

# Experiências do uso da técnica Análise de Pontos de Testes e uma ferramenta de apoio

Anybal Rocha, Priscila Souza, Carlos Henrique da Silva Diniz, Marcelo Werneck  
Barbosa

Instituto de Informática – Pontifícia Universidade Católica de Minas Gerais (PUC  
Minas – Belo Horizonte, MG – Brazil

{anybalrocha, priscilinhapsouza}@gmail.com, chenri@cemig.com.br,  
mwerneck@pucminas.br

**Abstract.** Estimating test effort is not a simple task since there are several factors that might influence the result. One technique used to estimate test effort is Test Point Analysis. This work presents a tool to support this technique. Tests carried out with this tool demonstrated that not only it performs correctly the estimation but also provides more agility to the whole estimation process. It was also demonstrated the use of this technique and tool in software development projects in order to support project management. Case studies were carried out to show the applicability and results obtained by the technique. Results have shown that it was possible to achieve more precise, reliable and consistent estimations.

**Resumo.** Estimar o esforço de testes não é uma tarefa simples, pois há diversos fatores que podem interferir no resultado. Uma técnica de estimativa de esforço para teste de software existente é a Análise de Pontos de Teste. Este trabalho apresenta uma ferramenta de apoio a esta técnica. Testes realizados com a ferramenta demonstraram que ela, não somente apresenta resultados corretos, mas também proporciona mais agilidade ao processo de estimativa. Foi demonstrado ainda o uso desta técnica e ferramenta em projetos de desenvolvimento de software como apoio ao gerenciamento de projetos. Foram realizados estudos de caso para ilustrar a aplicabilidade e resultados alcançados pela técnica. Os resultados mostraram que foi possível alcançar estimativas mais precisas, confiáveis e consistentes.

## 1 Introdução

As atividades de testes de software vêm sendo enriquecidas por metodologias próprias, ferramentas de automação e equipes independentes e capacitadas, agregando mais qualidade ao processo de desenvolvimento e ao produto final [9].

Assim como os projetos de desenvolvimento de software, os projetos de teste possuem desafios e problemas em relação a custo e prazo. Por isso, é importante definir um processo de teste e estabelecer estimativas precisas e controláveis ao longo do projeto. Os gerentes de testes devem ser capazes de planejar seus recursos apropriadamente, estimar o esforço necessário para executar um conjunto de testes e justificar demandas por mais recursos ou extensões de prazo [2].

Para que os testes sejam bem executados, esses devem ser gerenciados e planejados. É importante ter uma estimativa mais precisa do tempo necessário para realizar os testes para evitar atrasos nas entregas do projeto e a insatisfação dos clientes.

Estimar o esforço de testes não é uma tarefa simples, pois há diversos fatores que podem interferir no resultado. No entanto, há algumas técnicas que podem auxiliar a obter estimativas mais precisas ao considerar fatores que influenciam as atividades de testes. Entretanto, estimativa de esforço específica para teste de software ainda é um tema pouco explorado. Existem poucos relatos de seu uso na literatura e, conseqüentemente, pouco conhecimento sobre a efetiva e correta utilização de técnicas de estimativas de testes pelas organizações [9].

Uma técnica de estimativa de esforço para teste de software existente é a Análise de Pontos de Teste (APT) [11]. Apesar da afirmação dos autores da técnica de que a mesma foi utilizada por algumas organizações com bons resultados, não foram encontrados relatos detalhados ou medições que comprovassem tais afirmações [9]. Ao contrário, observaram-se relatos de estimativas divergentes com o uso desta técnica [5].

Trabalhos anteriores propuseram análises e melhorias nesta técnica [8] [9] e, através destas adaptações realizadas, conseguiram obter estimativas mais próximas dos esforços efetivamente gastos nos projetos avaliados. Nestes trabalhos, foi elaborada inicialmente uma planilha Excel para auxiliar o processo de elaboração da estimativa de esforço de teste. Percebeu-se, no entanto, que esta planilha, mesmo apresentando resultados corretos, possui algumas restrições, tais como dificultar a comparação ou análise de estimativas de vários projetos ou o registro de estimativas para consultas futuras. Sendo assim, neste trabalho, é apresentada uma ferramenta para automatizar o processo de estimativas com a técnica Análise de Pontos de Testes. Além disso, este trabalho apresenta ainda a aplicação da técnica Análise de Pontos de Testes em projetos em andamento da área de desenvolvimento de software de uma grande empresa brasileira do ramo de energia elétrica.

