**APF**

Análise de Pontos de Função (APF) é uma**técnica de medição das funcionalidades fornecidas por um software do ponto de vista de seus usuários.**

Ponto de função (PF) é a sua **unidade de medida**, que tem por objetivo tornar a medição **independente** da tecnologia utilizada para a construção do software.

 A APF mede o que o software **faz**, independentemente de como ele foi construído.

As métricas de software podem ser classificadas em medidas diretas (**quantitativas**) **e** medidas indiretas (**qualitativas**). Pontos de função são derivados por meio de uma relação empírica baseada em medidas calculáveis(diretas) do domínio de informações do software e avaliações qualitativas da complexidade do software. Ou seja, a métrica de pontos de função é quantitativa e qualitativa. Quantitativa por se basear em medidas calculáveis do domínio de informações do software. Qualitativa por se basear em avaliações qualitativas da complexidade do software.

O processo de medição ou de contagem de pontos de função é baseado na avaliação dos requisitos funcionais do usuário, como descrito nos artefatos do projeto, cujo procedimento está descrito pelo IFPUG em seu Manual de Práticas de Contagem (Counting Practices Manual).

É importante destacar que pontos de função não medem diretamente esforço, produtividade ou custo, sendo uma medida de tamanho funcional do software. Porém, a partir do tamanho funcional, correlacionando com outras variáveis torna-se possível identificar produtividade, estimar esforço e/ou custo de projetos de software. Pontos de função são exclusivamente para mensurar o tamanho funcional do software, mas esse tamanho, junto com **dados históricos** pode sim ser usado para obtenção de outras informações como custo e etc.

A **métrica ponto de função** pode ser usada efetivamente como um meio para medir a funcionalidade fornecida por um sistema. Segundo Pressman, por meio de **dados históricos**, a métrica FP pode ser empregada para:

* Estimar o custo ou trabalho necessário para projetar, codificar e testar o software;
* Prever o número de erros que serão encontrados durante o teste;
* Prever o número de componentes e/ou o número de linhas projetadas de código-fonte no sistema implementado.

A análise por ponto de função tem como principal objetivo medir a funcionalidade do sistema tendo como base a visão do usuário, de acordo com as seguintes características:

* É independente da tecnologia utilizada;
* Auxilia a produção de resultados consistentes;
* Baseia-se na visão do usuário;
* Tem significado para o usuário final;
* Utiliza-se de estimativas;
* Passível de automação;
* Dificuldade por possuir relativa subjetividade por refletir a visão do usuário.

Três tipos de contagens são possíveis:

* Contagem de projeto de desenvolvimento;
* Contagem de projeto de melhoria (manutenção);
* Contagem de aplicação.

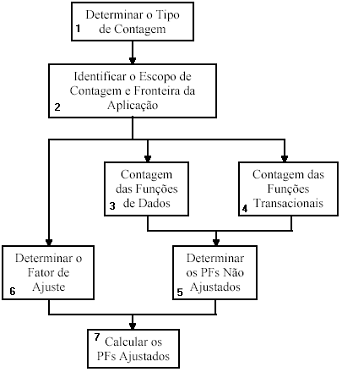
De maneira simplificada, o primeiro mede a funcionalidade fornecida aos usuários finais quando o projeto estiver pronto, no momento de sua instalação. Essa contagem também abrange conversão de dados necessários para implantação do software.

O segundo mede as modificações para uma aplicação já existente, o que inclui as funções incluídas, alteradas e excluídas do sistema pelo projeto, além das funções de conversão de dados. É importante lembrar que, sempre após uma contagem de manutenção e sua implantação, o número de pontos de função da aplicação deve ser atualizado de acordo com as alterações feitas na funcionalidade. Isso tem como objetivo **manter a contagem de pontos de função** (CPF) da aplicação sempre atualizada.

O terceiro e último tipo de contagem mede a funcionalidade fornecida ao usuário pela aplicação instalada e em produção para que a atual funcionalidade tenha uma medida.

O processo de contagem de pontos de função pode ser composto pelos seguintes passos:

1. **Obter a documentação disponível do projeto** (Identifique os requisitos funcionais);
2. **Identificar o propósito da contagem** (ex: necessidade de dimensionar um projeto de um novo sistema para auxiliar o processo de contratação do mesmo.);
3. **Identificar o tipo da contagem** (melhoria, desenvolvimento ou aplicação);
4. **Determinar o escopo da contagem** (quais funcionalidades serão incluídas na contagem de pontos de função);
5. **Determinar a fronteira da aplicação** (limite lógico entre o sistema sendo medido e os usuários);
6. **Contar funções de dados e funções de transacionais**;
7. **Calcular o tamanho funcional**;
8. **Documentar e reportar a contagem**.



