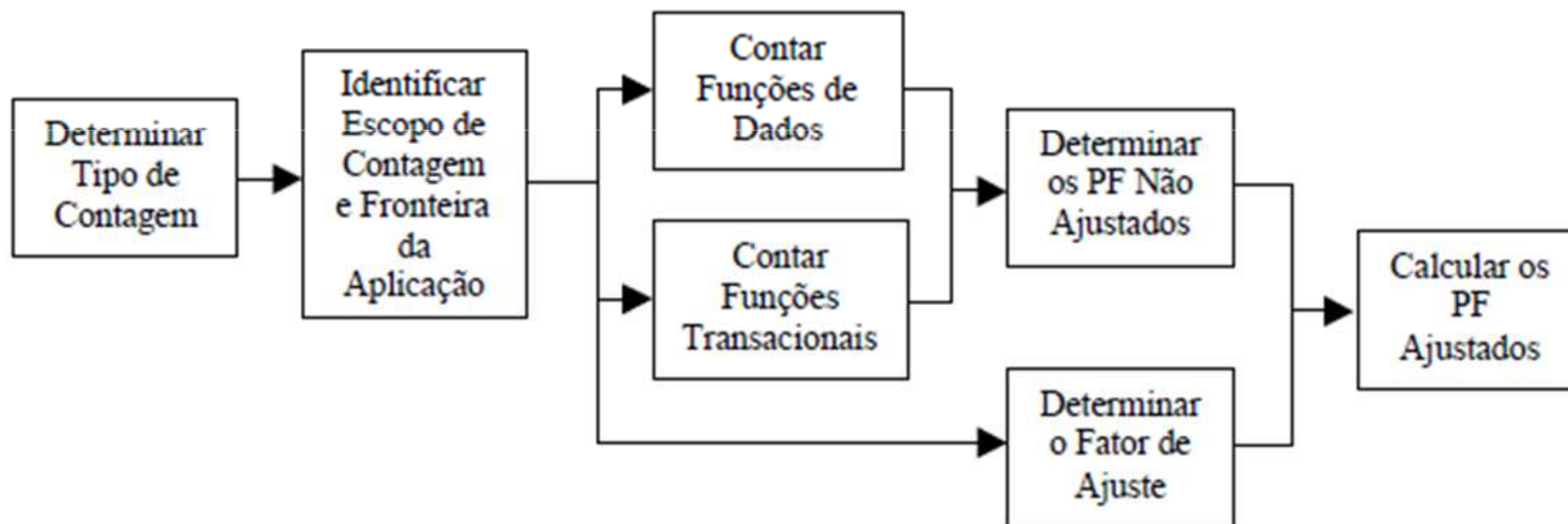
Parágrafo Único. Além dos projetos de desenvolvimento de novas soluções de software e de melhoria de software, também há necessidade de medir projetos de manutenção adaptativa de software. Assim, torna-se relevante a definição de procedimentos complementares de medição para dimensionar projetos de manutenção adaptativa de software cuja mensuração não são abordadas pelo manual de prática de contagem do IFPUG.

Art. 3º Recomenda-se que os órgãos integrantes do Sistema de Administração dos Recursos de Informação e Informática (SISP) adotem o roteiro de contagem nas suas contratações de serviços de desenvolvimento e manutenção de soluções de software.

Art. 4º Esta Portaria entra em vigor na data de sua assinatura.

MARIA DA GLÓRIA GUIMARÃES DOS SANTOS

- O Processo de Contagem de Pontos de Função



- O processo de contagem dos pontos de função constitui-se das seguintes etapas:
 1. Determinação do Tipo de Contagem
 2. Fronteiras da Aplicação
 3. Funções do Tipo Dados
 4. Funções do Tipo Transação
 5. Determinação dos Pontos de Função Brutos
 6. Determinação do Fator de Ajuste
 7. Cálculo dos Pontos de Função Ajustados

1. Determinar o Tipo de Contagem

- O primeiro passo para se fazer a contagem de pontos de função de uma aplicação é elencar que tipo de projeto estamos querendo contar dentre os três possíveis:
 - Projeto de Desenvolvimento: Contagem de pontos de função de uma aplicação nova, ainda não existente;
 - Projeto de Manutenção: Contagem de pontos de função de alterações em uma aplicação existente;
 - Projeto de Aplicação: Contagem de pontos de função de uma aplicação existente, instalada e em uso.