O trabalho está organizado da seguinte maneira. A Seção 2 apresenta conceitos de estimativas em particular de estimativas de testes de software descrevendo a técnica Análise de Pontos de Testes. A Seção 3 apresenta alguns trabalhos relacionados. A Seção 4 descreve as funcionalidades e características da ferramenta APTool enquanto a Seção 5 apresenta os resultados de seu uso, testes realizados e aplicação em projetos em andamento. A Seção 6 apresenta as conclusões do trabalho.

## **2 Estimativas**

Uma das principais causas dos projetos não atingirem seus objetivos dentro do cronograma e do orçamento previstos são estimativas inadequadas baseadas em informações insuficientes, baseadas apenas na experiência dos responsáveis estimativas realizadas e sem o uso de métodos específicos.

Quanto mais informações sobre o sistema em desenvolvimento ou em testes, mais precisa será a estimativa [6]. A exatidão dessa estimativa se estreita à medida que as fases do ciclo de vida do sistema progredem em virtude de conhecimento adquirido e das informações disponíveis.

A estimativa de Ponto de Função mede o tamanho do software pela quantificação de suas funcionalidades externas, baseadas no projeto lógico ou a partir do modelo de dados. Essa estimativa abrange a funcionalidade específica requerida pelo usuário para o projeto. A funcionalidade requerida diz o que será, ou é entregue para o usuário [10]. O tamanho de um sistema é determinado em pontos de função. A contagem do tamanho é utilizada para estimar o esforço para desenvolver o sistema.

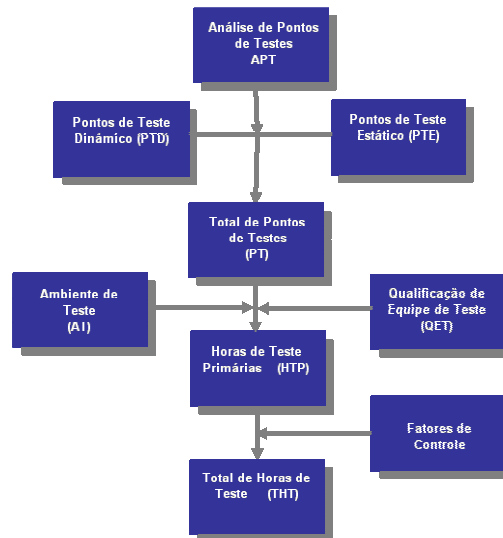
A análise de pontos de casos de uso é uma extensão da análise de pontos de função e estima o tamanho de um sistema baseado nas especificações de caso de uso [4].

Atualmente, existem empresas especializadas em prestar serviços de testes de software. Existem ainda empresas que preferem terceirizar estas atividades. Assim, o cenário atual faz com que técnicas, processos e ferramentas específicas para estimativas de testes de software sejam necessárias. Na próxima seção, é apresentada a técnica Análise de Pontos de Testes.

### **2.1 Estimativas para testes**

As atividades de testes de software fazem parte do ciclo de vida do desenvolvimento, muitas vezes, as organizações não realizam estimativas específicas quanto ao esforço de testes. Geralmente, elas fazem uma estimativa de esforço baseada em pontos de função para todo o desenvolvimento de software e designam uma porcentagem deste esforço para as atividades de testes. Além disso, essas estimativas incluem apenas os testes executados pela equipe de desenvolvimento (testes unitário), por isso, há necessidade de uma medida específica para o esforço de teste de software referente à equipe de testes [12].

Nesta seção, é apresentada uma técnica de estimativa de testes, a Análise de Pontos de Teste (APT). A APT é uma técnica de estimativa de tamanho de teste que gera previsões de esforço e tempo. É utilizada para estimar o esforço para definir, desenvolver e executar testes funcionais, baseados na complexidade do desenvolvimento de software. A APT possui como entrada o resultado da aplicação da técnica de Análise de Pontos de Função. A Análise de Pontos de Teste considera relevante, além do tamanho funcional determinado pelos pontos de função, outros dois elementos: a estratégia de testes e a produtividade [11]. A Figura 1 apresenta uma visão geral do modelo de Análise de Pontos de Teste.