**Informação de Controle -** Dados que influenciam um processo elementar da aplicação sendo contada. Especifica o que, quando ou como os dados devem ser processados. Em resumo, são parâmetros.

**ALR – Arquivo Logico Referenciado** - É um [arquivo lógico interno](https://ead.fattocs.com/mod/glossary/showentry.php?eid=379&displayformat=dictionary) ([ALI](https://ead.fattocs.com/mod/glossary/showentry.php?eid=379&displayformat=dictionary)) lido ou [mantido](https://ead.fattocs.com/mod/glossary/showentry.php?eid=539&displayformat=dictionary) pela [função transacional](https://ead.fattocs.com/mod/glossary/showentry.php?eid=407&displayformat=dictionary) ou um [arquivo de interface externa](https://ead.fattocs.com/mod/glossary/showentry.php?eid=366&displayformat=dictionary) ([AIE](https://ead.fattocs.com/mod/glossary/showentry.php?eid=366&displayformat=dictionary)) lido pela [função transacional](https://ead.fattocs.com/mod/glossary/showentry.php?eid=407&displayformat=dictionary). Também chamado de [Arquivo Lógico](https://ead.fattocs.com/mod/glossary/showentry.php?eid=429&displayformat=dictionary) Referenciado (ALR) ou File Type Referenced (FTR). A [complexidade funcional](https://ead.fattocs.com/mod/glossary/showentry.php?eid=375&displayformat=dictionary) de cada [EE](https://ead.fattocs.com/mod/glossary/showentry.php?eid=364&displayformat=dictionary), [SE](https://ead.fattocs.com/mod/glossary/showentry.php?eid=367&displayformat=dictionary) e [CE](https://ead.fattocs.com/mod/glossary/showentry.php?eid=365&displayformat=dictionary) é atribuída com base no número de arquivos referenciados e [tipos de dados](https://ead.fattocs.com/mod/glossary/showentry.php?eid=353&displayformat=dictionary).

Dentre os objetivos da APF está: Medir projetos de desenvolvimento e manutenção de SW, independente da tecnologia utilizada na implementação, de modo a acompanhar sua evolução. Pode-se usá-la para medir funcionalidades solicitadas antes do projeto de SW para estimar custo, tamanho, e ainda após o projeto, de modo a verificar os mesmos itens (custo e tamanho) e compará-los com o que foi originalmente estimado.

Basicamente o processo de medição funcional é o seguinte:

1. Reunir a documentação disponível;
2. Determinar o escopo e a fronteira da contagem;
3. **Medir as funções de dados (ALI e AIE);**
4. **Medir as funções de transação (CE - SE - EE);**
5. Calcular o tamanho funcional (Contagem de desenvolvimento; Aplicação; ou Melhoria);
6. Documentar e reportar a contagem de pontos de função; Ou seja;

Na contagem de pontos de função, as funções de dados são obtidas **antes** da contagem das funções de transações.

**Funções de dados:**

* **ALI (Arquivo Lógico Interno):** é um grupo de dados ou informações de controle, reconhecido pelo usuário e mantido **dentro** da fronteira de aplicação sendo medida. A principal intenção de um ALI é armazenar dados mantidos por um ou mais processos elementares da aplicação sendo medida. Os **ALIs** são grupos de dados logicamente relacionados, ou informações de controle, mantidos **dentro** da fronteira da aplicação. ALI - Dados Mantidos.
* **AIE (Arquivo de Interface Externa):** é um grupo de dados ou informações de controle, reconhecido pelo usuário, e que é apenas referenciado pela aplicação sendo medida, mas que são mantidos dentro da fronteira de **outra aplicação**. A principal intenção de um AIE é armazenar dados referenciados por um ou mais processos elementares da aplicação sendo medida. Isto significa que um AIE contado para uma aplicação deve ser obrigatoriamente um ALI em alguma outra aplicação. **AIE - Dados Referenciados**. Repitam SEMPRE esse mantra: "**um arquivo de interface externa (AIE) SEMPRE será um arquivo lógico interno (ALI) de outra aplicação.**".

