

Método de Pontos por Função a Partir de Especificação de Requisitos 1

software, por meio da cooperação entre universidades e empresas, com ênfase na utilização da Engenharia de Requisitos. A plataforma foi extremamente importante para um melhor conhecimento dos problemas enfrentados pelas organizações que participaram do projeto e da possibilidade da cooperação com as instituições de pesquisa para minorar esses problemas [4].

Identificados os problemas, os processos descritos em [5] foram executados para a implantação da Gerência de Requisitos no SERPRO tendo em vista a melhoria do processo preconizada para o Nível 2 do CMM. Com a efetiva utilização e ênfase dada aos requisitos durante a fase inicial do processo de desenvolvimento de software, vislumbrou-se a oportunidade e a necessidade de extrair métricas de tamanho de software a partir dos requisitos descritos em casos de uso [6], o que serviria de subsídios para as demais áreas-chaves de processo do CMM.

Concentrando esforços para descrever requisitos de software com caso de uso e medindo os requisitos utilizando a métrica Pontos por Função, projetos podem ser melhor gerenciados e controlados. O dimensionamento de projeto na fase de requisitos é um tópico que vem sendo bastante considerado quando o enfoque é Qualidade do Processo de Desenvolvimento de Sistemas. Com base no tamanho de um projeto, pode-se derivar uma série de indicadores gerenciais que proporcione uma postura pró-ativa dos gestores de desenvolvimento sob diversos aspectos.

Nos idos de 1979, em busca de uma métrica melhor, A. J. Albrecht (IBM) considerou a utilização dos aspectos externos visíveis de um software para gerar uma nova métrica, conhecida como Pontos de Função e que foi um dos primeiros métodos a medir e prever o desenvolvimento de software com alguma precisão [7].

Considerando esses tópicos, [6] propôs a medição de Pontos por Função a partir da especificação de requisitos baseada em casos de uso. Esta proposta visa antecipar o trabalho de medição de sistemas minimizando o esforço dos líderes de projeto, obtendo uma compreensão intuitiva do porte do projeto em termos de Pontos por Função a partir dos requisitos elicitados.

Na próxima Seção, são descritos trabalhos relacionados. Na Seção 3, faremos um resumo da Métrica Análise de Pontos por Função. Na Seção 4, apresentaremos uma proposta de medição de Pontos por Função a partir da descrição dos requisitos expressos em notação UML e um estudo de caso para facilitar o entendimento da proposta. E, finalmente, a Seção 5 é composta das conclusões e trabalhos futuros.

2 Trabalhos Relacionados

Recentemente, tem surgido grande número de extensões para o modelo de pontos por função. Vários autores têm proposto métricas para adaptar pontos de função a software orientado a objetos. Em [10] classes são tratadas como objeto, e serviços fornecidos a clientes como transações, enquanto em [11] cada classe é considerada como um arquivo interno, e mensagens enviadas através da fronteira do sistema são tratadas como

Medição de Pontos por Função a Partir de Especificação de Requisitos 1

vez que Pontos por Função possuem grandes bases históricas, é possível comparar a produtividade de utilizar métodos de casos de uso com outros métodos.

Ao saber o tamanho do projeto no início do ciclo de vida, melhores estimativas de tempo e custo podem ser desenvolvidas. Os casos de uso são mantidos atualizados, os Pontos por Função são mantidos atualizados, as estimativas do projeto podem ser facilmente atualizadas e os desacordos explicados.

O aumento do foco nos requisitos resulta em um software de maior qualidade. Casos de Uso e Pontos por Função podem ser usados juntos efetivamente para alcançar sucesso no gerenciamento dos requisitos e do projeto [8].

O principal aspecto deste trabalho é a utilização das informações contidas nos casos de uso para as regras de contagem de pontos por função tradicionais, o que permitiu a comparação de tamanho com os sistemas pontuados anteriormente que foram definidos a partir da especificação estruturada.

