

Roteiro de Métricas de Software da ANEEL - v1.0

Controle de Versão

Data	Versão	Descrição	Autor	Revisor	Aprovado por
24/09/2012	1.0	Emissão Inicial	João Celestino		

Sumário

1	Introdução	4
2	Objetivo	4
3	Métrica de Pontos de Função.....	5
3.1	Determinar o Escopo, as Fronteiras e identificar os Requisitos Funcionais do usuário.....	6
3.2	Função de Dados.....	6
3.3	Função de Transação	6
3.4	Calcular Tamanho Funcional	7
3.5	Fator de Ajuste.....	7
4	Métrica de Pontos de Função para a ANEEL.....	7
4.1	Contagem de Projeto de Desenvolvimento.....	9
4.2	Contagem de Projeto de Manutenção	10
4.2.1	Manutenção Evolutiva	10
4.2.2	Manutenção Corretiva	11
4.2.3	Manutenção Adaptativa	11
4.2.3.1	Mudança de Plataforma.....	12
4.2.3.2	Atualização de Plataforma - Browser/Linguagens de Programação.....	13
4.2.3.3	Atualização de Plataforma – Banco de Dados.....	13
4.2.3.4	Manutenção em Interface	13
4.2.3.5	Adaptação de Funcionalidades a Requisitos Não Funcionais	14
4.2.3.6	Projetos de Migração de Dados (Conversão de dados)	15
4.2.3.7	Apuração Especial	15
4.2.3.8	Atualização de Dados	17
4.2.3.9	Documentação de Sistemas Legados	17
4.2.3.10	Pontos de Função de Teste	18
4.2.3.11	Componente Interno Reusável	18
4.2.4	Contagem de Pontos de Função com Múltiplas Mídias.....	19
4.3	Cenários.....	20
4.3.1	Campos não editáveis	20
4.3.2	Subdivisão de uma funcionalidade em n funções.....	20
4.3.3	Impressão de termos e formulários	21
4.3.4	Envio de email	21
4.3.5	Identificação de Tipos de Registro e Arquivos Lógicos	22
4.3.6	Conversão de Dados	22
4.3.7	Migração de Dados dissociada de Projeto	22
4.3.8	Migração de Dados frequente/periódica.....	23
4.3.9	Consultas/Pesquisas com diversas opções de filtro	23
4.3.10	Gravação da Trilha de Auditoria.....	23
4.3.11	Consulta à Trilha de Auditoria	24
4.3.12	Dados da Trilha de Auditoria.....	24
4.4	Contagem Estimativa de Pontos de Função (CEPF)	24
4.5	Abordagem para Mudança de Requisitos.....	27
4.6	Fator de Criticidade.....	30
5	Relatório de Pontos de Função.....	30
6	Formação de base histórica	31

1 Introdução

Os métodos de medição permitem expressar quantitativamente qualidades de um objeto ou de um fenômeno. Quando se mede um objeto físico, as dimensões de interesse mais populares costumam ser comprimento, volume, área e peso. De maneira semelhante um software também possui várias dimensões. A *Análise de Pontos de Função* (APF) mede, especificamente, os requisitos funcionais do usuário e a essa dimensão é dado o nome de *Tamanho Funcional*.

Várias instituições públicas e privadas têm utilizado essa métrica nas estimativas e dimensionamento de tamanho funcional de projetos de software em virtude de diversos benefícios, como:

- ✓ Regras de contagem objetivas;
- ✓ Independência da solução tecnológica utilizada;
- ✓ Facilidade de estimativa nas fases iniciais do projeto.

É importante destacar que a Instrução Normativa nº 04 SLTI/MPOG recomenda o uso de métricas em contratos de projetos de software. Além disso, a Portaria nº 31 SLTI/MP, de 29 de novembro de 2010, recomenda o uso da métrica de Pontos de Função para órgãos integrantes do Sistema de Administração dos Recursos de Tecnologia da Informação - SISP, do Poder Executivo Federal, bem como a adoção do Roteiro de Métricas de Software do SISP, ou documento similar, na contratação de serviços de desenvolvimento e manutenção de soluções de software. Outros documentos de referência nesse contexto são os Acórdãos do Tribunal de Contas da União (TCU) que recomendam a utilização da métrica de Pontos de Função Não Ajustados em contratos de prestação de serviços de desenvolvimento e manutenção de sistemas.

Dando início ao processo de adequação às normas citadas, a Superintendência de Gestão Técnica da Informação (SGI) da ANEEL adotou como padrão para as regras de contagem de Pontos de Função o *Manual de Práticas de Contagem de Pontos de Função* (CPM 4.3.1) publicado pelo *International Function Point Users Group* (IFPUG).

É importante deixar claro que o CPM 4.3.1 é um documento que se destina a mensurar o tamanho funcional de projetos de software, não contemplando a geração de estimativas de prazo e custo desses e não tendo por objetivo principal suportar contratos de fábrica de software.

Assim, torna-se necessário criar roteiros complementares que abordem as questões não tratadas pelo CPM 4.3.1 do IFPUG, mas vivenciadas pela ANEEL.

2 Objetivo

O objetivo principal deste documento é apresentar um Roteiro de Métricas para todos os Projetos de Desenvolvimento e Manutenção de Sistemas gerenciados pela ANEEL, tendo como base as regras de contagem de Pontos de Função dos seguintes documentos:

- 1) Roteiro de Métricas do SISP versão 2.0;
- 2) Manual de Práticas de Contagem de Pontos de Função (CPM 4.3.1) do IFPUG;

- 3) Guia “Análise de Pontos de Função para Melhoria de Software” da *Netherlands Software Metrics Users Association* (NESMA);
- 4) Metodologia de Desenvolvimento de Sistemas da ANEEL (MDS-ANEEL).

Dessa forma, este roteiro contempla projetos para Desenvolvimento de Novas Aplicações, Projetos de Manutenção Evolutiva e Corretiva, Projetos para Carga/Conversão de Dados, bem como outras atividades não mensuráveis segundo o CPM 4.3.1.

Além da contagem detalhada de Pontos de Função, este roteiro apresenta um processo de estimativas com base nessa métrica, de forma a apoiar as estimativas de tamanho, custo, prazo e esforço dos projetos gerenciados pela ANEEL.

Por fim, o propósito é estimular o uso de métricas objetivas tanto em contratos de prestação de serviços de desenvolvimento/manutenção de sistemas como em projetos elaborados internamente e estabelecer uma terminologia comum com relação a conceitos, métodos e regras de contagem de Pontos de Função.

3 Métrica de Pontos de Função

A métrica PF mede o tamanho funcional de um projeto de software, observando as funcionalidades implementadas, considerando a visão do usuário. Ela é independente da metodologia e tecnologia utilizadas. A Análise de Pontos de Função (APF) é um método padrão para a medição de Projetos de Desenvolvimento/Manutenção de Sistemas, visando estabelecer uma medida de tamanho do software em Pontos de Função, com base na quantificação das funcionalidades solicitadas e entregues, sob o ponto de vista do usuário. Assim, a APF tem como objetivo medir o que o software faz, por meio de uma avaliação padronizada dos requisitos de negócio do sistema.

A Figura 01 ilustra o processo de contagem de Pontos de Função descrito ao longo deste roteiro.



Figura 01 – Procedimento de contagem de Pontos de Função.

3.1 Determinar o Escopo, as Fronteiras e identificar os Requisitos Funcionais do usuário

A contagem de Pontos de Função se inicia com a análise da documentação disponível do projeto em questão, visando à identificação dos requisitos funcionais.

O escopo da contagem define o conjunto dos requisitos funcionais do usuário que serão incluídos na contagem de Pontos de Função.

A fronteira é uma interface conceitual entre o software em análise e seus usuários.

De acordo com o seu propósito, a contagem de Pontos de Função pode ser de três tipos:

- ✓ Contagem de Pontos de Função de Projeto de Desenvolvimento;
- ✓ Contagem de Pontos de Função de Projeto de Melhoria;
- ✓ Contagem de Pontos de Função da Aplicação (*Baseline*).

3.2 Função de Dados

Uma Função de Dados representa a funcionalidade fornecida ao usuário para atender suas necessidades internas e externas de armazenamento de dados. Uma Função de Dados pode ser um Arquivo Lógico Interno ou um Arquivo de Interface Externa.

- ✓ Um *Arquivo Lógico Interno* (ALI) é um grupo de dados ou informações de controle, reconhecido pelo usuário e mantido dentro da fronteira da aplicação sendo medida. A principal intenção de um ALI é armazenar dados mantidos por um ou mais processos elementares da aplicação sendo medida.
- ✓ Um *Arquivo de Interface Externa* (AIE) 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 um ALI em alguma outra aplicação.

3.3 Função de Transação

Uma Função de Transação é um processo elementar que fornece funcionalidade ao usuário para processamento de dados. Uma Função de Transação pode ser uma Entrada Externa, Saída Externa ou Consulta Externa.