**Funções de transação:**

* As **EEs**são processos elementares que **processam dados**ou **informações de controle** que vêm de**fora das fronteiras da aplicação**. Uma entrada externa é um processo de controle, ela também realiza o processamento de dados do sistema e direciona o mesmo para atender os requisitos da aplicação. Qualquer ação que vá alterar o estado do sistema é considerado uma EE. Se for feita uma atualização de algum registro, vai alterar o estado do sistema. Se for feita uma inserção de um novo registro, vai alterar o estado do sistema. Se for feita uma exclusão de registro, vai alterar o estado do sistema.
* As **CEs**, como são conhecidas as Consultas Externas, são processos elementares que enviam dados ou informações de controle para **fora** das fronteiras da aplicação. Processo elementar que apresenta informação ao usuário ou a outra aplicação externa por meio de **recuperação simples.**
* As **SEs** são processos elementares que enviam dados ou informações de controle para **fora** das fronteiras da aplicação. Processo elementar destinado à apresentação de informação ao usuário ou a outra aplicação externa que **utiliza de cálculos** para processar essas informações.

Outra definição:

Tipos de funções estáticas  
1- Arquivo lógico interno (**ALI**): dentro do sistema.   
2- Arquivo de interface externa (**AIE**): Referenciado pela aplicação e mantido dentro da fronteira de outra aplicação.

b) Tipos de funções dinâmicas  
1- Entradas externas (**EE**): alteração do estado interno das informações do sistema.  
2- Saídas externas (**SE**): podem ser precedidas ou não de parâmetros. Pelo menos um dado deve ser calculado.  
3- Consultas externas (**CE**): são saídas que não mexem com dados.

Resumo:

* Inserir/update - entrada externa
* Ler valores - Consulta externa
* Ler valor(es) alterado(s) - saída externa

**Tipos de Contagem**

**1. A contagem detalhada é a contagem usual de pontos de função e é realizada da seguinte forma:**

* Determinam-se todas as funções de todos os tipos (ALI, AIE, EE, SE, CE).
* Determina-se a complexidade de cada função (Baixa, Média, Alta).
* Calcula-se o total de pontos de função não ajustados.

**2. A contagem estimativa é realizada da seguinte forma:**

* Determinam-se todas as funções de todos os tipos (ALI, AIE, EE, SE, CE)
* Toda função do tipo dado (ALI, AIE) tem sua complexidade funcional avaliada como Baixa, e toda função transacional (EE, SE, CE) é avaliada como de complexidade média.
* Calcula-se o total de pontos de função não ajustados.

Logo, a única diferença em relação à contagem usual de pontos de função é que a complexidade funcional não é determinada individualmente para cada função, mas pré-definida para todas elas.

**3. A contagem indicativa é realizada da seguinte forma:**

* Determina-se a quantidade das funções do tipo dados (ALIs e AIEs).
* Calcula-se o total de pontos de função não ajustados da aplicação da seguinte forma:

tamanho indicativo (pf) = 35 x número de ALIs + 15 x número de AIEs

Portanto esta estimativa é baseada somente na quantidade de arquivos lógicos existentes (ALIs e AIEs).

A contagem indicativa é baseada na premissa de que existem aproximadamente três EEs (para adicionar, alterar, e excluir dados do ALI), duas SEs, e uma CE na média para cada ALI, e aproximadamente uma SE e uma CE para cada AIE.

***A NESMA reconhece três tipos de contagem de pontos de função:***

Contagem de pontos de função detalhada;

Contagem de pontos de função estimativa;

Contagem de pontos de função indicativa;

* A contagem **detalhada** é a contagem usual de pontos de função e é realizada da seguinte forma:

\* determinam-se todas as funções de todos os tipos (ALI, AIE, EE, SE, CE);  
\* determina-se a complexidade de cada função (Baixa, Média, Alta);  
\* calcula-se o total de pontos de função não ajustados;

* A contagem **estimativa** é realizada da seguinte forma:

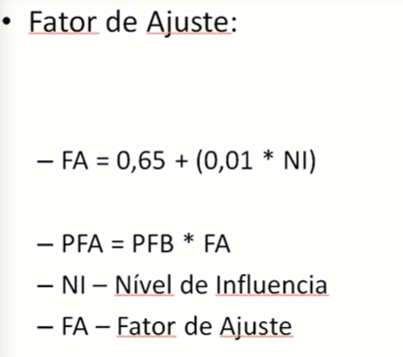
\* determina-se todas as funções de todos os tipos (ALI, AIE, EE, SE, CE)  
\* toda função do tipo dado (ALI, AIE) tem sua complexidade funcional avaliada como Baixa,  
\* e toda função transacional (EE, SE, CE) é avaliada como de complexidade média  
\* calcula-se o total de pontos de função não ajustados;

* A contagem **indicativa** é realizada da seguinte forma:

\* determina-se a quantidade das funções do tipo dado (ALIs e AIEs);

\* calcula-se o total de pontos de função não ajustados da aplicação da seguinte forma: tamanho indicativo (pf) = 35 x número de ALIs + 15 x número de AIEs.