3 Análise de Pontos por Função (APF)

De acordo com a técnica Análise de Pontos por Função, uma aplicação de software, vista sob a ótica do usuário, é um conjunto de funções ou atividades do negócio que o beneficiam na realização de suas tarefas [20]. O manual do IFPUG classifica os seguintes tipos de elementos funcionais[21]:

1. Entrada Externa – EI (*External Input*) – transações lógicas onde dados entram na aplicação e mantém dados internos.
2. Saída Externa – EO (*External Output*) – transações lógicas onde dados saem da aplicação para fornecer informações para usuários da aplicação.
3. Consulta Externa – EQ (*External Query*) – transações lógicas onde uma entrada solicita uma resposta da aplicação.
4. Arquivos Lógicos Internos – ILF (*Internal Logical File*) – grupo lógico de dados mantido pela aplicação.
5. Arquivos de Interface Externa – EIF (*External Logical File*) – grupo lógico de dados referenciado pela aplicação, mas mantido por outra aplicação.

O manual do IFPUG fornece tabelas e diretrizes para determinar a complexidade de cada elemento funcional [7]. A complexidade dos ILF e EIF é baseada número de registros lógicos – RET (Record Element Types) e no número de itens de dados referenciados – DETs (Data Element Types) e a complexidade das transações é baseada no número de Arquivos Referenciados – FTR (File Types Referenced) e no número de Itens de Dados Referenciados – DET (Data Element Type).

Os elementos funcionais identificados são totalizados para calcular obtenção dos Pontos por Função não Ajustados. Então é calculado é calculado a partir de 14 (quatorze) características gerais dos projetos, que permitem uma avaliação geral da funcionalidade da aplicação: Comunicação de Dados, Processamento Distribuído, Atualização de Dados Online, Entrada de Dados Online, Volume de Transações, Eficiência do Usuário Final,

Medição de Pontos por Função a Partir de Especificação de Requisitos 1

SERPRO, Serviço Federal de Processamento de Dados, é uma empresa de informática vinculada ao Ministério da Fazenda, cuja função principal é a execução de serviços de tratamento de informações e processamento de dados para o governo federal. A Empresa está sediada em Brasília, com representações regionais em 10 capitais (Belém, Belo Horizonte, Curitiba, Fortaleza, Porto Alegre, Recife, Rio de Janeiro, Salvador, São Paulo e a própria capital federal) e um quadro de 8.774 funcionários. A Empresa encontra-se estruturada em Unidades de Gestão (UG) responsáveis por um segmento da Administração Pública. Cada segmento atende a pelo menos um órgão federal com características e necessidades próprias), por intermédio de projeções da UG em cada uma das unidades organizacionais. A natureza diversa desses clientes e a descentralização do desenvolvimento em suas diversas representações exigem do SERPRO a manutenção de um complexo parque de desenvolvimento e o envolvimento direto com as mais diferentes plataformas tecnológicas. Neste cenário, a adoção de práticas para padronização, organização e controle do processo de software torna-se fundamental para a perfeita integração entre as unidades de negócio. Dentre as Superintendências, a SUNAT (Superintendência de Negócios - Administração Tributária) iniciou, em 1998, o processo de implantação de um plano de medições definindo como métrica a ser adotada a Análise de Pontos por Função (APF). A proposta apresentada neste artigo está sendo implementada em 120 projetos desenvolvidos por cerca de 700 desenvolvedores. O Sistema Senha, que será usado como exemplo, foi desenvolvido, primeiramente, utilizando-se métodos tradicionais, e, a medição deste projeto foi realizada com a métrica Análise de Pontos por Função. O total de Pontos por Função encontrado foi 186,60. Considerando a necessidade da Empresa de iniciar a migração de suas aplicações para um novo ambiente e visando obter uma maior produtividade no desenvolvimento, casos de uso passaram a ser utilizados para documentar requisitos. O Sistema Senha foi um dos primeiros projetos a serem especificados com este paradigma e a ser aplicada a proposta apresentada neste artigo. O resultado obtido nessa nova contagem foi 186,40 Pontos por Função o que implica numa diferença mínima de pontuação.