- ✓ Uma *Entrada Externa* (EE) é um processo elementar que processa dados ou informações de controle recebidos de fora da fronteira da aplicação. A intenção primária de uma EE é manter um ou mais ALI e/ou alterar o comportamento do sistema.
- ✓ Uma *Saída Externa* (SE) é um processo elementar que envia dados ou informações de controle para fora da fronteira da aplicação e inclui processamento adicional além daquele existente em uma consulta externa. A intenção primária de uma Saída Externa é apresentar dados ao usuário através de lógica de processamento

que não seja apenas recuperação de dados ou informações de controle. A lógica de processamento deve conter ao menos uma fórmula matemática ou cálculo, e/ou criar dados, e/ou manter um ou mais ALIs, e/ou alterar o comportamento do sistema.

✓ Uma *Consulta Externa* (CE) é um processo elementar que envia dados ou informações de controle para fora da fronteira da aplicação. A intenção primária de uma Consulta Externa é apresentar dados ao usuário através de recuperação de dados ou informação de controle. A lógica de processamento não contém fórmula matemática, nem cálculo, nem cria dados derivados. Nenhum ALI é mantido durante o processamento, nem o comportamento do sistema é alterado.

3.4 Calcular Tamanho Funcional

O Tamanho Funcional representa o tamanho do software obtido pela quantificação dos requisitos funcionais do usuário.

A funcionalidade específica da aplicação do usuário é avaliada em termos "*do que é entregue*" e não "*como é entregue*". Apenas componentes solicitados e definidos pelo usuário são contados.

O Tamanho Funcional é obtido através da medição das Funções de Dados e de Transação.

Após a identificação do tipo funcional, deve-se avaliar a sua complexidade (Baixa, Média ou Alta) e a sua contribuição para a contagem, observando as regras descritas no CPM 4.3.1.

A Tabela 01 apresenta a contribuição dos tipos funcionais na contagem de Pontos de Função.

Tipo Funcional	Complexidade		
	Baixa	Média	Alta
Arquivo Lógico Interno (ALI)	7 PF	10 PF	15 PF
Arquivo de Interface Externa (AIE)	5 PF	7 PF	10 PF
Entrada Externa (EE)	3 PF	4 PF	6 PF
Saída Externa (SE)	4 PF	5 PF	7 PF
Consulta Externa (CE)	3 PF	4 PF	6 PF

Tabela 01 – Contribuição funcional dos tipos funcionais.

3.5 Fator de Ajuste

A ANEEL não utiliza a avaliação do Fator de Ajuste em suas contagens, sendo esse fixado no valor 1,00. Ou seja, as características gerais do sistema, apresentadas no CPM 4.3.1 do IFPUG, não alteram o tamanho dos Pontos de Função Não Ajustados a fim de evitar quantificações subjetivas, conforme orientação do TCU (Acórdão nº 1910/2007).

4 Métrica de Pontos de Função para a ANEEL

Este capítulo tem como propósito descrever os diversos tipos de projetos de software e definir métricas baseadas nas regras de contagem de Pontos de Função do CPM 4.3.1 para o seu dimensionamento.

Seguem abaixo as definições dos termos técnicos da Análise de Pontos de Função utilizados nas fórmulas de dimensionamento de projetos de software definidas neste roteiro.

- ✓ *PF_INCLUÍDO* = Pontos de Função associados às novas funcionalidades que farão parte da aplicação após um projeto de desenvolvimento ou de manutenção;
- ✓ *PF_ALTERADO* = Pontos de Função associados às funcionalidades existentes na aplicação que serão alteradas no projeto de manutenção;
- ✓ *PF_EXCLUÍDO* = Pontos de Função associados às funcionalidades existentes na aplicação que serão excluídas no projeto de manutenção;
- ✓ *PF_CONVERSÃO* = Pontos de Função associados às funcionalidades de conversão de dados dos Projetos de Desenvolvimento ou de Manutenção. Exemplos de Funções de Conversão incluem: migração ou carga inicial de dados para popular as novas tabelas criadas (Entradas Externas) e relatórios associados à migração de dados, caso requisitado pelo usuário (Saídas Externas ou Consultas Externas). Os dados carregados em um processo de migração não devem ser contados como Arquivos de Interface Externa.
- ✓ *FI* = Fator de Impacto, que pode variar de 50% a 90%, conforme abaixo:
 - a) *FI* = 50% para funcionalidade de sistema desenvolvida ou mantida por meio de um projeto de melhoria pela empresa contratada;
 - b) *FI* = 75% para funcionalidade de sistema não desenvolvida ou mantida por meio de um projeto de melhoria pela empresa contratada e sem necessidade de redocumentação da funcionalidade;
 - c) *FI* = 90% para funcionalidade de sistema não desenvolvida ou mantida por meio de um projeto de melhoria pela empresa contratada e com necessidade de redocumentação da funcionalidade. A contratada deve redocumentar a funcionalidade mantida, gerando a documentação completa da mesma, aderente à Metodologia de Desenvolvimento de Sistemas da ANEEL (MDS-ANEEL). Se houver uma nova demanda de projeto de melhoria na funcionalidade em questão, será considerado que a contratada desenvolveu a funcionalidade. Dessa forma, o percentual de 90% apenas será considerado na primeira demanda de projeto de melhoria em cada funcionalidade.

A critério da ANEEL, *PF_CONVERSÃO* poderá ser suprimido das fórmulas de contagem de Pontos de Função de Projetos de Desenvolvimento e de Melhoria nos casos específicos em que for caracterizado um esforço relativamente maior dessa atividade. Por exemplo, os projetos que envolvam a migração de dados de banco de dados hierárquico para banco de dados relacionais, e o tratamento das funções complexas de migração de dados. Nesses casos, os serviços serão tratados como projetos separados de migração de dados.

Este roteiro define o um Fator de Redocumentação de 15% para Projetos de Manutenção (Evolutiva, Corretiva e Adaptativa). Projetos específicos de documentação devem seguir o que consta na seção 4.2.3.9.

Os percentuais de *FI* acima correspondem à contratação de todas as fases do Processo de Desenvolvimento/Manutenção de Software. Caso alguma fase não seja contratada, deve-se aplicar ao *FI* um redutor que corresponde ao percentual da fase não contratada, conforme percentuais definidos na Tabela 02 e de acordo com a Metodologia de Desenvolvimento de Sistemas da ANEEL.

Fase do Projeto de Desenvolvimento/Manutenção de Software	Percentual de esforço (%)
Iniciação	5%
Elaboração	20%
Construção	65%
Transição	10%

Tabela 02 – Distribuição de esforço por fase do projeto.

Funções Alteradas

Uma Função de Dados (Arquivo Lógico Interno ou Arquivo de Interface Externa) é considerada alterada quando houver inclusão ou exclusão de Tipo de Dados ou quando algum tipo de dado sofrer mudança de tamanho (número de posições) ou tipo de campo (por exemplo: mudança de numérico ou alfanumérico). No último caso, a mudança deve decorrer de alteração de regra de negócio.

Uma Função de Transação (Entrada Externa, Consulta Externa ou Saída Externa) é considerada alterada, quando a alteração contemplar:

- ✓ Mudança de Tipos de Dados;
- ✓ Mudança de Arquivos Referenciados;
- ✓ Mudança de Lógica de Processamento.

Este roteiro considera Função Transacional alterada qualquer mudança em funcionalidades da aplicação, devido às mudanças de regras de negócio. Por exemplo, uma funcionalidade de cadastro implicava na inclusão de um telefone do gerente. Devido a mudanças no processo de negócio, a funcionalidade deve sofrer uma manutenção para cadastrar dois telefones do gerente. O presente roteiro considera essa função como uma Entrada Externa alterada, ou seja, *PF_ALTERADO* em um Projeto de Manutenção Evolutiva (Projeto de Melhoria), mesmo que não exista mudança de lógica, mudança de Tipos de Dados e mudança de Arquivos Referenciados.

Serão tratadas como manutenções adaptativas as manutenções que implicarem apenas em mudanças em requisitos não funcionais. Se uma mesma funcionalidade tiver mudanças em requisitos funcionais e não funcionais, essa deve ser contada apenas uma vez, como função alterada em um Projeto de Manutenção Evolutiva (Projeto de Melhoria).

4.1 Contagem de Projeto de Desenvolvimento

É considerada uma nova aplicação aquele projeto cujos requisitos funcionais não são atendidos por outro sistema já existente na ANEEL ou quando não caracterizam um Projeto de Manutenção, mesmo que Evolutiva, de sistema legado. Além disso, o desenvolvimento de um novo sistema, para substituir um já existente, também se enquadra nessa categoria.