**Tabela exemplo:**



**6.1.2 Estimativa de Esforço de Projetos de Software**

Uma vez que o tamanho do projeto foi estimado em pontos de função, **o próximo passo é estimar o esforço de desenvolvimento do projeto**, bem como sua distribuição pelas fases do ciclo de vida do desenvolvimento do software.

O Modelo Simplificado de Estimativas consiste em obter um índice de produtividade em horas/PF para o projeto específico em questão, e então multiplicar o tamanho em PF do projeto pelo índice de produtividade, conforme a fórmula [Vazquez, 2012]:

**Esforço (horas) = Tamanho (PF) x Índice de Produtividade (HH/PF)**

**Projetos de melhoria de software:**

* PF\_MELHORIA = PF\_INCLUIDO + (FI x PF\_ALTERADO) + (0,30 x PF\_EXCLUIDO) + PF\_CONVERSÃO - Sendo que o FI (Fator de Impacto)pode variar de 50% a 90%

**Projetos de migração de dados:**

* PF\_CONVERSÃO = PF\_INCLUIDO

**Projetos de desenvolvimento de software:**

* PF\_DESENVOLVIMENTO = PF\_INCLUIDO + PF\_CONVERSÃO

**Manutenção Corretiva**

* PF\_CORRETIVA = FI x PF\_ALTERADO

**Macete calculo APF**

Sequencia a ser decorada: 4-5-7-10-15

* **SE** = 4-5-7
* **EE-CE = SE-1 – (3-4-6)**
* **AIE** = 5-7-10
* **ALI** = 7-10-15

PFA = PFNA \* FA.

Os pontos de função ajustados equivalem aos pontos de função não-ajustados multiplicado pelo fator de ajuste calculado na etapa anterior.

***Diagrama de fluxo de Dados***

O **DFD nível 0** é também chamado de diagrama de contexto. É uma visão geral básica de todo o sistema ou processo a ser analisado ou modelado.

O **DFD nível 1** oferece maiores detalhes de peças do diagrama de contexto.

O **DFD nível 2** aprofunda ainda mais partes do nível 1.

A progressão aos níveis 3, 4 e acima é possível, mas é incomum ir além do nível 3. Ela pode criar complexidades que dificultam a comunicação, compararão ou modelagem de forma eficaz.

Componentes do diagrama de fluxo de dados:

**- Entidade externa:** um sistema externo que envia ou recebe dados, comunicando-se com o sistema sendo diagramado. É a fonte e o destino das informações que entram ou saem do sistema. Ela pode ser uma organização externa ou pessoa, um sistema de computador ou um sistema de negócios. Outros termos através dos quais também são conhecidos: terminadores; fontes e sumidouros; ou atores. É tipicamente desenhada nas bordas do diagrama.

**- Processo:**qualquer processo que altere os dados, produzindo uma saída. Pode realizar cálculos, classificar dados com base na lógica ou direcionar o fluxo de dados com base em regras de negócios. Um rótulo curto é utilizado para descrever o processo, tal como “Enviar o pagamento”.

**- Armazenamento de dados:** arquivos ou repositórios que armazenam informações para uso posterior, como uma tabela de banco de dados ou um formulário de adesão. Cada armazenamento de dados recebe um rótulo simples, como “Pedidos”.

- **Fluxo de dados:** a rota dos dados entre entidades externas, processos e armazenamentos de dados. Ele retrata a interface entre os outros componentes, é representado por setas e geralmente é rotulado com um nome de dados curto, como “Detalhes de faturamento”.

Bottom-Up consiste em uma abordagem que parte do mais detalhado para o genérico, enquanto que Top-Down é o inverso (geral para o específico).

O método da **Análise Essencial** de Sistemas preconiza que, de uma forma geral, um sistema deve ser modelado através de três dimensões:

\* **dados**: diz respeito aos aspectos estáticos e estruturais do sistema;

\* **controle**: leva em conta aspectos temporais e comportamentais do sistema;

\* **funções**: considera a transformação de valores;