**Figura 01 - Visão geral do modelo de Análise de Pontos de Teste**

O resultado da aplicação da Análise de Pontos de Teste é medido em unidade de esforço (horas), diferentemente da Análise de Pontos de Função, que mede o tamanho funcional do projeto de software.

O maior benefício da Análise de Pontos de Teste está em conseguir reunir de forma sistemática fatores que influenciam o esforço específico de uma das etapas do processo de desenvolvimento, produzindo resultados mais precisos. Além disso, permite avaliar o esforço de teste por atividade e leva em consideração as etapas de planejamento e controle que são essenciais para o sucesso de qualquer projeto e devem ser consideradas em técnicas de estimativas de esforço. A APT é uma técnica que permite estimar o esforço independentemente do processo de desenvolvimento, considera as características de qualidade do sistema, o fator do ambiente de teste e a qualificação e produtividade da equipe envolvida.

Os pontos de teste podem ser classificados como estáticos ou dinâmicos. O número total de pontos de teste é dado pelo somatório destes. As métricas de teste de software dependem de uma série de fatores relevantes fundamentais para se alcançar uma precisão mais exata.

Os pontos de teste dinâmicos são calculados com base em cada função individual do sistema a ser testado. As funções medidas na análise de pontos de função são tratadas em pontos de teste, além das funções dependentes e a qualidade dos requisitos relacionados com as características de qualidade a serem testadas dinamicamente. Além das funções medidas na Análise de Pontos de Função (entradas externas, saídas externas e consultas externas), os pontos de teste dinâmicos têm como base as funções dependentes e a qualidade dos requisitos relacionados com as características de qualidade a serem testados dinamicamente [12].

As funções dependentes estão relacionadas com o grau de dependência que têm das funções correlatas medidas na Análise de Pontos de Função. Existem cinco fatores que compõem as funções dependentes:

- Importância do usuário: Fator para medir o grau de importância para o usuário da função a ser testada. Pode ser definido com os próprios desenvolvedores ou com os usuários.

- Intensidade de uso: Fator que determina a intensidade de uso esperada por cada função. Em um intervalo de tempo quaisquer, existem funções que são utilizadas repetida ou eventualmente.
- Interface: Fator que determina o nível de inter-relacionamento entre os arquivos lógicos e as funções que estão sendo medidas. Deve ser considerado o número de arquivos afetados pela função que está sendo medida e o número de funções que afetam os arquivos lógicos.
- Complexidade: Fator que determina a complexidade do algoritmo da função. Originalmente, é medido pela quantidade de comandos de condição existentes no algoritmo. Na técnica APT adaptada, é medido pela contagem de pontos de função.
- Uniformidade: Fator que mede o nível de reutilização do material de teste.

Os pontos de teste estáticos, por sua vez, consideram o sistema como um todo e devem ser utilizados somente quando a equipe de testes adotar processos de revisão para documentação e códigos. O cálculo dos pontos de teste estáticos é baseado nos critérios de qualidade para a avaliação de funcionalidade, desempenho, segurança e aderência a partir de *checklists*.

Para calcular a estimativa do número de horas de testes primárias, após a contagem dos pontos de testes dinâmicos e estáticos, deve-se considerar ainda a qualificação da equipe e do ambiente de testes. O valor da qualificação da equipe de teste é determinado por uma base histórica e varia de acordo com a experiência e a qualificação da equipe. Para o ambiente de testes, são considerados fatores como ferramentas de testes, qualidade da documentação, ambiente de desenvolvimento, linguagem de programação entre outros.

As horas primárias de teste devem ser corrigidas, por fim, através da inclusão das atividades de planejamento e controle, no qual são considerados o tamanho da equipe de testes e as ferramentas de gerência utilizadas, como ferramentas de registro de tempo e gerência de defeitos.