A próxima Seção relata, passo a passo, o processo de medição de algumas funcionalidades do Sistema Senha que tem como objetivo fazer o controle de acesso dos usuários às Aplicações de uma empresa, onde ocorrem interações entre o cadastrador, que inicia a ação, e o processo de inclusão de usuário. O Sistema Senha tem como finalidade cadastrar os usuários, com as respectivas permissões de acesso às transações do Sistema Integrado de Informações - SII, garantindo que apenas usuários habilitados possam acessá-las. O Sistema de Controle de Acesso – SENHA controla, verifica e gerencia os usuários, os perfis e os relacionamentos entre estes e as transações; possibilita ao projeto SII a segurança necessária e imprescindível ao ambiente e às informações; e está baseado nos seguintes conceitos:

- USUÁRIO - Pessoa física cadastrada no Sistema Integrado de Informações (SII) e habilitada para acesso às informações.
- CADASTRADOR - Usuário do Sistema Senha que exerce as funções de inclusão e habilitação de outros cadastradores, usuários e perfis do SII.

Medição de Pontos por Função a Partir de Especificação de Requisitos 1

4.2 Diagramas de Caso de Uso (Use Case)

O desenvolvimento de Diagramas de Caso de Uso requer que a equipe de desenvolvimento de software dedique mais tempo nos estágios iniciais do plano de desenvolvimento e garanta que os requisitos sejam completos, rastreados e bem documentados.

Tabela 1. Descrição de Caso de Uso Cadastrar Usuário

<i>Cadastrar Usuário – Entrada Externa</i>
<p>Fluxo Principal de Eventos: O Use Case inicia o sistema e exibe para o cadastrador a tela para cadastrar um usuário no Sistema SENHA. O cadastrador deve informar o CPF e a senha do cadastrador Inclui Validar Acesso O cadastrador deve informar o Tipo do Usuário Ponto de extensão [Tipo UI] ← <interno> Ponto de extensão [Tipo UE] ← <externo> Ponto de extensão [Tipo UO] ← <Outra UA> Fazer a opção na tela: finalizar o use case ou inicializar</p> <p>Fluxo Excepcional de Eventos – O cadastrador pode cancelar a transação em qualquer momento.</p>
<i>Cadastrar Usuário Interno – Ponto de Extensão</i>
<p>Para inclusão de Usuário Interno, <<ATUALIZAR>>: CPF, SEQUENCIAL, DATA CRIAÇÃO, DATA EXTINÇÃO, DATA REGISTRO, DISPONIBILIDADE, DATA VALIDADE SENHA, SENHA ATUAL, TENTATIVAS INVÁLIDAS, ATIVO, TIPO. Através da interação com o ator PESSOA FÍSICA, que é o Sistema Cadastro de Pessoa Física, obteremos o nome do usuário.</p> <p>Fluxo Excepcional de Eventos – No Sistema Pessoa Física o CPF do usuário não é válido, o USE CASE informa e reinicia não permitindo a inclusão do usuário.</p> <p>Fluxo Excepcional de Eventos – O cadastrador pode cancelar a transação em qualquer momento.</p>

Aplicar Pontos por Função aumenta a qualidade do caso de uso. A aplicação da APF ajuda a determinar se o caso de uso está num nível de detalhe apropriado. Isto é, se é capaz de descrever como dados são passados do ator para dentro da fronteira ou como dados fluem de dentro da fronteira da aplicação para o ator, então está num nível adequado de detalhe. Por outro lado, se não é possível descrever este nível de funcionalidade, então o caso de uso necessita de mais detalhes. A idéia básica é que se é possível contar Pontos por Função, o caso de uso está num nível de detalhe apropriado. Depende de quão grande é o seu caso de uso é bom estabelecer um limite superior de tamanho de seu caso de uso de forma que seu tamanho seja controlado. Normalmente, o

Medição de Pontos por Função a Partir de Especificação de Requisitos 1

tamanho dos requisitos do usuário é crítico para o gerenciamento e controle do projeto, como visto anteriormente.