Para contagens de Projetos de Desenvolvimento, o cálculo do Tamanho Funcional é definido no CPM 4.3.1 de acordo com a fórmula abaixo:

$$PF = PF_INCLUÍDO + PF_CONVERSÃO$$

4.2 Contagem de Projeto de Manutenção

Uma demanda de manutenção é caracterizada por modificações em requisitos funcionais e não funcionais, conforme descrito nessa seção. Existem quatro tipos de Projetos de Manutenção, a saber:

- ✓ *Projeto de Manutenção Evolutiva*: necessidade de melhoria ou evolução dos requisitos funcionais de um sistema que esteja em produção;
- ✓ *Projeto de Manutenção Corretiva*: necessidade de correção de erros em requisitos funcionais de um sistema em produção;
- ✓ *Projeto de Manutenção Adaptativa*: contempla alterações em requisitos não funcionais. Fornece melhorias necessárias para acomodar mudanças no ambiente no qual o software deve operar. Essas mudanças são aquelas que devem ser feitas para que acompanhem as mudanças no ambiente.

O CPM 4.3.1 não contempla todos os tipos de Projetos de Manutenção, apenas os que alteram requisitos funcionais. Este capítulo tem como propósito descrever os diversos Projetos de Manutenção e definir métricas baseadas nas regras de contagem de Pontos de Função do CPM 4.3.1 para seu dimensionamento. Com isso, visa estender essas regras para considerar a prática dos projetos de software da ANEEL.

4.2.1 Manutenção Evolutiva

O Projeto de Melhoria (*Enhancement*), aqui denominado de Manutenção Evolutiva, está associado às mudanças em requisitos funcionais da aplicação, ou seja, à inclusão de novas funcionalidades, alteração ou exclusão de funcionalidades em aplicações implantadas.

Segundo o padrão IEEE Std 1219, esta manutenção seria um tipo de Manutenção Adaptativa, definida como: modificação de um produto de software concluído após a entrega para mantê-lo funcionando adequadamente em um ambiente com mudanças. O Projeto de Melhoria é considerado um tipo de projeto de Manutenção Adaptativa com mudanças em requisitos funcionais da aplicação, ou seja, com funcionalidades incluídas, alteradas ou excluídas na aplicação, segundo o CPM 4.3.1.

Este documento separa a Manutenção Evolutiva (quando as mudanças são associadas aos requisitos funcionais) da Manutenção Adaptativa (quando as mudanças estão associadas aos requisitos não funcionais da aplicação).

Um projeto de Manutenção Evolutiva consiste em demandas de criação, exclusão e alteração de novas funcionalidades (grupos de dados ou processos elementares) em aplicações implantadas.

Segue a fórmula de contagem de Pontos de Função para projetos de Manutenção Evolutiva.

$$PF = PF_INCLUÍDO + (FI \times PF_ALTERADO) + (0,30 \times PF_EXCLUÍDO) + PF_CONVERSÃO$$

4.2.2 Manutenção Corretiva

Mesmo com a execução de atividades de Garantia da Qualidade, é possível identificar defeitos na aplicação entregue. Assim, a Manutenção Corretiva altera o software para correção de defeitos. Encontram-se nesta categoria as demandas de correção de erros (*bugs*) em funcionalidades de sistemas em produção.

A estimativa e dimensionamento de tamanho de Projetos de Manutenção Corretiva em Pontos de Função devem levar em consideração a documentação do sistema, os artefatos a serem mantidos e a empresa que desenvolveu, conforme abaixo:

a) Aplicação fora da garantia e correção feita pela mesma empresa que desenvolveu a funcionalidade;

Nestes casos, a aferição do tamanho em Pontos de Função da funcionalidade ou das funcionalidades corrigidas considera 50% do *PF_ALTERADO*.

$$PF = PF_ALTERADO \times 0,50$$

a) Aplicação fora da garantia e correção feita por empresa diferente daquela que desenvolveu a funcionalidade;

Nestes casos, a aferição do tamanho em Pontos de Função da funcionalidade ou das funcionalidades corrigidas considera 75% do *PF_ALTERADO*.

$$PF = PF_ALTERADO \times 0,75$$

As demandas de manutenção corretiva não contemplam atualização de documentação da funcionalidade corrigida, exceto quando o erro for causado por documentação dúbia ou imprecisa dessa. Nesse último caso, a manutenção corretiva contemplará os ajustes na documentação, mesmo fora da garantia.

Caso seja demandada a redocumentação da funcionalidade corrigida, porque a documentação não existe ou está desatualizada, deve-se adicionar ao FI um Fator de Redocumentação de 15%.

4.2.3 Manutenção Adaptativa

Este roteiro considera Projeto de Manutenção Adaptativa as atualizações efetuadas em requisitos não funcionais de sistemas em produção.

A Análise de Pontos de Função não contempla diretamente os requisitos não funcionais de um projeto. Entretanto, este roteiro descreverá algumas categorias de Manutenção Adaptativa cujo dimensionamento será efetuado usando essa métrica.

4.2.3.1 Mudança de Plataforma

São considerados nesta categoria, projetos que precisam ser migrados para outra plataforma. Por exemplo, um sistema legado em COBOL que necessita ser reimplementado em JAVA; o banco de dados de um sistema legado que precisa ser migrado para o DB2.

As próximas subseções apresentam os tipos de projetos de mudança de plataforma. Os projetos de mudança de plataforma que se enquadram em mais de uma subseção, devem ser contados apenas uma vez, considerando o tipo de projeto com maior contagem de pontos de função.

4.2.3.1.1 Mudança de Plataforma - Linguagem de Programação

Nesta categoria encontram-se as demandas de reconstrução de sistemas em outra linguagem de programação. Como esses projetos legados, frequentemente, encontram-se sem documentação, devem ser considerados como novos Projetos de Desenvolvimento. Assim, será utilizada a fórmula de Projetos de Desenvolvimento do CPM 4.3.1.

Caso não exista mudança nas Funções de Dados, ou seja, o Banco de Dados da aplicação seja mantido, as Funções de Dados não devem ser contadas.

Outro ponto a ser observado são as fases contratadas. Caso o projeto já possua documentação de requisitos, a fase de requisitos não será contratada. Deve-se considerar apenas os percentuais das fases contratadas.

$$PF = PF_INCLUÍDO + PF_CONVERSÃO$$

4.2.3.1.2 Mudança de Plataforma - Banco de Dados

Nesta categoria encontram-se as demandas de reconstrução de sistemas para utilizar outro Sistema Gerenciador de Banco de Dados.

Caso não exista mudança nas Funções de Dados, ou seja, o Banco de Dados da aplicação seja mantido, então as Funções de Dados não devem ser contadas.

Em caso de mudança de banco hierárquico para relacional, em sistemas sem documentação, devido às mudanças envolvidas, deve-se considerar como um novo Projeto de Desenvolvimento, ou seja, as Funções de Dados e Funções Transacionais devem ser contadas. Assim, será utilizada a fórmula de projeto de desenvolvimento do CPM 4.3.1, conforme fórmula abaixo:

$$PF = PF_INCLUÍDO + PF_CONVERSÃO$$

Caso o projeto já possua documentação de requisitos, então a fase de requisitos não deve ser contratada. Deve-se considerar apenas os percentuais das fases contratadas.

Caso a demanda de reconstrução seja de um Sistema Gerenciador de Banco de Dados relacional para outro relacional, deve ser utilizada a seguinte fórmula:

$$PF = (PF_ALTERADO \times 0,30) + PF_CONVERSÃO$$

O *PF_ALTERADO* deve considerar apenas as funcionalidades impactadas.

4.2.3.2 Atualização de Plataforma - Browser/Linguagens de Programação

São consideradas nesta categoria as demandas para que uma aplicação existente ou parte dela execute em versões mais atuais de browsers (ex: versão atual do Internet Explorer, Mozilla, Firefox, etc) ou de linguagens de programação (ex: versão mais atual do JAVA ou do Banco de Dados).

Também são consideradas nesta categoria aplicações Web desenvolvidas para executar em Internet Explorer que precisam executar também em browser em software livre.

Nesses casos, a aferição do tamanho em Pontos de Função, da aplicação ou da parte da aplicação que sofreu impacto, considera-se 30% dos PF Não Ajustados, seguindo a fórmula de Projetos de Desenvolvimento do CPM 4.3.1.

$$PF = PF_NÃO_AJUSTADO \times 0,30$$

Um ponto importante a ser observado é a classificação, em alguns casos, dessas demandas como *Componente Interno Reusável*.

4.2.3.3 Atualização de Plataforma – Banco de Dados

Nesta categoria encontram-se as demandas de atualização de versão do Sistema Gerenciador de Banco de Dados. As Funções de Dados não devem ser contadas. Estas demandas devem ser dimensionadas de acordo com a fórmula abaixo.

$$PF = PF_ALTERADO \times 0,30$$

4.2.3.4 Manutenção em Interface

A Manutenção em Interface é associada às demandas de alterações em elementos de interface gráfica como: fonte de letra, cores de telas, logotipos, mudança de botões na tela, mudança de posição de campos, mudanças de texto em mensagens de erro, validação, aviso, alerta ou conclusão de processamento, etc.