2. Identificar a Fronteiras da Aplicação

- É definido o escopo da aplicação com as funcionalidades a serem contadas.
- Neste passo, definem-se as funcionalidades que serão incluídas em uma contagem de PFs específica.
- A fronteira da aplicação é definida estabelecendo um limite lógico entre a aplicação que esta sendo medida, os usuários e outras aplicações.
- É de extrema importância a correta identificação da fronteira da aplicação antes do início da contagem, a fim de não ocorrer o erro de contar funções que não fazem parte do escopo ou deixar de fora funções que deveriam fazer parte do escopo.

3. Contar as Funções de Dados

- Nesse ponto será necessária a contagem de cada ALI (Arquivo Lógico Interno) e AIE (Arquivo de Interface Interna), identificando seu grau de complexidade.
- **ALI (Arquivo Lógico Interno):**
 - Consiste de um grupo lógico de dados ou informações de controle, referenciado pela aplicação e mantido fora da fronteira da aplicação que está sendo controlada.
 - Ou seja, representam as necessidades de grupos de dados logicamente relacionadas, utilizados pela aplicação, mas que sofrem manutenção a partir de outra aplicação;
 - *Por exemplo: as tabelas ou classes do sistema.*
- **AIE (Arquivo de Interface Externa):**
 - Consiste de um grupo lógico de dados ou informações de controle, referenciado pela aplicação e mantido fora da fronteira da aplicação que está sendo controlada.
 - Ou seja, representam as necessidades de grupos de dados logicamente relacionadas, utilizados pela aplicação, mas que sofrem manutenção a partir de outra aplicação;
 - *Por exemplo: as tabelas acessadas em outro sistema.*
- A diferença básica entre um ALI e um AIE é que o último **não é mantido** pela aplicação que está sendo contada.

4. Contar as Funções de Transação

- Da mesma forma que as funções de dados, as funções de transação CE (Consultas Externas), EE (Entradas Externas) e SE (Saídas Externas) devem ser contadas com identificação de seu tamanho.

Entradas Externas (EE):

- Processo elementar da aplicação que processa dados ou informações de controle que vêm de fora da fronteira da aplicação que está sendo controlada.
- Ou seja, representam as atividades vindas diretamente do usuário, através de um processo lógico único, com o objetivo de inserir, modificar ou remover dados dos arquivos lógicos internos.
- Tais dados podem vir de uma tela de entrada de dados ou através de outro aplicativo.
- *Exemplos: inclusão de dados numa tabela, alteração de dados de um tabela ou exclusão de dados de uma tabela.*