Na próxima seção, são apresentados alguns trabalhos relacionados a estimativas de testes, mais especificamente à técnica APT.

### 3 Trabalhos Relacionados

Este trabalho faz parte de um esforço maior de pesquisa da técnica Análise de Pontos de Testes. Em trabalho anterior [9], foram propostas e avaliadas adaptações nesta técnica de modo a facilitar seu uso bem como incorporá-la a um processo de testes. Este trabalho anterior teve como objetivo definir um processo de teste que possibilitasse a aplicação de uma técnica de estimativa de esforço, a Análise de Pontos de Testes. Esta técnica pode ser considerada de difícil utilização, pois não é clara quanto à realização dos cálculos da estimativa e também não leva em consideração alguns aspectos importantes do processo de testes. Por esses motivos, foram necessárias algumas adaptações.

Alguns estudos de caso foram realizados para ilustrar a aplicabilidade do processo e da técnica adaptada. Eles foram aplicados em uma organização na qual foi necessária uma mudança de cultura para aplicação da técnica. Todos os envolvidos no processo de desenvolvimento de software foram convencidos da importância de se preencher corretamente todos os artefatos de entrada no processo de teste de software para se obter todas as informações necessárias para a realização das estimativas. Os testadores também foram orientados sobre a importância do exato apontamento das horas gastas na realização das atividades.

Os estudos de caso foram realizados em 3 ambientes diferentes. Em todos os estudos, foi aplicado o processo proposto juntamente com a técnica APT adaptada. Os resultados foram satisfatórios, pois alcançaram estimativas mais próximas do esforço realizado, contradizendo alguns trabalhos relacionados [5] e [13] nos quais a margem de erro chegou até a 511% de diferença. Isso mostra que o fato da empresa ter um processo bem definido, com informações disponíveis para a aplicação da técnica Análise de Pontos de Testes e um bom entendimento da técnica podem permitir alcançar resultados mais precisos e confiáveis.

O objetivo do trabalho [7] foi propor uma análise do ensino de estimativas usando a técnica Análise de Pontos de Teste. Foi objetivo identificar pontos complexos na técnica, nos quais os alunos encontram maior dificuldade de aplicação assim como etapas com maior índice de erros, ou seja, valores divergentes dos valores esperados. Como contribuições, este trabalho proporcionou o treinamento de 262 alunos na técnica APT e a identificação de pontos de dificuldade de compreensão da técnica APT.

Existem algumas ferramentas de apoio à contagem de pontos de função como [1], porém desconhece-se alguma ferramenta automatizada para auxiliar o processo de estimativa com a técnica Análise de Pontos de Testes. Este trabalho apresenta ainda a implementação e uso da ferramenta APTool, descrita na próxima seção.

## 4 A Ferramenta APTool

A ferramenta proposta neste trabalho foi desenvolvida de forma a automatizar o processo de estimativa com a técnica de Análise de Ponto de Testes adaptada. Outro objetivo incluso no desenvolvimento desta ferramenta foi tornar os cálculos transparentes ao usuário, de forma que o mesmo terá conhecimento apenas do resultado final. Além de executar os cálculos e gerar a estimativa de esforço para teste de software com base nas informações de projetos e funcionalidades deste, a ferramenta armazena as estimativas calculadas, permitindo uma análise comparativa e recuperação desta informação. Permite ainda a geração de bases históricas das estimativas nos projetos.

A seguir, são apresentadas as principais funcionalidades bem como sua descrição técnica.

### 4.1 Principais Funcionalidades

A ferramenta é constituída de 3 funcionalidades principais: Gerenciar Projeto, Gerenciar Funcionalidade, Gerar Estimativa.

A funcionalidade Gestão de Projetos permite que os projetos sejam mantidos pelo sistema. Esse cadastro é importante, pois a análise das técnicas selecionadas pode ser feita em relação aos parâmetros dos projetos cadastrados. Em Gerenciar Funcionalidade, são cadastradas todas as funções do sistema. Essas funções são as mesmas consideradas na contagem de pontos de função, utilizada como entrada para cálculo da estimativa.

A principal funcionalidade do sistema é Gerar Estimativa. Nesta funcionalidade, o usuário deve preencher todas as informações necessárias pela técnica para gerar a estimativa final de horas. A Figura 2 ilustra o preenchimento de algumas informações das funções do sistema.