Nesses casos, a aferição do tamanho em Pontos de Função das funcionalidades alteradas será efetuada com a aplicação de um fator, de forma que seja considerado 20% da contagem de uma Função de Transação de Baixa Complexidade (3 PF), independentemente da complexidade da funcionalidade alterada. Caso seja utilizada uma mesma tela para duas ou mais funcionalidades, deve ser contada apenas uma Função Transacional de Baixa Complexidade.

Está contemplada a atualização da documentação das funcionalidades da aplicação impactadas pela manutenção nas demandas desta categoria. Caso não exista documentação para as funcionalidades alteradas, não

será contemplada a redocumentação das funcionalidades da aplicação impactadas pela manutenção nas demandas desta categoria.

$$PF = 0,60 PF \times QUANTIDADE_DE_PROCESSOS_ELEMENTARES_ALTERADOS$$

As demandas específicas de desenvolvimento ou atualização de help estático de funcionalidades serão enquadradas nesta seção de forma que seja considerado 10% da contagem de uma Função de Transação de Baixa Complexidade (3 PF). Em caso de requisitos de usuário para o desenvolvimento de funcionalidades de manutenção de help, deve-se contar a função de dados de help e as funcionalidades de manutenção de help (por exemplo: incluir help de tela, consultar help de campo) de acordo com o CPM 4.3.1.

$$PF = 0,30 PF \times QUANTIDADE_DE_PROCESSOS_ELEMENTARES_ALTERADOS$$

4.2.3.5 Adaptação de Funcionalidades a Requisitos Não Funcionais

São consideradas nesta categoria as demandas de Manutenção Adaptativa associadas a solicitações que envolvem aspectos não funcionais, sem alteração em requisitos funcionais. Por exemplo:

- ✓ Aumentar a quantidade de linhas por página em um relatório;
- ✓ Colocar paginação em um relatório;
- ✓ Permitir exclusões múltiplas em uma funcionalidade que antes só possibilitava a exclusão de um item;
- ✓ Adaptação de uma funcionalidade para possibilitar a chamada por um *Webservice* ou para outro tipo de integração com outros sistemas;
- ✓ Replicação de funcionalidade: chamar uma consulta existente em outra tela da aplicação;
- ✓ Replicação de base de dados ou criação de base temporária para resolver problemas de desempenho ou segurança;
- ✓ Alteração na aplicação para adaptação às alterações realizadas na interface com rotinas de integração com outros softwares, por exemplo, alteração em sub-rotinas chamadas por este software.

Nesta categoria foram observadas demandas dos seguintes tipos:

a) **Funcionalidade de sistema desenvolvida ou mantida por meio de um projeto de melhoria pela empresa contratada**

Nestes casos, a aferição do tamanho em Pontos de Função da funcionalidade ou das funcionalidades que foram afetadas deve considerar 50% do *PF_ALTERADO*, seguindo as definições do CPM 4.3.1, apresentadas no capítulo 4 deste roteiro. Será contemplada a documentação das funcionalidades nas demandas desta categoria.

$$PF = PF_ALTERADO \times 0,50$$

b) Funcionalidade de sistema não desenvolvida ou mantida por meio de um projeto de melhoria pela empresa contratada

Nestes casos, a aferição do tamanho em Pontos de Função da funcionalidade ou das funcionalidades que sofreram impacto deve considerar 75% do *PF_ALTERADO*, seguindo as definições do CPM 4.3.1, apresentados no capítulo 4 deste roteiro. Será contemplada a documentação das funcionalidades nas demandas desta categoria.

$$PF = PF_ALTERADO \times 0,75$$

4.2.3.6 Projetos de Migração de Dados (Conversão de dados)

Os Projetos de Migração de Dados devem ser contados como um novo Projeto de Desenvolvimento de um sistema, contemplando minimamente: os ALIs mantidos pela migração, Entradas Externas - considerando as cargas de dados nos ALI e, caso seja solicitado pelo usuário, relatórios gerenciais das cargas (que serão contados como Saídas Externas). Todas as contagens de PF devem ser realizadas com base nas funcionalidades requisitadas e recebidas pelo usuário.

Os projetos de migração de dados devem ser contados como um novo Projeto de Desenvolvimento de um sistema, seguindo as orientações citadas, de acordo com a fórmula abaixo:

$$PF = PF_INCLUÍDO$$

4.2.3.7 Apuração Especial

São funcionalidades executadas apenas uma vez para: corrigir problemas de dados incorretos na base de dados das aplicações ou atualizar dados nessas bases, gerar um relatório específico ou arquivo para o usuário por meio de recuperação de informações nas bases da aplicação.

Caso a apuração seja de correção de dados devido a erros de funcionalidades de aplicações desenvolvidas pela contratada deve-se observar as cláusulas contratuais com relação a garantias e prazos de correção.

A contratada armazenará todos os scripts gerados e os fornecerá à ANEEL, sem ônus, sempre que solicitada.

4.2.3.7.1 Apuração Especial – Base de Dados

Este tipo de apuração especial é um projeto que inclui a geração de procedimentos para atualização da base de dados. Deve-se destacar que estas funções são executadas apenas uma vez, não fazendo parte da aplicação, visando a correção de dados incorretos na base de dados da aplicação ou atualização em função de modificação da estrutura de dados, por exemplo inclusão de valor “sim” ou “não” no campo “indicador de matriz” referente ao CNPJ.

Normalmente, nesse tipo de atualização são afetados múltiplos registros. Nesses casos, considera-se a

contagem de pontos de função das funcionalidades desenvolvidas. Essas funcionalidades são classificadas como Entradas Externas. Nesse caso, como artefato de homologação da demanda, deve ser gerado um relatório para validação do usuário.

É importante ressaltar que as funções de dados associadas aos dados atualizados não devem ser contadas, considerando que não há mudanças nas estruturas dos Arquivos Lógicos Internos.

Foram identificados três tipos de Apuração Especial - Base de Dados, cujas fórmulas de cálculo são apresentadas a seguir:

a) Atualização de Dados sem Consulta Prévia

$$PF = PF_INCLUÍDO$$

b) Consulta Prévia sem Atualização

A consulta prévia não é definida pela empresa contratada, obrigatoriamente essa deve ser solicitada pela ANEEL para a avaliação da viabilidade de implementar a Apuração Especial - Base de Dados. Esta consulta prévia, classificada como Consulta Externa ou Saída Externa deve ser dimensionada considerando-se o tamanho da funcionalidade em questão, conforme a fórmula abaixo:

$$PF = PF_INCLUÍDO$$

c) Consulta Prévia sem Atualização

Caso a Apuração Especial - Base de Dados seja solicitada após uma demanda de consulta prévia, deve-se aplicar um fator de 60% na fórmula de contagem da Apuração Especial - Base de Dados, seguindo a fórmula abaixo.

$$PF = PF_INCLUÍDO \times 0,60$$

4.2.3.7.2 Apuração Especial – Geração de Relatórios

Este tipo de apuração especial é um projeto que inclui a geração de relatórios em uma ou mais mídias para o usuário. Em alguns casos, são solicitadas extrações de dados e envio desses para outros sistemas. Caso, nesse envio de dados, sejam requisitadas atualizações no sistema de origem, então essas funções transacionais são Saídas Externas, devido à atualização do Arquivo Lógico Interno.

Deve-se destacar que essas funções são executadas apenas uma vez, não fazendo parte da aplicação. Nestes casos, considera-se contagem de pontos de função das funcionalidades desenvolvidas. Frequentemente, essas

funcionalidades são classificadas como Saídas Externas. Também podem ser classificadas como Consultas Externas, caso não possuam cálculos ou criação de dados derivados.

É importante ressaltar que as funções de dados associadas aos dados atualizados não devem ser contadas, considerando que não há mudanças nas estruturas dos Arquivos Lógicos.

$$PF = PF_INCLUÍDO$$

4.2.3.7.3 Apuração Especial – Reexecução

A ANEEL solicita formalmente à contratada o armazenamento de todos os scripts executados. Se for solicitada a reexecução de uma apuração especial, esta deve ser dimensionada com a aplicação de um fator redutor de 10% na contagem de pontos de função da apuração especial em questão, da seguinte maneira:

$$PF = PF_NÃO_AJUSTADO \times 0,10$$

4.2.3.8 Atualização de Dados

Em alguns casos, as demandas de correção de problemas em base de dados estão associadas a atualizações manuais (de forma interativa), diretamente no banco de dados em um único registro, e que não envolvem cálculos ou procedimentos complexos.

São exemplos desse tipo de demanda, a atualização do valor de um campo de uma tabela cadastrado erroneamente ou a exclusão de um registro de uma tabela.