Tabela 3. Descrição de Caso de Uso Validar Acesso

<i>Validar Acesso - Consulta Externa (EQ)</i>
<p>Fluxo Principal de Eventos: O Use Case é iniciado pelo cadastrador e será usado para <<CONSULTAR>> se o cadastrador está autorizado a acessar a aplicação. Para esta validação o usuário deverá informar o CPF e a Senha. Caso sejam satisfeitas estas condições, o Senha liberará aos usuários, as transações vinculadas aos perfis a ele associado na unidade em questão.</p> <p>Fluxo Excepcional de Eventos – O cadastrador pode cancelar a opção de incluir transação em qualquer momento pressionando o BOTÃO CANCELAR, isto reinicializará o USE CASE.</p> <p>Fluxo Excepcional de Eventos – Caso o cadastrador não esteja autorizado a acessar esta transação específica, não será permitido o acesso para execução desta transação.</p>

4.3 Diagrama de Classes

A partir das informações obtidas durante a elicitação e documentação de requisitos utilizando casos de uso, esta proposta sugere que seja elaborada uma versão inicial do diagrama de classes de forma a melhor visualizar as informações de funções de dados para a contagem. O Diagrama de Classes deve ser elaborado de acordo com a UML. Como visto no início deste artigo, estaremos utilizando um subconjunto da notação UML, ou seja, estaremos tratando no Diagrama de Classes dos relacionamentos: associação, agregação/composição e herança.

Vale ressaltar que na nossa Empresa temos todo o modelo de dados mapeado, ou seja, no início de qualquer especificação de requisitos de um projeto já temos o Diagrama de Classes. Caso não houvesse o Diagrama de Classes no momento da especificação dos requisitos poderíamos mapear as funções de dados da métrica Pontos por Função a partir dos casos de uso. Na Figura 2, apresentamos o Diagrama de Classes do Sistema Senha que de acordo com os requisitos, são identificadas três classes principais: Usuário, Perfil e Transação. Como podem existir três tipos de usuários, com características distintas, estes estão definidos num relacionamento de herança. Como as associações Usuário-Perfil e Perfil-Transação possuem propriedades próprias, são definidas as classes associação Perfil-Usuário e Transação-Perfil. Algumas classes relacionam-se a classes de outros sistemas: a superclasse Usuário com a classe Pessoa Física (independente do tipo de usuário) e, para os tipos específicos, a classe Usuário Externo relaciona-se à classe Estabelecimento, e a classe Usuário OutraUA, às classes Unidade Administrativa e Setor. A classe transação está associada à classe Aplicação a qual pertence a outro sistema.

Medição de Pontos por Função a Partir de Especificação de Requisitos 1

As regras seguintes foram formuladas para garantir um mapeamento consistente e coerente entre o diagrama Use Case e os procedimentos de medição de Pontos por Função. Regras propostas para Fronteiras:

- I. Considere cada ator humano como um usuário do sistema
- II. Considere cada ator não-humano, o qual não é um sistema desenvolvido apenas para fornecer funcionalidades para o sistema medido, como uma aplicação externa.

A documentação requerida para este passo é o diagrama de Casos de Uso exibindo atores e Casos de Uso em um alto nível.

Após a identificação da fronteira da aplicação, a elaboração do diagrama de classe e diagrama de Caso de Uso devidamente descrito poderemos, então, iniciar o processo de medição.

A contagem de Pontos por Função poderá mostrar, através dos questionamentos necessários ao mapeamento para identificação das funções de dados e funções transacionais, se existe falta de clareza nos requisitos, porque Pontos por Função vincula os dados armazenados (objetos) a sua manipulação e uso dentro das funções do software.

4.5 Identificar as Funções de Dados

Um passo fundamental da medição de Pontos por Função é a identificação das funções de dados: os Arquivos Lógicos Internos (ILFs) e Arquivos de Interface Externa (EIFs). Cada função de dado deve ser classificada de acordo com sua complexidade funcional relativa, que é baseada no número de registros lógicos, Tipo de Elemento de Registro (RET), e no número de itens de dados referenciados, Tipos de Elementos de Dados (DETs).