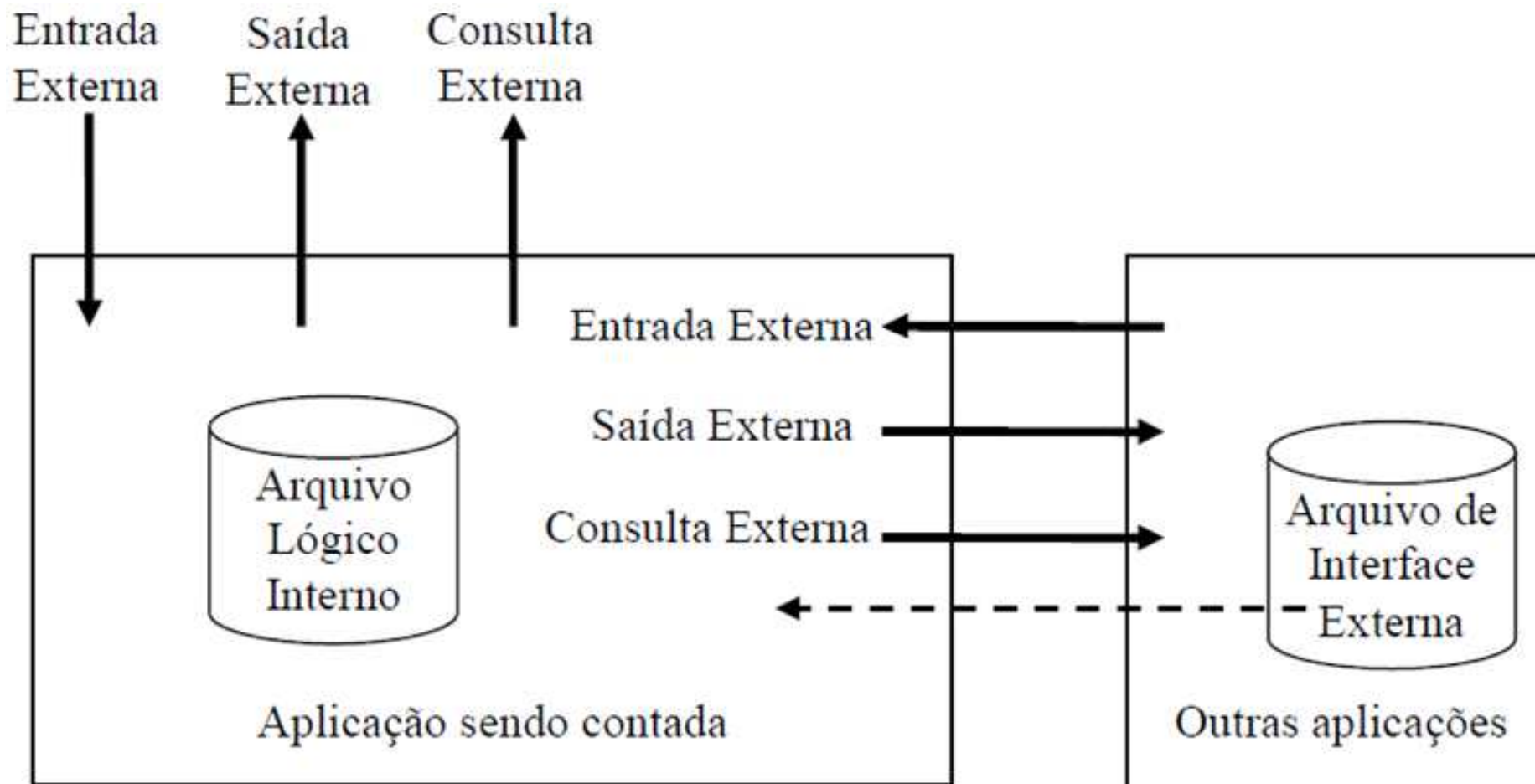
Saídas Externas (SE):

- Processo elementar da aplicação que gera dados ou informações de controle que são enviados para fora da fronteira da aplicação que está sendo controlada.
- Ou seja, representam as atividades da aplicação (processos) que têm como resultado a extração de dados da aplicação.
- *Exemplos: relatórios e gráficos.*

Consultas Externas (CE):

- Processo elementar da aplicação que representa uma combinação de entrada (solicitação de informação) e saída (recuperação de informação).
- Ou seja, representam as atividades que, através de uma requisição de dados (entrada), gera uma aquisição e exibição imediata dos dados (saída).
- *Exemplos: consultas implícitas, verificação de senhas e recuperação de dados com base em parâmetros.*

Visão Geral da APF



5. Determinação dos Pontos de Função Brutos

- Todo ponto de função é considerado não ajustado ou bruto por haver variação quanto ao tipo de sistema implementado, ambiente de uso e outras variáveis que podem influenciar no seu tamanho.
- Com base na contagem obtida nos itens 3 e 4, devemos contar os PFNA's (Pontos de função não ajustados), que é a contagem bruta dos pontos de função.
- Cada tipo de função (ALI, AIE, EE, SE, CE) é avaliado, recebendo complexidade Baixa, Média ou Alta.

