



Universidade Federal de Campina Grande  
Laboratório de Sistemas Embarcados e Computação  
Pervasiva (Embedded)

Estudo Dirigido:  
Estimativa de Teste de Software

Rafaela Lacerda de Araújo

Campina Grande - PB  
30 de Junho de 2016

## Estimativa de teste de software

Estimar é uma atividade que pode ser de grande valia no planejamento, gerenciamento e controle dos projetos. Ela deve existir, principalmente, para melhorar o processo de planejamento e controle, permitindo previsões mais certas, montar cronogramas, prever recursos e custos necessários, etc.. Caso não existam estimativas, será mais difícil acompanhar o andamento do projeto, não sendo possível fazer um acompanhamento efetivo a partir do cronograma.

Estimativas são consideradas complexas de se realizar em função de diversos fatores internos e externos que podem impactar diretamente no seu resultado. Para definir uma estimativa próxima da realidade do projeto, dois itens em especial devem ser vistos: o custo e o prazo. Além disso, para garantir a execução dos testes é necessário que haja tanto um planejamento quanto um gerenciamento dos mesmos. Para isso é importante que seja definida uma estimativa mais próxima do tempo realmente necessário para realização dos testes.

Uma boa técnica de medição e estimativa deve sempre levar em consideração o ambiente onde será utilizada. Entretanto, é válido ressaltar que estimar o esforço dos testes não é uma tarefa simples, e diversos fatores como recursos humanos, técnicos, políticos e ambientais podem interferir no seu resultado.

Porém, devido a complexidade dessa tarefa, várias vezes ela é baseada em “achismos”, gerando resultados que podem conflituosos. Para preencher essa lacuna, atualmente existem diversas técnicas que fornecem métricas que permitam atender com uma margem menor de erro às necessidades do projeto. Algumas dessas principais técnicas, são:

### ● Análise de Pontos de Função (APF):

É uma técnica de medição utilizada no ciclo de desenvolvimento de software que tem como objetivo definir o tamanho do sistema, utilizando como métrica a análise dos pontos por função levantada nos requisitos do projeto.

Essa é a técnica de estimativa mais utilizada na área de desenvolvimento de software. Através do Ponto de Função mede-se o tamanho do software pela quantificação de suas funcionalidades externas, baseadas no projeto lógico ou a partir do modelo de dados.

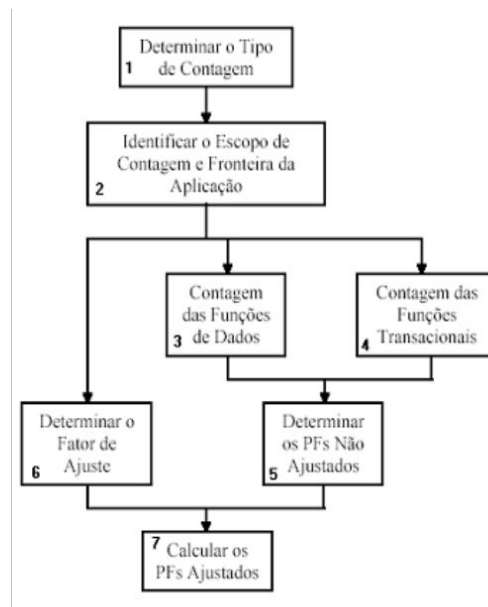
A análise de pontos por função possibilita além de medir o tamanho do sistema no que se refere às funcionalidades disponibilizadas ao usuário, estimar seu tamanho em qualquer fase do ciclo de vida (mesmo que os requisitos ainda não tenham sido detalhados).

As organizações que utilizam a Análise de Pontos por Função podem aplicá-la como:

- Uma ferramenta que permite controlar o tamanho de pacotes de software que foram adquiridos, através de todos os pontos por função envolvidos no projeto;
- Uma ferramenta para determinar o tamanho de pacotes de software adquiridos, através da contagem de todos os Pontos por Função incluídos no pacote;

- Uma ferramenta para apoiar a análise da qualidade e da produtividade;
- Um mecanismo para estimar custos e recursos envolvidos em projetos de desenvolvimento e manutenção de software;
- Um fator de normalização para comparação de software.