Tabela 4. Arquivos Lógicos Internos e Arquivos de Interface Externa

ILFs	EIFs
USUARIO	PESSOA FISICA
PERFIL	ESTABELECIMENTO
TRANSACAO	SETOR UA
PERFIL POR USUARIO	APLICACAO
TRANSACAO POR PERFIL	UNIDADE ADMINISTRATIVA
Total Arquivos Lógicos Internos (ILFs) = 5	Total Arquivos de Interface Externa (EIFs) = 5

Como estamos utilizando uma prévia do diagrama de classe e o diagrama de Casos de Uso nas fases iniciais do desenvolvimento, eles serão a principal fonte de informação para identificação dos Arquivos Lógicos Internos (ILFs), Arquivos de Interface Externa (EIFs), Tipos de Elementos de Dados (DETs) e Tipo de Elemento de Registro (RETs).

Regras de mapeamento proposta para classes:

- III. Selecione toda classe como um arquivo lógico. As exceções são aquelas que fazem parte de um relacionamento de agregação/composição.
- IV. Atributos de classes são candidatos a Tipos de Elementos de Dados (DETs) para arquivos e para a transações pelas quais são lidos e/ou mantidos.

Medição de Pontos por Função a Partir de Especificação de Requisitos 1

De posse dos Arquivos Lógicos Internos (ILFs) e Arquivos de Interface Externa (EIFs) é necessário determinar suas complexidades onde precisaremos identificar os Tipos de Elementos de Registros (RETs) e os Tipos de Elementos de Dados (DETs):

Contagem de Tipos de Elementos de Registros (RETs):

Baseados nas regras de contagem do IFPUG, identificamos no diagrama de classe da figura 2 os RETs (grupos de elementos de dados reconhecidos pelo usuário) do Sistema SENHA, conforme mostrado na Tabela 5.

Contagem de Tipos de Elementos de Dados (DETs) (Data Element Types):

Baseados nas regras de contagem do IFPUG, identificamos os DETs (Tipos de elementos de dados) reconhecidos pelo cliente da classe Usuário do Sistema SENHA conforme mostrado na Tabela 6.

Adicionalmente será necessário identificar como um diagrama de classes que possua relacionamentos de associação, agregação, composição e herança poderá ser mapeado para as funções de dados: Arquivos Lógicos Internos (ILFs) e Arquivos de Interface Externa (EIFs) e seus Tipos de Elementos de Registros (RETs) e os Tipos de Elementos de Dados (DETs). Maiores detalhes podem ser obtidos em [6].

Associação. Regras de mapeamento proposta para relacionamentos de associação:

- VI. Selecione a classe associação como um arquivo lógico, considerando, para a contagem dos Tipos de Elementos de Dados (DETs), os atributos da classe associação e os atributos que são chaves primárias das classes associadas.
- VII. selecione as classes associadas como arquivos lógicos.

A classe Transação-Perfil é uma classe associação. Esta será contabilizada como um Arquivo Lógico Interno (ILF), considerando como Tipos de Elementos de Dados (DETs) os atributos Data-Início, Data-Fim e Data-Registro, Código-Perfil (chave da classe associada Perfil), Código-Transação (chave da classe associada Transação).

Agregação. Regras de mapeamento propostas para relacionamentos de agregação/composição:

- VIII. Uma classe que agregada a outra classe, é uma candidata a um Tipo de Elemento de Registro (RET) para o arquivo relacionado à classe agregada e não a um Arquivo Lógico Interno.

Herança. Regras de mapeamento proposta para relacionamentos de herança:

- IX. Uma classe abstrata não é uma candidata a um arquivo lógico. É uma candidata a um Tipo de Elemento de Registro (RET) em cada classe que herda suas propriedades.
- X. Subclasses de uma classe concreta são candidatas a arquivos lógicos ou a um Tipo de Elemento de Registro (RET) daquela classe. Se as subclasses não são contadas como arquivos lógicos, são subgrupos opcionais do arquivo relacionado à superclasse.