Nesses casos, a aferição do tamanho em Pontos de Função deve considerar 10% do PF de uma Entrada Externa e os Tipos de Dados da Entrada Externa são todos os TD considerados na funcionalidade – campos atualizados e campos utilizados para a seleção do registro.

$$PF = PF_INCLUÍDO \times 0,10$$

4.2.3.9 Documentação de Sistemas Legados

Nesta seção são tratadas demandas de documentação ou atualização de documentação de Sistemas Legados. O desenvolvedor deve realizar uma Engenharia Reversa da aplicação para gerar a documentação.

Conforme Metodologia de Desenvolvimento de Sistemas da ANEEL (MDS-ANEEL), a documentação mínima exigida para um projeto é composta de:

- ✓ Especificação de Requisitos, contendo os Casos de Uso e Regras de Negócios das alterações realizadas;
- ✓ Documento de Arquitetura;
- ✓ Modelo de Dados;

- ✓ Manual do Usuário.

Para este tipo de projeto, caso a demanda seja apenas a Especificação de Requisitos, serão considerados 20% dos Pontos de Função Não Ajustados da aplicação em questão, conforme a fórmula abaixo:

$$PF = PF_NÃO_AJUSTADO \times 0,20$$

Caso a demanda seja a geração de todos os artefatos de documentação descritos anteriormente, será considerado um percentual de 40%, conforme fórmula abaixo:

$$PF = PF_NÃO_AJUSTADO \times 0,40$$

Importante ressaltar que todos os artefatos descritos nesta sessão deverão seguir o padrão definido pela Metodologia de Desenvolvimento de Sistemas da ANEEL.

4.2.3.10 Pontos de Função de Teste

Muitas vezes em projetos de manutenção o conjunto de Funções Transacionais a serem testadas é maior do que a quantidade de funções a serem implementadas, isto é, além das funcionalidades que são afetadas diretamente pelo projeto de manutenção, outras precisam ser testadas. O tamanho das funções a serem testadas deve ser aferido em Pontos de Função de Teste (PFT). Não considerar as funcionalidades incluídas, alteradas ou excluídas do projeto de manutenção na contagem de Pontos de Função de Teste.

A contagem de PFT deve considerar o tamanho em Pontos de Função de todas as Funções Transacionais envolvidas no teste.

PFT = Somatório dos tamanhos das Funções Transacionais testadas;

A conversão do PFT em Ponto de Função deve ser feita de acordo com a fórmula abaixo:

$$PF = PFT \times 0,15$$

É importante ressaltar que os testes considerados no PFT devem ser documentados (casos de testes, plano de testes, etc.) conforme metodologia adotada pela ANEEL. Essas funções farão parte do escopo do projeto de manutenção.

É importante ressaltar que no caso de uma função ser testada várias vezes, com cenários diferentes, a função só pode ser contada uma vez.

4.2.3.11 Componente Interno Reusável

Em alguns casos são demandadas manutenções em componentes específicos de uma aplicação e estes são reusados por várias funcionalidades. Por exemplo, uma mudança em uma rotina de validação de um CPF usada em várias funcionalidades de cadastro.

Este roteiro define que o componente, o qual deverá ser testado, seja considerado como um processo elementar independente e contado como uma funcionalidade. Além disso, as funcionalidades da aplicação que

necessitem de teste devem ser requisitadas pela ANEEL e dimensionadas por meio da métrica Pontos de Função de Teste proposta na seção 4.2.3.10.

PF = PF_NÃO_AJUSTADO

Seguem alguns exemplos de manutenção de componentes:

- ✓ Alteração de valores de elementos internos de configuração que afetem o comportamento ou a apresentação do sistema de forma geral, tais como páginas de estilos (arquivos CSS de sistemas Web), arquivos com mensagens de erro, arquivos de configuração de sistema, arquivos de internacionalização;
- ✓ Mudança em tópico de um menu de um sistema em PHP que aparece em todas as telas da aplicação. A contagem pode ser realizada considerando o componente “Apresentar Menu”.

4.2.4 Contagem de Pontos de Função com Múltiplas Mídias

Esta seção tem como propósito apresentar as diretrizes de contagem de Pontos de Função em relação ao tema Múltiplas Mídias.

Considerando a contagem de Pontos de Função de funcionalidades entregues em mais de uma mídia, a aplicação das regras definidas no CPM 4.3.1 tem levado a duas abordagens alternativas, *single instance* e *multiple instance*.

A abordagem *single instance* considera que a entrega de uma Função Transacional em múltiplas mídias não deve ser utilizada na identificação da unicidade da função (várias mídias são consideradas como uma única funcionalidade, sendo contadas apenas uma vez).

A abordagem *multiple instance* leva em consideração que a mídia utilizada na entrega da funcionalidade é uma característica de identificação da unicidade da função (cada mídia é contada como uma funcionalidade entregue, portanto várias mídias implicam em várias contagens). Assim, funcionalidades únicas são reconhecidas no contexto da mídia na qual elas são requisitadas para operar.

O IFPUG reconhece ambas as abordagens para a aplicação das regras definidas no CPM 4.3.1. A determinação da contagem de PF seguindo a abordagem *single instance* ou *multiple instance* depende da avaliação da Equipe de Métricas da ANEEL.

As estimativas e contagens de PF abordadas neste roteiro serão baseadas em *single instance* exceto os casos aderentes aos cenários a seguir, que poderão utilizar abordagem *multiple instance*:

a) Cenário 01 - Mesmos dados de entrada batch e on-line

Uma informação pode ser carregada na aplicação por meio de dois métodos: arquivo batch e entrada on-line.

O processamento do arquivo batch executa validações durante o processamento, da mesma forma que o processamento da entrada on-line também executa validações das informações. Nesse caso, será a utilizada a abordagem *multiple instance*, que conta duas funcionalidades: a entrada de dados batch e a entrada de dados on-line.

Geralmente, a lógica de processamento utilizada nas validações em modo batch é diferente da lógica de processamento das validações nas entradas de dados on-line.

b) Cenário 02 - Múltiplos canais de entrega da mesma funcionalidade

Uma funcionalidade deve ser disponibilizada em múltiplos canais, por exemplo: consulta de dados em página Web e consulta de dados no telefone celular. Neste caso, será utilizada a abordagem *multiple instance*, que conta duas funcionalidades: consulta de dados na Web e consulta de dados via celular.

Considera-se que a funcionalidade é desenvolvida duas vezes, uma para cada canal de saída. Algumas vezes, são até projetos de desenvolvimento distintos, um projeto relativo ao sistema Web e outro para o sistema via celular. Lembrando que caso o projeto seja claro o suficiente para dizer que o desenvolvimento é o mesmo, será utilizada a abordagem *single instance*.

4.3 Cenários

Este capítulo aborda alguns exemplos de situações práticas que podem gerar mensurações divergentes e como deverão ser interpretadas.

4.3.1 Campos não editáveis

Para telas de pesquisa, cadastro e alteração que apresentem automaticamente dados preenchidos em campos não editáveis, é importante avaliar a sua necessidade funcional.

- a) Campo preenchido automaticamente para restringir valores de entrada do usuário, de acordo com o perfil do usuário autenticado: os campos de negócio que atravessam a fronteira da aplicação e são utilizados como parâmetros para a funcionalidade de filtro devem ser contados como Tipos de Dados da funcionalidade em questão. Vale ressaltar que o fato destes campos serem carregados automaticamente não é suficiente para identificação de um segundo processo elementar para os usuários com perfil de gestor, uma vez que trata-se de uma facilidade na restrição do escopo de pesquisa.
- b) Campos preenchidos automaticamente em uma consulta implícita para alteração de dados: informações que não sejam essenciais para a alteração em si, porém são carregadas anteriormente por uma consulta implícita não devem ser contadas como tipos de dado da transação de alteração. Uma dica é separar a funcionalidade de alteração.

4.3.2 Subdivisão de uma funcionalidade em n funções

Para a correta quebra de uma funcionalidade em diversos processos elementares, é importante avaliá-las a partir de uma perspectiva de negócio, verificando quais funcionalidades são completas e reconhecidas pelos usuários do negócio.