Tipo de Função	Simples	Média	Comple-xa
EE	X 3	X 4	X 6
SE	X 4	X 5	X 7
CE	X 3	X 4	X 6
ALI	X 7	X 10	X 15
AIE	X 5	X 7	X 10

Complexidade dos ALI e AIE

- Número de itens de dados (ID)
 - *Campos da tabela*
- Número de registros lógicos (RL)
 - *Chaves estrangeiras/relacionamentos*

Tipo de Função	Simple	Média	Comple- xa
EE	X 3	X 4	X 6
SE	X 4	X 5	X 7
CE	X 3	X 4	X 6
ALI	X 7	X 10	X 15
AIE	X 5	X 7	X 10

Número de itens de dados			
Número de Registros lógicos	1 a 19 ID	20 a 50 ID	51 ou mais ID
1 RL	SIMPLES	SIMPLES	MÉDIA
2 a 5 RL	SIMPLES	MÉDIA	COMPLEXA
6 RL ou mais	MÉDIA	COMPLEXA	COMPLEXA

Complexidade da EE (Entrada Externa)

- Número de itens de dados (ID)
 - *Campos da tabela*
- Número de arquivos referenciados (AR)
 - *Número de tabelas*

Tipo de Função	Simples	Média	Comple- xa
EE	X 3	X 4	X 6
SE	X 4	X 5	X 7
CE	X 3	X 4	X 6
ALI	X 7	X 10	X 15
AIE	X 5	X 7	X 10

Número de itens de dados			
Número de Registros lógicos	1 a 4 ID	5 a 15 ID	16 ou mais ID
0 ou 1 AR	SIMPLES	SIMPLES	MÉDIA
2 AR	SIMPLES	MÉDIA	COMPLEXA
3 AR ou mais	MÉDIA	COMPLEXA	COMPLEXA

Complexidade da SE (Saída Externa)

- Número de itens de dados (ID)
 - *Campos da tabela*
- Número de arquivos referenciados (AR)
 - *Número de tabelas*

Tipo de Função	Simples	Média	Comple- xa
EE	X 3	X 4	X 6
SE	X 4	X 5	X 7
CE	X 3	X 4	X 6
ALI	X 7	X 10	X 15
AIE	X 5	X 7	X 10

Número de itens de dados			
Número de Registros lógicos	1 a 4 ID	5 a 15 ID	16 ou mais ID
0 ou 1 AR	SIMPLES	SIMPLES	MÉDIA
2 ou 3 AR	SIMPLES	MÉDIA	COMPLEXA
4 AR ou mais	MÉDIA	COMPLEXA	COMPLEXA

Complexidade da CE (Consulta Externa)

- Número de itens de dados (ID)
 - *Campos da tabela*
- Número de arquivos referenciados (AR)
 - *Número de tabelas*

Tipo de Função	Simples	Média	Comple- xa
EE	X 3	X 4	X 6
SE	X 4	X 5	X 7
CE	X 3	X 4	X 6
ALI	X 7	X 10	X 15
AIE	X 5	X 7	X 10

Número de itens de dados			
Número de Registros lógicos	1 a 5 ID	6 a 19 ID	20 ou mais ID
0 ou 1 AR	SIMPLES	SIMPLES	MÉDIA
2 ou 3 AR	SIMPLES	MÉDIA	COMPLEXA
4 AR ou mais	MÉDIA	COMPLEXA	COMPLEXA

6. Determinação do Fator de Ajuste

- O passo final na contagem de pontos de função envolve o ajuste da contagem através de um Fator de Ajuste de Valor (FAV), o qual avalia restrições de negócio adicionais do software não consideradas pelos cinco tipos de funções.
- Para ajustar os pontos de função de acordo com o tipo de sistema que estamos desenvolvendo, são consideradas 14 características que deverão ser contadas de acordo com seu nível de influência, sendo considerado 0 para o nível mais baixo de influência até 5 para o nível mais alto.
- Feita a identificação de cada característica, calcularemos o NIT (nível de influência total).
- O NIT indica a variação dada pelas características da aplicação e será utilizado para ajuste dos pontos de função.