ESTIMATIVA  
Projetos Funções Estimativas Sair

ADICIONAR/EDITAR FUNCIONALIDADES

Projeto: P. Planilha 01 PF Total: 72

Funcionalidade	Qualidade	Ambiente de Teste
Nome da Função:	<input type="text" value="Listar Medico"/>	
Sigla:	<input type="text" value="LM"/>	
Número de PF:	<input type="text" value="4"/>	
Importância do Usuário (Ue):	<input type="text" value="Baixa"/>	
Intensidade de Uso (Ug):	<input type="text" value="Alta"/>	
Complexidade da Interface (I):	<input type="text" value="Baixa"/>	
Complexidade da Funcionalidade (C):	<input type="text" value="Média"/>	
Uniformidade (U):	<input type="text" value="O material de teste não pode ser reutilizado"/>	

Figura 2. Preenchimento de informações na ferramenta

Ao final do preenchimento de todas essas informações, o sistema exibe a estimativa de horas de teste para o projeto em horas bem como algumas informações do sistema usadas para gerar a estimativa, conforme exibido na Figura 3.

**ESTIMATIVA**  
 Projetos    Funções    Estimativas    Sair

**Estimativa de Esforço para Teste de Software 20/05/2010**

Projeto:	P. Planilha 01		
Pontos de Função:	72		
Total de horas:	<b>85,47</b>		
<b>Qualidade</b>			
		<b>Implícita</b>	<b>Checklist</b>
Funcionalidade:	A qualidade dos requisitos tem importância média	<input type="checkbox"/>	<input type="checkbox"/>
Desempenho:	A qualidade dos requisitos tem importância média	<input type="checkbox"/>	<input type="checkbox"/>
Segurança:	A qualidade dos requisitos tem importância média	<input type="checkbox"/>	<input type="checkbox"/>
Aderência:	A qualidade dos requisitos tem importância média	<input type="checkbox"/>	<input type="checkbox"/>
<b>Ambiente de teste</b>			
Tamanho da Equipe de Teste:	Entre 1 e 4 técnicos		
Produtividade da Equipe de Teste:	Alta		
Ferramenta de Teste:	Não será utilizada ferramenta de automação de teste		
Teste de Precedencia:	Não existe um plano para o teste precedente		
Documentação de Teste:	São usados padrões de documentação e templates e não acontecem revisões periódicas da documentação		
Ambiente de Desenvolvimento:	Sistema desenvolvido com linguagem de 4 geração integrada a um sistema de gerência de banco de dados		
Ambiente de Teste:	O ambiente de teste é completamente novo e experimental		
Testware:	Não existe testware disponível		
Ferramenta de Gerência:	Apenas uma das ferramentas citadas acima está disponível		

**Figura 3. Estimativa final exibida pela ferramenta**

Um diferencial da ferramenta em relação à planilha é permitir gerar e armazenar várias estimativas para o mesmo projeto, por exemplo, para comparar estimativas de diferentes Analistas ou mesmo estimativas realizadas em momentos distintos do ciclo de vida do software. A Figura 4 exibe como o sistema permitir comparar estimativas geradas para o mesmo projeto.

**ESTIMATIVA**  
 Projetos    Funções    Estimativas    Sair

**Estimativa de Esforço para Teste de Software 20/05/2010**

Projeto:	Projeto 01		
Pontos de Função:	49		
Total de horas:	<b>12,83</b>		
Projeto:	P. Planilha 01		
Pontos de Função:	72		
Total de horas:	<b>85,47</b>		

**Figura 4. Comparações de estimativas para o mesmo projeto**

## 4.2 Descrição Técnica

O ambiente integrado de desenvolvimento utilizado foi o Microsoft Visual Studio 2008. A linguagem adotada foi a C# .NET (Framework versão 3.5). Para formatação do leiaute, foi utilizado um CSS. AJAX também foi utilizado em algumas páginas para melhorar a navegação e atualização, principalmente nas grades de listagens que aparecem nas telas de pesquisa. ADO.NET foi utilizado como objeto de persistência no banco de dados. Por fim, como SGBD, foi utilizado o Microsoft SQL Server 2005.