- a) Mais de um processo elementar identificado para a funcionalidade: são contabilizadas n funções quando para o usuário estas são atividades distintas que refletem em situações diferentes para o negócio da aplicação.
- b) Um único processo elementar identificado para a funcionalidade: é necessário avaliar neste caso se cada uma das telas constitui um processo elementar, atendendo principalmente ao requisito de ser autocontido. Neste caso, a primeira coisa a se fazer é a de tentar identificar que razão levou à divisão da funcionalidade em diversas telas. É comum que formulários de cadastro sejam quebrados em etapas com objetivo de tornar a atividade de cadastro mais intuitiva e organizada, ou seja, apenas para atender a requisitos não funcionais. Assim sendo, apenas um único Processo Elementar pode ser identificado. Caso contrário, se a funcionalidade tiver sido subdividida em diversas telas para atender a uma necessidade do negócio, será caracterizado mais de um Processo Elementar. Duas dicas para auxiliar o analista de métricas em situações semelhantes são:
 - ✓ Verificar se, caso a funcionalidade não fosse fragmentada e houvesse uma única tela, haveria algum impacto no negócio da aplicação ou se a funcionalidade se tornaria apenas menos usual.
 - ✓ Verificar se há usuário de áreas de negócio distintas responsáveis por preencher telas específicas da funcionalidade, não tendo competência (mesmo que munidos de todas as informações necessárias) para o preenchimento completo do formulário.

4.3.3 Impressão de termos e formulários

Para correta contagem dos tipos de dados em funções responsáveis pela emissão de termos e formulários, é importante verificar se estes possuem campos recuperados de Arquivos Lógicos Internos ou Arquivos de Interface Externa (Arquivos referenciados da função) ou até mesmo gerados dinamicamente.

Formulário composto por dados resgatados de vários Arquivos Lógicos: deve-se identificar em quais arquivos lógicos os campos resgatados são armazenados para que estes sejam identificados como Arquivos Referenciados da função.

4.3.4 Envio de email

Muitas funções possuem embutidos em seu processamento, o envio de emails de notificação ao final.

- a) Envio de email como consequência de um processo: se o envio de email é sempre acionado após a efetivação de tal transação, será considerado parte da respectiva funcionalidade e não um processo elementar independente. A atividade que é significativa para o usuário é o conjunto de ações referentes à efetivação e ao envio do email, e não ambas isoladamente. Dessa forma, tanto os Arquivos Referenciados como os Tipos de Dado utilizados pela ação de envio de email devem ser contabilizados na transação responsável pelo seu disparo.

- b) Envio apenas do email, independente de outras transações: partindo do princípio que tal funcionalidade tenha sido criada devido a solicitações funcionais do negócio, a ação de envio de email, independente de quaisquer outras ações do sistema é consistente, significativa para o usuário e completa em si? Nesse caso deve ser classificada como um novo processo elementar.

4.3.5 Identificação de Tipos de Registro e Arquivos Lógicos

Para a correta contagem dos Tipos de Registro e Arquivos Lógicos, é importante verificar as entidades do negócio e suas respectivas dependências funcionais.

É necessária uma atenção especial com relação à distinção dos conceitos de Dependência Funcional e Obrigatoriedade de Relacionamento. O último restringe associações dentro de um modelo de dados, ou seja, uma entidade deve obrigatoriamente apontar para um registro em outra entidade, porém isto não é suficiente para declará-las como dependentes funcionais entre si. Uma entidade pode ter significado para o negócio independente de qualquer outra, mesmo que esta possua diversas obrigatoriedades de relacionamento (restrições de FK).

4.3.6 Conversão de Dados

Para correta medição de Funções de Conversão, é importante verificar se a atividade relacionada realmente deve ser classificada como uma Função de Conversão de Dados ou se trata apenas de uma atividade de suporte, que não pode ser classificada como uma função.

As Funções de Conversão de Dados devem possuir dados atravessando a fronteira de aplicação e serem descartáveis (executadas uma única vez em produção). Esse tipo de funcionalidade possui a intenção primária de manter um ou mais ALIs da aplicação e, portanto, são classificadas como Entradas Externas.

Também é importante atentar-se à análise correta dos relatórios comumente emitidos por estas funcionalidades. Esses são consequência da atividade de migração, não sendo completos em si (autocontidos) e, portanto, não devem ser medidos como uma nova função, independente da migração em si.

A dica geral é fazer a analogia entre a funcionalidade de conversão com a atividade de migração de registros de forma manual (usuário entrando com dados por meio de uma tela de cadastro). Se não houvesse a funcionalidade, o usuário informaria os campos na tela (dados entrando pela fronteira) e faria toda esta atividade uma única vez (descartável), já que, depois de concluída a migração, os registros passariam a ser cadastrados conforme o negócio da aplicação. Ou seja, a função de conversão de dados é responsável apenas por automatizar a execução de toda a atividade de entrada de dados, caracterizada pela busca dos dados que entrarão pela fronteira da aplicação em alguma fonte externa (Ex: planilhas, tabelas, arquivos txt) e pelo fato de ser descartável, já que não terá mais utilidade após a sua execução.

4.3.7 Migração de Dados dissociada de Projeto

Neste caso, a migração de dados será identificada normalmente como uma função de conversão e deverá ser medida em PF.

O fato de a migração ter sido solicitada isoladamente em nada afeta a análise de Pontos de Função. Esta seria medida da mesma forma caso fizesse parte da demanda que solicitou a criação de um novo módulo. O cuidado a ser tomado é com relação à recontagem da mesma migração como uma Função de Conversão, caso esta já tenha sido medida e paga anteriormente.

4.3.8 Migração de Dados frequente/periódica

Nesta situação, não há Função de Conversão de Dados. As migrações são executadas periodicamente (mesmo que em períodos longos de um ano), o que confronta o caráter descartável das Funções de Conversão. Há na realidade uma Função Transacional responsável pela carga dos dados na aplicação, que fatalmente passará por manutenções funcionais ao início de cada ano, salvo situação em que esta atividade seja executada diretamente pela equipe de desenvolvimento, sem que haja uma funcionalidade responsável por tal tarefa.

4.3.9 Consultas/Pesquisas com diversas opções de filtro

Para correta análise de funcionalidades de pesquisa, compostas por diversos campos e opções de filtro, é necessário verificar se o seu requisito funcional é o que efetivamente foi solicitado pelo usuário da aplicação. O mais importante é conseguir separar corretamente quais opções adotadas durante a construção da funcionalidade foram tomadas com base em requisitos não funcionais, evitando a contagem de diversas Funções Transacionais em situações em que um único processo elementar pode ser identificado.

O usuário solicita uma funcionalidade que será responsável pela consulta de sistemas a partir de sua sigla ou gerente responsável pela aplicação. O retorno da pesquisa, independente da opção de filtro selecionado, é sempre o mesmo.

O que o usuário solicitou é que deve ser a informação utilizada como insumo para medição sendo realizada. O requisito do usuário é que determinará a quantidade de processos elementares existentes e não a forma pela qual a solução foi construída.

A dica geral é sempre fazer uma analogia com uma tela de pesquisa em que todos os campos estivessem sempre disponíveis e avaliar qual o impacto no negócio da aplicação. Desta forma, decisões não funcionais tomadas durante a implementação da funcionalidade ficam mais evidentes (Ex: limitação de espaço em tela, melhora a usabilidade do sistema, etc).

4.3.10 Gravação da Trilha de Auditoria

Normalmente a gravação de trilha de auditoria ocorre para todas (ou quase todas) as transações da aplicação. Não sendo, portanto, um requisito específico e particular dessas transações, mas sim requisitos gerais da aplicação.

Uma evidência disto, é que esse requisito não estará presente nos casos de uso que especificam as transações, mas em uma parte geral da especificação de requisitos. Logo não há transação adicional a ser considerada pela gravação da trilha de auditoria; assim como não haverá nestas transações a contagem de um arquivo referenciado para os dados de auditoria.

Caso haja um requisito específico e particular de gravar para determinado grupo lógico de dados informações de configurações passadas antes e após uma atualização, isso deve ser considerado na contagem. Isso se dá pela contagem de um novo Tipo de Registro nesse arquivo (afinal normalmente existe mais de um campo funcionalmente dependente dessa configuração anterior como a data em que o evento ocorreu ou quem foi o responsável pela substituição da configuração anterior por uma nova ou mesmo pela exclusão de um registro).

A geração desse registro está associada ao fato gerador e com isso não deve ser contada uma função especificamente para este fim, mas sim considerar a geração desse registro como um passo do processo responsável pelo fato gerador.

4.3.11 Consulta à Trilha de Auditoria

Caso seja de responsabilidade da aplicação em análise disponibilizar consultas e relatórios aos dados de auditoria, estas transações deverão ser consideradas também funções desta aplicação, pois representam requisitos específicos da mesma.

Caso as consultas aos dados de auditoria sejam efetuadas por outra aplicação, nada relativo a estas consultas deve ser considerado na contagem da primeira aplicação.

4.3.12 Dados da Trilha de Auditoria

Na situação que em alguns grupos lógicos de dados da aplicação exigirem a manutenção do histórico de configurações anteriores; e houver necessidade de integrar esses dados em consultas que façam uma referência cruzada desses dados históricos independentemente do grupo lógico de dados em particular e da atual configuração desses dados; e os dados históricos são mantidos mesmo quando os dados atuais não forem mais relevantes para o negócio, cabe a contagem de um grupo lógico de dados especificamente para a sua manutenção.