- O fator de ajuste é calculado a partir de 14 características gerais dos sistemas, que permitem uma avaliação geral da funcionalidade da aplicação. As características gerais de um sistema são:

- 1 – Comunicação de Dados
- 2 – Processamento Distribuído
- 3 – Performance
- 4 – Utilização de Equipamento
- 5 – Volume de Transações
- 6 – Entrada de Dados "on-line"
- 7 – Eficiência do Usuário Final
- 8 – Atualização "on-line"
- 9 – Processamento Complexo
- 10 – Reutilização de Código
- 11 – Facilidade de Implantação
- 12 – Facilidade Operacional
- 13 – Múltiplos Locais
- 14 – Facilidade de Mudanças

Atribui-se um peso de 0 a 5 para cada característica, de acordo com o seu nível de influência (NI) na aplicação:

- 0 - Nenhuma influência;**
- 1 - Influência mínima;**
- 2 - Influência moderada;**
- 3 - Influência média;**
- 4 - influência significativa;**
- 5 - Grande influência.**

$$\text{Fator de Ajuste} = (NI * 0,01) + 0,65$$

7. Calcular Pontos de Função Ajustados

- Uma vez calculados os PF não ajustados e o fator de ajuste, é possível calcular os PFs ajustados.
- Para projetos de desenvolvimento, o cálculo é dado por:

Pontos de Função ajustados = PFNA * Fator de ajuste

onde

PFNA = Número de PFs não ajustados

Resumindo:

- 1º) Definir tipo de contagem (exemplo: nova aplicação)
- 2º) Definir claramente o escopo do projeto
- 3º) Identificação das funções de dados (banco de dados):
 - ALI – Arquivo lógico interno
 - AIE – Arquivo de Interface Externa
- 4º) Identificação das transações de dados
 - EE – Entrada externa
 - SE – Saída externa
 - CE – Consulta externa
- 5º) Fazer contagem dos Pontos de Função Brutos
- 6º) Definir fator de ajuste
- 7º) Calcular pontos de função ajustados

Contagem dos pontos de função não ajustados

TIPO DE FUNÇÃO	COMPLEXIDADE FUNCIONAL	TOTAL COMPLEX.	TOTAL TIPO FUNÇÃO
ARQUIVO	SIMPLES X 7 = MÉDIA X 10 = COMPLEXA X 15 =		
INTERFACE	SIMPLES X 5 = MÉDIA X 7 = COMPLEXA X 10 =		
ENTRADA	SIMPLES X 3 = MÉDIA X 4 = COMPLEXA X 6 =		
SAÍDA	SIMPLES X 4 = MÉDIA X 5 = COMPLEXA X 7 =		
CONSULTA	SIMPLES X 3 = MÉDIA X 4 = COMPLEXA X 6 =		
* * * TOTAL DE PONTOS DE FUNÇÃO NÃO - AJUSTADOS =			

EXEMPLO

- Calcule o custo do sistema seguinte, levando em consideração que em cada hora de trabalho consegue-se realizar dois pontos de função e que cada hora/pessoa custa M\$ 80,00 (oitenta unidades monetárias).
- Um levantamento de dados realizado por analistas de sistemas apresentou os seguintes resultados em relação a um sistema de gerenciamento de leitos hospitalares:
 - As informações internas do sistemas estão agrupadas no cadastro de diagnósticos e leitos (simples) e de remédios (médio).
 - As informações externas a serem utilizadas pelo sistema são: Cadastro de médicos e de pacientes (simples).
 - Os principais relatórios do sistema são de histórico do paciente (complexa) e de ocupação de leitos (simples).
 - As principais entradas do sistema são: Incluir, excluir e alterar os dados dos cadastros internos.
 - Todas as entradas são consideradas simples.
 - O sistema permite consulta da disponibilidade de remédios, disponibilidade de leitos e de horários de trabalho dos médicos.
 - Todas as consultas são de complexidade média.
 - A comunicação de dados é crítica, portanto de grande influência.
 - O volume de transações e a eficiência do usuário final possuem significativa influência.
 - Há uma preocupação moderada em reutilização de código.
 - O sistema será utilizado em vários ambientes de hardware e software, portando múltiplos locais possui influência significativa.
 - O restante das características terá influência mínima.