A classe Usuário é considerada como Arquivo Lógico Interno (ILF) e as subclasses Usuário Local, Usuário Externo e Usuário Outra UA como RETs da classe Usuário, pois estas são subgrupos opcionais da classe Usuário.

Candidatos adicionais a arquivos. Alguns dados que são considerados por convenção do Pontos por Função, como arquivos internos/externos podem não ser representados em

Medição de Pontos por Função a Partir de Especificação de Requisitos 1

expressão <<CONSULTA>> e a opção de exibir informações com a expressão <<SAIDAS>>.

Definição da complexidade das Entradas Externas (EIs), Consultas Externas (EQs) e Saídas Externas (EOs)

A complexidade das funções transacionais é baseada na determinação de Tipos de Elementos de Dados (DETs) e Tipos de Arquivos Referenciados (FTRs). Esta informação deve ser extraída da documentação detalhada dos Diagramas de Casos de Uso e do Diagrama de Classes.

4.6.1 Contagem de FTRs (File Types Referenced) para Entradas Externas (EIs)

Cada Entrada Externa deve ser classificada de acordo com sua complexidade funcional relativa, que é baseada no número de arquivos referenciados (FTRs) e no número de Itens de dados referenciados (DETs).

Tabela 7. Número de arquivos referenciados das EIs

Entradas Externas (EI)	ILF apenas mantido	ILF ou EIF apenas referenciado	ILF mantido ou referenciado	Total
Cadastra Usuário	Usuário	Pessoa Física Estabelecimento Setor UA Unidade- Administrativa		5

O número de arquivos referenciados é o somatório do número de Arquivos Lógicos Internos e de arquivos de Interface Externa atualizados ou consultados na Entrada Externa, conforme apresentado na Tabela 7, o exemplo relativo ao diagrama de Casos de Uso descrito na Figura 1.

Tabela 8. Contagem dos DETs das EIs

Cadastra Usuário
CPF
SEQUENCIAL
DATA CRIACAO
DATA EXTINCAO
DATA REGISTRO
DISPONIBILIDADE
DATA VALIDADE SENHA
SENHA ATUAL
ATIVO
TIPO
CNPJ
UNIDADE ADMINISTRATIVA EXERCÍCIO
SETOR EXERCÍCIO
UNIDADE ADMINISTRATIVA LOTAÇÃO
Total = 14

Método de Pontos por Função a Partir de Especificação de Requisitos 1

Para contagem dos Tipos de Elementos de Dados (DETs) de Saídas Externas (EOs), iremos considerar as informações do diagrama de classes da Figura 2, e a descrição do use case da Tabela 2, conforme mostrado na Tabela 9.

De acordo com o número de itens de dados (DETs) e de arquivos referenciados (FTRs), classificam-se as Saídas Externas em simples, médias ou complexas.

Tabela 10. Número de arquivos referenciados das EOs

Saída Externas (EI)	ILF apenas mantido	ILF ou EIF apenas referenciado	ILF mantido ou referenciado	Total
Relatório		Usuário	Pessoa Física	3
Usuário			Estabelecimento	

Para o exemplo acima teremos 3 Tipos de Arquivos referenciados (FTRs) e 7 Tipos de Elementos de Dados (DETs) o que implica em uma complexidade MÉDIA, ou seja, corresponde a 5 Pontos por Função para a Saída Externa Emite Relatório de Usuário.

4.6.3 Contagem de FTRs (File Types Referenced) para Consultas Externas (EQs)

Cada Consulta Externa deve ser classificada de acordo com sua complexidade funcional relativa, que é baseada no número de arquivos referenciados e no número de itens de dados referenciados.