## 5 Testes e Resultados

Após a conclusão do desenvolvimento da ferramenta, foram realizados alguns testes para verificar o funcionamento do sistema e benefícios esperados. Foram realizados testes de confiabilidade da ferramenta, ou seja, para verificar se a planilha gera uma estimativa igual, para um mesmo projeto, à estimativa gerada pela planilha Excel. Preocupou-se ainda em medir o tempo demandado para gerar a estimativa a partir do cadastro do projeto a fim de confirmar a agilidade da ferramenta.

É relatado ainda o uso da ferramenta APTool em projetos em andamento na área de desenvolvimento de software de uma grande empresa.

### 5.1 Testes da Ferramenta

Para realizar os testes, foi realizado inicialmente o cadastro de um projeto, com suas funcionalidades, parâmetros e respostas na planilha APT adaptada. Em seguida, o mesmo projeto foi cadastrado na ferramenta, com as mesmas características de funcionalidade, parâmetros e respostas. Os resultados apresentados pela planilha e pela ferramenta foram iguais para os quatro projetos testados como ilustram as Figuras 5 e 6.

ESTIMATIVA			
Projetos	Funções	Estimativas	Sair
<b>Estimativa de Esforço para Teste de Software 20/05/2010</b>			
Projeto:	P. Planilha 01		
Pontos de Função:	72		
Total de horas:	85,69		

<b>NÚMERO TOTAL DE HORAS DE TESTE(THT) *</b>
85,69

Figura 5. Estimativas com a planilha (esquerda) e ferramenta (direita)

ESTIMATIVA			
Projetos	Funções	Estimativas	Sair
<b>Estimativa de Esforço para Teste de Software 20/05/2010</b>			
Projeto:	Projeto 02		
Pontos de Função:	18		
Total de horas:	25,77		

<b>NÚMERO TOTAL DE HORAS DE TESTE(THT) *</b>
25,77

Figura 6. Estimativas com a planilha (esquerda) e ferramenta (direita)

Pode-se observar durante os testes que, no Projeto 01, foi necessário consultar as tabelas de parâmetros várias vezes. No Projeto 02, ainda sim foram consultadas algumas tabelas, mas o usuário já havia memorizado alguns valores. Para os Projetos 03 e 04, a planilha praticamente foi preenchida sem a consulta às tabelas, reduzindo o tempo total de estimativa com a planilha nestes projetos.

A Tabela 1 exibe um comparativo do tempo necessário para preencher todas as informações de um projeto, até se obter a estimativa de esforço para os testes.

**Tabela 1. Comparativo de tempo gasto para se obter estimativas de testes**

	<b>Número de funcionalidades</b>	<b>Tempo gasto usando a planilha</b>	<b>Tempo gasto usando a ferramenta</b>	<b>% Tempo correspondente da planilha</b>
Projeto 01	18	00:23:00	00:09:20	40,57
Projeto 02	4	00:07:50	00:04:15	54,25
Projeto 03	10	00:13:40	00:07:30	54,87
Projeto 04	15	00:14:00	00:08:20	59,52

Com base nestas informações de teste, pode-se verificar que com o uso repetitivo da planilha, diminui-se a dependência de consultas às tabelas de contagem. Entretanto, de qualquer forma, percebe-se que o tempo demandado pela planilha ainda sim é maior que o tempo demandado pela ferramenta para o cadastro das funcionalidades e geração das estimativas.

## **5.2 Resultados Projetos em Andamento**

Após os primeiros resultados bem sucedidos de aplicação da técnica APT, foi decidido utilizá-la em uma quantidade maior de projetos. Nesta seção, são apresentados os resultados do uso da técnica através da ferramenta APTool em projetos ainda em andamento.

Como o esforço realizado total final dos projetos não está disponível, buscou-se uma métrica que pudesse ser utilizada para verificar se o resultado alcançado pela técnica é consistente. Foi utilizada como tal métrica a quantidade de casos de teste. Sendo assim, foi objetivo deste estudo verificar se, para cada projeto, a quantidade de casos de testes previstos e o tempo médio de escrita e execução para cada um deles era consistente com os respectivos tamanhos funcionais.