Exemplo

1) Dados gerais para resolução:

- Cada Hora de trabalho = 2 pontos de função
- Hora/Pessoa = M\$ 80,00

2) Pontos de Função Não Ajustados

- **Arquivos Lógicos Internos**

- Diagnósticos - Simples 7
- Leitos - Simples 7
- Remédios - Média 10

→ **Arquivos Lógicos Internos $\Rightarrow 2 \times 7 + 1 \times 10 = 24$**

- **Arquivos de Interface Externa**

- Médicos - Simples 5
- Pacientes - Simples 5

→ **Arquivos de Interface Externa $\Rightarrow 2 \times 5 = 10$**

- **Entradas Externas**

- Incluir Diagnósticos - Simples 3
- Incluir Leitos - Simples 3
- Incluir Remédios - Simples 3
- Excluir Diagnósticos - Simples 3
- Excluir Leitos - Simples 3
- Excluir Remédios - Simples 3
- Alterar Diagnósticos - Simples 3
- Alterar Leitos - Simples 3
- Alterar Remédios - Simples 3

→ **Entradas Externas => 9X3=27**

- **Saídas Externas**

- Histórico do Paciente - Complexa 7
- Ocupação de Leitos - Simples 4
- **Saídas Externas => 1X7 + 1X4 = 11**

- **Consultas Externas**

- Disponibilidade de Remédios - Média 4
- Disponibilidade de Leitos - Média 4
- Horário de Trabalho dos Médicos - Média 4
- **Consultas Externas => 3X4 = 12**

Total de Pontos de Função Não Ajustados

- Arquivos Lógicos Internos 24
- Arquivos de Interface Externa 10
- Entradas Externas 27
- Saídas Externas 11
- Consultas Externas 12
- **TOTAL 84**



<i>Características</i>	<i>Nível de influência</i>	<i>Peso</i>
Comunicação de Dados	Grande influência	5
Processamento Distribuído	Nenhuma influência	0
Performance	Nenhuma influência	0
Utilização de Equipamento	Nenhuma influência	0
Volume de Transações	Influência significativa	4
Entrada de Dados "on-line"	Nenhuma influência	0
Eficiência do Usuário Final	Influência significativa	4
Atualização "on-line"	Nenhuma influência	0
Processamento Complexo	Nenhuma influência	0
Reutilização de Código	Influência mínima	2
Facilidade de Implantação	Nenhuma influência	0
Facilidade Operacional	Nenhuma influência	0
Múltiplos Locais	Influência significativa	4
Facilidade de Mudanças	Nenhuma influência	0

→ **TOTAL 28**

Fator de Ajuste = (NI * 0,01) + 0,65

Fator de Ajuste = (19 * 0,01) + 0,65

Fator de Ajuste = 0,84

4. Pontos de Função Ajustados

- PF Ajustados = PF não ajustados * Fator de Ajuste
- PF Ajustados = 84 * 0,84
- PF Ajustados = 70,56

5. Custo do Sistema

- Custo do sistema = (PF Ajustados / Pontos de função por hora) * Custo hora
- Custo do sistema = (70,56/2) * 80
- Custo do sistema = M\$ 2.822,40