Tabela 11. Contagem de FTRs

Consulta Externa (EI)	ILF ou EIF apenas referenciado	Total
Valida Acesso	Usuário	1

O número de arquivos referenciados é o somatório do número de Arquivos Lógicos Internos e de Arquivos de Interface Externa acessados na Consulta Externa, conforme apresentado na Tabela 11, o exemplo relativo ao diagrama de caso de uso descrito na Figura 1.

Table 12. Contagem de DETs

EQ
valida acesso
CPF
SENHA
VALIDACAO
Total = 3

Contagem de Tipos de Elementos de Dados (DETs) para Consultas Externas (EQs):

O número de itens de dados referenciados é o número de parâmetros informados para obtenção do resultado da Consulta Externa mais os itens de saída, contando os repetidos apenas uma vez.

Para contagem dos Tipos de Elementos de Dados (DETs), iremos considerar as informações do diagrama de classe da Figura 2 e a descrição do caso de uso da tabela 3, conforme mostrado na Tabela 12.

Medição de Pontos por Função a Partir de Especificação de Requisitos 1

No SERPRO, a SUNAT (Superintendência de Negócios - Administração Tributária) iniciou, em 1998, o processo de implantação de um plano de medições definindo como métrica a ser adotada a Análise de Pontos por Função (APF), baseando-se no padrão internacional estabelecido pelo IFPUG (International Function Points Users Group). Quando utilizada em combinação com outras medidas, a Análise de Pontos por Função (APF) pode ser utilizada para determinar, por exemplo, o nível de produtividade da equipe, o esforço requerido para o desenvolvimento do sistema, o custo, a taxa de produção e de manutenção ou até a qualidade do sistema do ponto de vista do usuário. Com a compatibilização da Contagem de Pontos por Função no paradigma Tradicional e Orientada a Objeto, aproveitamos a cultura existente na Empresa em Pontos por Função.

A métrica de Análise de Pontos por Função já está sendo amplamente utilizada por todos os Pólos de Desenvolvimento da SUNAT, no Brasil, além de outras Superintendências. Foram pontuados 100% dos projetos da SUNAT. Com a familiarização dos líderes de projeto e criação da base de histórico (*baseline*), nessa primeira fase, está sendo possível avaliar o acervo de projetos, além de gerar subsídios para a tomada de decisões. Para tal foi utilizada uma ferramenta chamada Estimativa[21], desenvolvida pelo SERPRO, para ajudar na contagem e armazenamento de dados históricos.

Podemos verificar que com a medição dos Pontos por Função a partir da especificação dos requisitos obtivemos, na fase inicial do projeto, uma estimativa do total de Pontos por Função do projeto (186,40), praticamente igual a medição do projeto desenvolvido com o paradigma tradicional (186,60). Vale salientar que o projeto foi desenvolvido simultaneamente por duas equipes distintas, onde uma utilizou a especificação de requisitos tradicional e a outra a especificação de requisitos orientada a objeto. Na conclusão do projeto o total de pontos por função obtidos foi de 201. Esta experiência foi realizada para podermos verificar a validade da aplicação da proposta de medição apresentada em [6].

Durante o processo de implantação das práticas necessárias para atingir o Nível 2 do CMM, utilizamos esta Proposta de Medição em mais 120 projetos da organização e verificamos a exatidão das estimativas realizadas nas pontuações dos diversos projetos na fase de requisitos.

Bibliografia

1. Greer, D., Bustard, D.W., Sunazuka, T.: Prioritisation of System Changes using Cost-Benefit and Risk Assessments. Fourth IEEE International Symposium on Requirements Engineering RE'99. Limerick, Ireland. (1999) 180--187
2. Tavares, H., Paim, F., Carvalho, A.: Implantando CMM Nível 2: A Estratégia SERPRO. I Simpósio Brasileiro de Qualidade de Software SBQS. Gramado, Brasil. (2002)
3. Leite, J., Castro, J., Pinheiro, F.: Plataforma Tecnológica em Engenharia de Requisitos – Estratégias para o Aumento da Qualidade no Desenvolvimento de Sistemas. <http://www.cic.unb.br/~facp/per/perhome.html>