Nesta seção, é apresentada uma comparação entre a técnica de Análise de Pontos de Testes e dados estatísticos de esforço de teste de acordo com o tamanho funcional de um software, segundo a análise de pontos de função. Segundo [3], existe uma relação entre a quantidade de casos de testes e o tamanho funcional de um software. Isso é explicado pela natureza complexa dos sistemas e a quantidade de interrelacionamentos entre as diversas funções internas deste mesmo sistema. O crescimento desses interrelacionamentos é exponencial à medida que o software cresce.

Segundo [3], um sistema deveria ter aproximadamente a quantidade de pontos de função elevada a 1,2 casos de testes. Desta forma, tomando como base, por exemplo, um sistema de 200 pontos de função, a quantidade de casos de testes que deveriam ser escritos seria de aproximadamente 577 casos de teste.

Com o intuito de verificar se a técnica APT produz estimativas em horas consistentes em relação a quantidade de casos de testes, foi realizado um estudo em uma empresa de grande porte do estado de Minas Gerais. Este piloto consistiu em realizar a contagem de pontos de função de alguns sistemas, a estimativa de Análise de Pontos de Testes, o cálculo da quantidade de casos de testes segundo a métrica [3] e o cálculo do esforço médio em horas necessário para escrever e testar cada caso de teste.

A técnica APT foi aplicada em 12 projetos em andamento nesta organização. A Tabela 2 apresenta os valores obtidos pela contagem de pontos de função, pela estimativa APT, a quantidade esperada de casos de testes e o tempo estimado para cada caso de teste. Os nomes dos sistemas indicados são apenas uma referência aos mesmos e não refletem seus verdadeiros nomes.



Tabela 02 – Estimativa de tempo por casos de testes

ID	Sistema	Pontos de Função Medidos (A)	APT Calculado (B)	Qtde Casos de Teste (C = A <sup>1,2</sup> )	Tempo por Caso de Teste (D=(B/C)*60)
1	SRGNC1	760	1906,60	2864,02	39,94
2	SRGNC2	886	2106,90	3442,87	36,72
3	SACVC	88	116,84	215,47	32,54
4	SACN	120	165,81	312,62	31,82
5	SCORG	177	362,03	498,39	43,58
6	SCCO	123	215,88	322,02	40,22
7	SCPET	319	625,39	1010,53	37,13
8	SGPRD	208	326,24	604,89	32,36
9	SINV	267	415,82	816,23	30,57
10	SOUV	93	180,27	230,24	46,98
11	SGPG	128	223,88	337,79	39,77
12	SEMM	494	1322,01	1707,94	46,44

A Tabela 3 ilustra a média, a mediana, o desvio padrão dos tempos médios estimados para um caso de teste em cada um dos projetos avaliados.

Tabela 03 – Estimativa de tempo por casos de testes

Média	Mediana	Desvio Padrão	Maior Tempo	Menor Tempo
38,17	38,45	5,42	46,98	30,57

Para os cálculos foram consideradas as seguintes regras que estão em conformidade com a técnica APT. Todas as funções dependentes foram classificadas de acordo com a percepção do usuário. Pelo fato da estimativa ter sido realizada em um momento inicial do projeto, ainda não havia sido possível formar uma percepção mais madura de cada função. Neste caso, quando ainda não era possível para o usuário ter uma percepção correta, foi atribuído o valor médio para as seguintes classificações da técnica APT: Importância para o Usuário, Intensidade de Uso, Interface e Complexidade. Em função das restrições de espaço, não se pode detalhar neste texto os parâmetros utilizados para aplicar a técnica APT nestes projetos.

Pelos dados apresentados na Tabela 2, pode-se perceber que, para os projetos analisados, os valores médios estimados por caso de teste, convergiram para um valor próximo a 37 horas. Pode-se observar também que o desvio padrão obtido não foi alto, evidenciando a convergência dos valores para os projetos analisados. Os valores realizados de horas de testes serão obtidos e analisados no término de cada projeto.