Especificamente para as transações que mantêm dados nesse arquivo, um novo Arquivo Referenciado será contabilizado ao avaliar a complexidade dessas funções.

4.4 Contagem Estimativa de Pontos de Função (CEPF)

Estimativa de Pontos de Função: significa fornecer uma avaliação aproximada do tamanho de um software utilizando métodos diferentes da Contagem de Pontos de Função do IFPUG.

O método CEPF visa aferir o tamanho em PF de maneira simplificada, com base no conhecimento dos requisitos iniciais do projeto. Inicialmente os requisitos funcionais iniciais, documentados nos documentos de escopo

preliminar, são mapeados nos tipos funcionais da Análise de Pontos de Função: Arquivo Lógico Interno (ALI), Arquivo de Interface Externa (AIE), Entrada Externa (EE), Consulta Externa (CE) e Saída Externa (SE) (Figura 01). Em seguida, os Pontos de Função são associados a cada função identificada, baseando-se nas tabelas de complexidade e de contribuição funcional do CPM 4.3.1 (Tabela 01).

O estimador deve realizar uma leitura no documento inicial de requisitos, buscando informações relevantes para a identificação de processos elementares. O processo elementar é definido como a menor unidade de atividade significativa para o usuário. O processo elementar deve ser completo em si mesmo, independente e deixar a aplicação em um estado consistente. Em outras palavras, os processos elementares são Funções Transacionais independentes, isto é, funções sequenciais pertencem a um mesmo processo elementar e funções independentes constituem processos elementares diferentes.

Uma vez identificado o processo elementar, o estimador deve buscar o entendimento deste para classificá-lo em uma das Funções Transacionais: Entrada Externa, Consulta Externa ou Saída Externa. Adicionalmente, o estimador deve descobrir os dados associados ao processo elementar, visando à determinação da complexidade funcional da função identificada. Caso não seja possível a identificação da complexidade da funcionalidade em questão, utiliza-se a complexidade *Média*.

Na análise do processo elementar também são identificados, os grupos de dados lógicos da aplicação e Funções de Dados, que são classificados como Arquivos Lógicos Internos ou Arquivos de Interface Externa. Caso não seja possível a identificação da complexidade da Função de Dados em questão, recomenda-se a utiliza-se a complexidade *Baixa*. É importante ressaltar que se o estimador identificar mais de um Registro Lógico no Arquivo Lógico Interno utiliza-se a complexidade *Média*.

A seguir são apresentadas dicas para ajudar no mapeamento dos requisitos funcionais da aplicação nos tipos funcionais da APF. As necessidades e funcionalidades especificadas para o projeto, contidas no Documento de Escopo Preliminar devem ser enquadradas em uma das seguintes tabelas:

Nº ALIs (Baixa)	x 7 PF
Nº ALIs (Média)	x 10 PF
Nº ALIs (Alta)	x 15 PF
Total PF	

Tabela 03 – Contagem dos Arquivos Lógicos Internos (ALI): Banco de Dados Lógico da Aplicação (tabelas e arquivos mantidos pela aplicação).

Considerações: Deve-se identificar os grupos de dados lógicos de aplicação nos modelos de dados ou diagrama de classes ou a partir dos requisitos funcionais, descritos na Especificação de Requisitos. Não se deve considerar arquivos físicos, arquivos de índices, arquivos de trabalho e tabelas de relacionamento sem atributos próprios (tabelas que existem para quebrar o relacionamento n:m e apenas transportam as chaves estrangeiras). As entidades fracas também não são consideradas um ALI. Se possível, deve-se descobrir os atributos lógicos, campos reconhecidos pelo usuário, e subgrupos de dados existentes para obter a complexidade funcional, segundo as regras de contagem do CPM 4.3.1. Caso não seja possível, usam-se os ALIs dos sistemas como complexidade *Baixa*.

Nº AIEs (Baixa)	x 5 PF
Nº AIEs (Média)	x 7 PF
Nº AIEs (Alta)	x 10 PF
Total PF	

Tabela 04 – Contagem de Arquivos de Interface Externa (AIE): Banco de Dados de outras Aplicações, apenas referenciados pela aplicação que está sendo estimada (tabelas e arquivos mantidos por outra aplicação).

Considerações: Deve-se identificar os grupos de dados lógicos de outras aplicações referenciados pela aplicação que está sendo estimada. Frequentemente, a referência de dados ocorre para a validação de informações em cadastros ou consultas. Algumas vezes, relatórios ou consultas referenciam dados externos de outras aplicações, também considerados AIE. Não são considerados arquivos físicos, arquivos de índice, arquivos de trabalho, tabelas de relacionamento sem atributos próprios e entidades fracas. Geralmente, os AIE dos sistemas possuem a classificação de complexidade *Baixa*, já que são considerados para a determinação da complexidade funcional do AIE apenas os atributos referenciados pela aplicação que está sendo contada.

Nº EEs (Baixa)	x 3 PF
Nº EEs (Média)	x 4 PF
Nº EEs (Alta)	x 6 PF
Total PF	

Tabela 05 – Contagem de Entradas Externas (EE): Funcionalidades que mantêm os Arquivos Lógicos Internos (ALI) ou alteram o comportamento da aplicação.

Considerações: Deve-se identificar as funcionalidades de manutenção de dados. São contadas separadamente a inclusão, alteração e exclusão de dados. A aplicação possui funções de entrada de dados que alteram o comportamento dela, por exemplo: processamentos batch, ou processamento de informações de controle? Caso positivo, estas funções também devem ser identificadas como Entradas Externas. Não havendo o conhecimento sobre o processo elementar (funcionalidade analisada), considerar as Entradas Externas identificadas com complexidade Média.

Nº CEs (Baixa)	x 3 PF
Nº CEs (Média)	x 4 PF
Nº CEs (Alta)	x 6 PF
Total PF	

Tabela 06 – Contagem de Consultas Externas (CE): funcionalidades que apresentam informações para o usuário sem a utilização de cálculos ou algoritmos. São os processos elementares do tipo “lê - imprime”, “lê - apresenta dados”, incluindo consultas, relatórios, geração de arquivos pdf, xls, downloads, entre outros.

Considerações: As funções que apresentam informações para o usuário, mas não possuem cálculos ou algoritmos para derivação dos dados referenciados, não alteram ALI e/ou não mudam o comportamento do sistema devem ser identificadas como Consultas Externas.

Não havendo conhecimento suficiente sobre o processo elementar (da funcionalidade analisada), considerar as Consultas Externas com complexidade *Média*.

Nº SEs (Baixa)	x 4 PF
Nº SEs (Média)	x 5 PF
Nº SEs (Alta)	x 7 PF
Total PF	

Tabela 07 – Contagem de Saídas Externas (SE): Funcionalidades que apresentam informações para o usuário com utilização de cálculos ou algoritmos para derivação de dados ou atualização de Arquivos Lógicos Internos ou mudança de comportamento da aplicação. São as consultas ou relatórios com totalização de dados, relatórios estatísticos, gráficos, geração de arquivos com atualização log, downloads com cálculo %, entre outros.

Considerações: Você está desenvolvendo uma funcionalidade para apresentar informações para o usuário: uma consulta ou relatório com totalização de dados, etiquetas de código de barras, gráficos, relatórios estatísticos, download com percentual calculado, geração de arquivo com atualização de log? Caso positivo, estas funções devem ser identificadas como Saídas Externas. Esta função deve ter cálculos ou algoritmos para processar os dados referenciados nos arquivos lógicos ou atualizar campos (normalmente indicadores) nos arquivos ou mudar o comportamento da aplicação. Caso não haja conhecimento da aplicação de APF ou sobre o processo elementar (funcionalidade analisada), considere as Saídas Externas com complexidade *Média*.

A estimativa de tamanho do projeto em PF deve ser gerada totalizando-se os PF obtidos nas Tabela 02 a Tabela 06.

4.5 Abordagem para Mudança de Requisitos

Uma mudança de requisito anterior à implantação do projeto gera retrabalho para a equipe de desenvolvimento, aumentando assim o esforço e o custo do projeto. Neste roteiro, as demandas de mudança de requisitos serão dimensionadas como *PF_RETRABALHO*, contadas à parte do projeto de desenvolvimento ou de manutenção.

O método de contagem de mudança de requisitos descrito neste roteiro tem os seguintes pressupostos:

- ✓ As demandas de mudança de requisitos são contagens à parte da contagem do projeto de desenvolvimento ou de manutenção e devem considerar as funcionalidades antes da mudança;
- ✓ A quantidade de *PF_RETRABALHO* apurada leva em conta o esforço já realizado no processo de desenvolvimento da funcionalidade até o momento da solicitação de mudança de requisitos aplicando o percentual das fases concluídas na fórmula do cálculo do *PF_RETRABALHO*. A distribuição de esforço apresentada na Tabela 02 estabelece os percentuais por fase, de forma a permitir a contagem de mudança de requisito conforme o estágio do projeto.