TRABALHO INDIVIDUAL

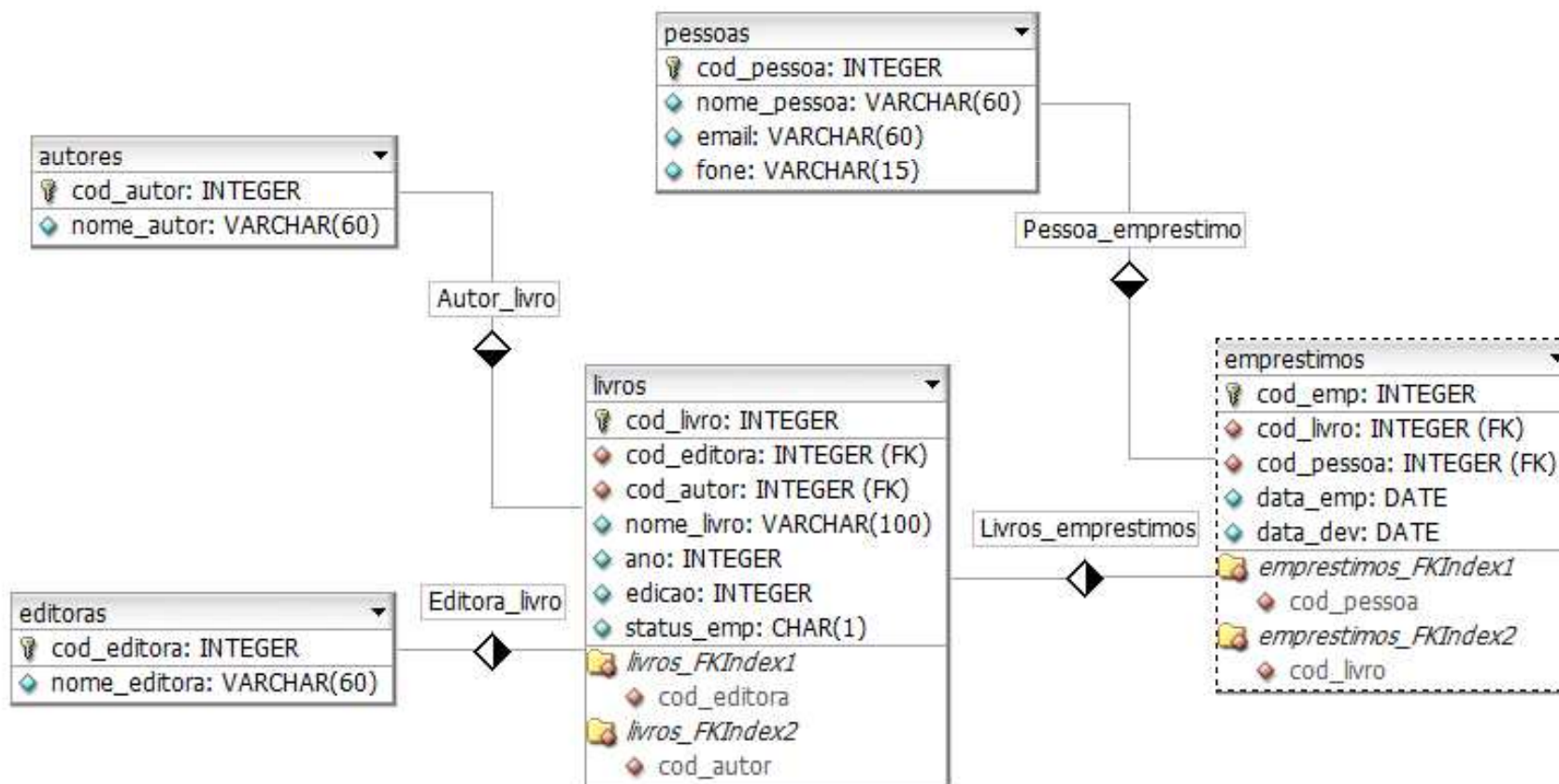
Considere um sistema de catálogo de livros.

- 1. Calcule os pontos de função ajustados AFP.
- 2. Calcule o tempo de desenvolvimento considerando que cada hora de trabalho consegue-se realizar dois pontos de função .
- 3. Calcule o custo do projeto considerando um custo de R\$40,00 por hora trabalhada.

Descrição do trabalho

Objetivo: informatizar um catálogo pessoal de livros

- O Modelo entidade relacionamento abaixo apresenta o modelo de dados do sistema



Descrição do trabalho

- **O sistema deverá ter:**
 - Cadastro com inclusão, alteração e exclusão de todas tabelas as sistema: autores, editoras, livros, pessoas e empréstimos.
 - Relatório geral de: livros, autores, editoras, pessoas e empréstimos.
 - Consulta de livros por autor e editora, conforme for informado pelo usuário
 - Consulta de empréstimos por livro mostrando todos os dados da reserva
 - A comunicação de dados é crítica, portanto de grande influência.
 - O sistema deverá ser acessado e consultado através da web