## 6 Conclusão

Este trabalho apresentou a ferramenta APTool para automatizar o processo de estimativas usando a técnica Análise de Pontos de Testes. Após o desenvolvimento da ferramenta proposta neste trabalho, foram realizados testes para assegurar que a ferramenta executa os cálculos corretamente de acordo com a técnica. Outro teste realizado foi medir o tempo necessário para se realizar a estimativa final. Este teste mostrou que a ferramenta proporcionou uma estimativa mais ágil que através da planilha. Foi ainda relatado o uso da ferramenta e da técnica em projetos em andamento. Os dados preliminares mostram que a técnica tende a trazer resultados consistentes para estes projetos uma vez que foi obtida uma média bastante próxima de tempo necessário para elaboração dos casos de testes de cada projeto.

Como possíveis trabalhos futuros, podem-se identificar:

- Importar as funcionalidades de uma planilha de contagem de pontos de função para evitar que o usuário tenha que inserir novamente as informações sobre as funções do sistema na ferramenta APTool;
- Geração de relatórios gerenciais na ferramenta APTool;
- Incluir na ferramenta a implementação de outras técnicas de estimativa de esforço para teste de software permitindo diferentes comparações dos resultados;
- Utilizar a ferramenta em treinamentos da técnica;
- Após conclusão dos projetos em andamento, coletar e analisar os dados de esforço realizado comparando com os dados estimados com o uso da ferramenta e da técnica de Análise de Pontos de Testes.

## 7 Referências

1. APF – Ferramenta Análise de Pontos de Função (2010). Disponível em <http://www.baixaki.com.br/download/apf-analise-de-pontos-de-funcao.htm>. Acessado em 20/06/2010
2. Aranha, E.; Borba, P (2003). An Estimation Model for Test Execution Effort. In First International Symposium on Empirical Software Engineering and Measurement.
3. BFPUG (2010). Brazilian Function Points Users Group. <http://www.bfpug.com.br>
4. Karner, G. (1993). Use Case Points - Resource Estimation for Objectory Projects, Objective Systems SF AB, University of Linköping, Suécia.
5. Lopes, F. A.; Nelson, M. A. V. (2008). Análise das técnicas de estimativas de esforço para o processo de teste de software. In: III Encontro Brasileiro de Testes de Software, 2008, Pernambuco.
6. Sommerville, I. (2003) Engenharia de Software – 6a. Edição; Ed. Addison Wesley.
7. Silva, R. R.; Souza, P. P.; Barbosa, M.W.; Rocha, A. (2010). Reduzindo erros de estimativas por meio de treinamentos da técnica de estimativa Análise de Pontos de Testes. XI Congreso de Educación Superior en Ciencias de la Computación. XXIX International Conference of the Chilean Computer Society. Antofagasta. Chile.
8. Souza, P. P.; Barbosa, M. W (2009). Uma análise da adaptação e uso da técnica de estimativa de esforço Análise de Pontos de Testes em um processo de teste de software. In: XXX Congresso Ibero Latino Americano em Métodos Computacionais em Engenharia (CILAMCE), 2009, Búzios - Rio de Janeiro.
9. Souza, P. P.; Barbosa, M. W. (2010). Tailoring the Test Point Analysis Estimation Technique in a Software Testing Process. In: IV Encontro Brasileiro de Testes de Software, 2010, Pernambuco.
10. Vazquez, C.E., Simões, G.S., Albert, R.M. (2003) Análise de Pontos de Função: Medição, Estimativas e Gerenciamento de Projetos de Software - 7ª Edição.
11. Veenendaal, E.V., Dekkers, T. (1999) Test point analysis: a method for test estimation, Project Control for Software Quality, Editors, Rob Kusters, Adrian Cowderoy, Fred Heemstra and Erik van Veenendaal. Shaker Publishing.

12. Bastos, A.; Rios, E.; Cristalli, R.; Moreira, T. (2007). Base de conhecimento em teste de software. 2 ed., São Paulo: Martins.
13. Almeida, E. R. C., Abreu, B. T., Moraes, R. e Martins, E.. Avaliação de um Método para Estimativa de Esforço para Testes baseado em Casos de Uso. In VII Simpósio Brasileiro de Qualidade de Software. Florianópolis. Junho 2008.