A contagem do projeto de desenvolvimento ou de manutenção deverá ser atualizada a cada demanda de mudança de requisitos, visando refletir as funcionalidades após a mudança.

- ✓ Para fins de planejamento ou de faturamento, a quantidade total de pontos de função será obtido da seguinte forma:

$$PF_TOTAL = PF_PROJETO + \sum PF_RETRABALHO$$

Em que: *PF_PROJETO* é a última versão da contagem do escopo do projeto.

A contagem de *PF_RETRABALHO* leva em conta as seguintes características:

- ✓ Requisito original: é o requisito do projeto de desenvolvimento ou de manutenção original, que pode ser incluir, alterar ou excluir funcionalidades de um aplicativo.
- ✓ Tipo da mudança do requisito: é a natureza da mudança de requisitos no projeto em andamento, que pode ser acrescentar um requisito, alterar um requisito definido ou desistir de um requisito.

A Tabela 08 resume os percentuais que devem ser aplicados sobre as funções alteradas (considerando o tamanho antes da mudança) para obtenção de *PF_RETRABALHO*:

Fator			Requisito Original		
			Incluir Função	Alterar Função	Excluir Função
Tipo de Mudança de Requisito	Acréscimo				
	Alteração	Alteração de Requisitos	75%	75%	
		Alteração de Interface	0,6 PF	0,6 PF	
	Desistência		130%	105%	30%

Tabela 08 – Percentuais definidos para mudança de requisitos.

Observações Importantes:

- 1) O registro das demandas de alteração de requisitos deve ser realizado em separado, sendo contado em uma planilha de *PF_RETRABALHO* à parte da contagem de PF do projeto. Apesar das medições em separado, elas ainda devem guardar vínculo com o projeto em andamento, fazendo parte da sua *Baseline* de tamanho.
- 2) O cálculo do *PF_RETRABALHO* deve registrar o percentual das fases concluídas do processo de desenvolvimento até o momento da mudança de requisitos, conforme Tabela 02.

A seguir são descritos os tipos de mudança nos projetos.

1) Acréscimo de funcionalidades ao escopo do projeto

As mudanças que não tragam impacto aos requisitos originais do projeto, caracterizadas pelo acréscimo de funcionalidades ao escopo do projeto de desenvolvimento ou de manutenção, serão acrescentadas na contagem de PF do projeto e não geram contagem de *PF_RETRABALHO*, ou seja, representam um trabalho adicional e não retrabalho.

2) Alteração de função

A contagem de *PF_RETRABALHO* referente à alteração deve considerar o percentual de 75% sobre o tamanho da função antes da alteração, independentemente do requisito original. Este item se refere somente à alteração de requisitos de funcionalidades que estavam sendo criadas ou alteradas no projeto original.

Em caso de mudanças em interface (cosméticas), considerar o percentual de 20% da contagem de uma função transacional de mais baixa complexidade (3 PF), ou seja 0,6 PF, independentemente da complexidade da função antes da alteração.

Sobre a quantidade de *PF_RETRABALHO* obtida, para fins de gestão e faturamento, deverá ser aplicado o percentual das fases concluídas até o momento da solicitação de mudança de requisitos, conforme Tabela 02.

A contagem de PF do projeto deve ser atualizada para refletir o novo grau de complexidade da função após a mudança.

3) Desistência de incluir, alterar ou excluir uma função

Em caso de desistência de incluir, alterar ou excluir uma função, deve-se verificar qual era o requisito original, pois o percentual a ser utilizado na contagem de *PF_RETRABALHO* varia para cada situação, conforme apresentado na Tabela 08. Além do trabalho de retirar o que foi requisitado (percentuais definidos na Tabela 08), deve-se considerar também em *PF_RETRABALHO*, o trabalho realizado até o momento da desistência desse requisito. Por fim, o requisito original deve ser removido do *PF_MELHORIA*. Enquadram-se nesta situação somente as desistências de incluir, de alterar ou de excluir funcionalidades que constavam no escopo do projeto.

Quando a mudança no projeto é deixar de incluir uma função, aplica-se o percentual de 130% ao tamanho da função original. Esse valor é resultado da soma do percentual de 100% da inclusão (escopo original) com os 30% correspondentes à exclusão dessa mesma função.

Quando a mudança no projeto é deixar de alterar uma função, aplica-se o percentual de 105% ao tamanho da função original. Esse valor é o resultado da soma do percentual de 75% da alteração (escopo original) com os 30% referentes à exclusão dessa mesma função.

Quando a mudança no projeto é deixar de excluir uma função, aplica-se apenas o percentual de 30% referente à exclusão da função original.

Em todos os casos, a contagem de *PF_MELHORIA* deve ser atualizada removendo-se as funções que não fazem mais parte do escopo do projeto.

Da mesma forma que no item 2 (Alteração de função), para fins de gestão e faturamento, sobre a quantidade de *PF_RETRABALHO* é aplicado o percentual das fases concluídas até o momento da solicitação de mudança de requisitos, conforme Tabela 02.

4) Desistência de alterar uma função seguida de exclusão da função

Quando a solicitação de mudança for não apenas para deixar de fazer o que estava no projeto original, mas também excluir a função da aplicação, deve-se considerar esses dois aspectos separadamente, como se fossem duas mudanças consecutivas:

- a) Conta-se a desistência de alterar a função conforme descrito no item 3 (Desistência de incluir, alterar ou excluir uma função), apurando a quantidade de *PF_RETRABALHO* correspondente e a atualização do *PF_MELHORIA*;
- b) Conta-se o acréscimo ao escopo do projeto (excluir a função da aplicação) conforme descrito no item 1 (Acréscimo ao escopo do projeto), atualizando-se *PF_MELHORIA*.

4.6 Fator de Criticidade

Em função da criticidade e da necessidade de alocação de recursos extras para atendimento de determinadas demandas em um prazo estipulado pela ANEEL poderá ser adotado um Fator de Criticidade de 1,25 (um vírgula vinte e cinco), que deverá ser multiplicado pelo tamanho funcional da demanda considerada crítica, a critério da ANEEL, de modo a remunerar adequadamente o aumento do esforço de atendimento.

5 Relatório de Pontos de Função

O resultado da contagem deverá ser registrado no artefato Relatório de Contagem de Pontos de Função definido pela ANEEL, considerando:

- ✓ O registro de todas as Funções de Transação inerentes ao escopo com, no mínimo, os seguintes dados: Requisito/Caso de Uso, Nome da Função, Status (Incluída/Alterada/Excluída), Tipo, Quantidade de ARL e Descrição, Quantidade de DER e Descrição, Complexidade, Nº de Pontos de Função e Observações Associadas;
- ✓ Caso a Função de Transação não se caracterize como processo elementar, a exemplo: função que mantém uma *Code Table*, é obrigatório seu registro no artefato de contagem com preenchimento de todos os dados da planilha.
- ✓ Registrar no campo *Obs* todas as evidências ou interpretações para a respectiva Função de Transação ou ainda qualquer informação importante para o entendimento da análise e justificativa da classificação da função.
- ✓ Modelo de informações exigidas no preenchimento das Funções de Transação do artefato Relatório de Contagem de Pontos de Função da ANEEL.

Funções de Transação										
Nome do Requisito / Caso de Uso	Nome da Função	Incluída/ Alterada/ Excluída	Tipo	ALR		DER		Complexidade	PF	Obs
				Qtd	Descrição	Qtd	Descrição			

- ✓ O registro de todas as Funções de Dados inerentes ao escopo com, no mínimo, os seguintes dados: nome da função, status, quantidade de RLR e descrição, e ainda, quantidade de DER e descrição incluindo o nome da tabela do modelo de dados relacionado na demanda.
- ✓ Caso a Função de Dados represente uma *Code Table*, é obrigatório seu registro no artefato de contagem com preenchimento de todos os dados da planilha.
- ✓ Registrar no campo observação todas as evidências ou interpretações para a respectiva Função de Dados ou ainda qualquer informação importante para o entendimento da análise e justificativa da classificação da função.

- ✓ Modelo de informações exigidas no preenchimento das Funções de Dados do artefato Relatório de Contagem de Pontos de Função da ANEEL.

Funções de Dados								
Nome da Função	Incluída/ Alterada/ Excluída	Tipo	RLR		DER		Complexidade	PF
			Qtd	Descrição	Qtd	Descrição		

6 Formação de base histórica

Todas as informações de contagem, inclusive aquelas que identificam e detalham as Funções de Dados e as Funções de Transação devem ser armazenadas em base histórica e fornecidas à ANEEL após a execução de cada serviço, ao final do contrato e sempre que solicitadas.

A empresa contratada para prestação dos serviços de mensuração deverá manter base histórica, em estrutura relacional, dos dados das contagens realizadas.