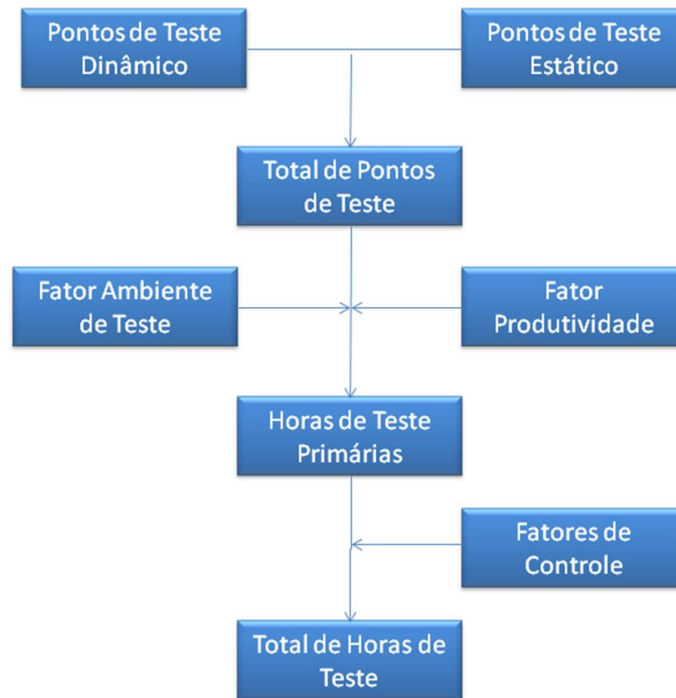
A imagem a seguir mostra as etapas que precisam ser seguidas para diagnosticar a quantidade de pontos de função, os pontos de teste estáticos e dinâmicos.



### ● Análise de Ponto de Teste (APT):

A APT é uma técnica de estimativa de esforço específica para o Teste de Software, mas ainda é pouco difundida no Brasil. Ela se baseia na análise do tamanho do software a ser testado considerando as informações coletadas com a equipe de desenvolvimento. O tamanho do sistema em pontos de função é a base do cálculo inicial utilizada nessa estimativa.

A imagem a seguir mostra as etapas que precisam ser cumpridas para calcular, primeiramente, o tamanho do sistema em pontos de teste (medida de tamanho) e, posteriormente, o número de horas necessárias para a execução dos testes (esforço).



O número de pontos de teste necessários para os testes dinâmicos é calculado para cada função com base no número de pontos de função atribuídos à função, os fatores funções dependentes (complexidade, interfaces, uniformidade, importância do usuário e intensidade de uso) e a qualidade dos requisitos relacionados ou a estratégia de teste para as características de qualidade dinâmica. A soma destes pontos de teste atribuídos a cada função é o número de pontos de teste dinâmico.

O número de pontos de teste estático para medir as características de qualidade é calculado com base no número total de pontos de função para o sistema e a qualidade dos requisitos ou estratégia de teste para as características de qualidade estáticas. Isto resulta no número de pontos de teste estático.

O total de pontos de teste é a soma dos pontos de teste dinâmico e estático. O número de horas de teste primárias pode ser calculado multiplicando o número total de pontos de teste pelos fatores ambiente de teste e produtividade. As horas de teste primárias representam o volume de trabalho envolvido nas atividades de teste primárias, como por exemplo, o tempo necessário para as fases de Preparação, Especificação, Execução e Conclusão.

Finalmente, o total de horas de teste é obtido adicionando subsídio para as atividades de teste secundárias (Planejamento e Controle) ao total de horas primárias. O tamanho deste subsídio, que representa o volume de trabalho envolvido nas atividades de gerenciamento, dependem do tamanho do time de teste e da disponibilidade de ferramentas de gerenciamento. O total de horas de teste é uma estimativa requerida para todas as atividades de teste, excluindo a elaboração do plano de teste.

Fazendo uma correlação entre o processo de estimativa de testes de software e métodos ágeis, vale a pena destacar a técnica conhecida como Planning Poker, que consiste em obter a estimativa por meio de um jogo de

cartas, que deve permitir que todos os membros da equipe de desenvolvimento participem colocando a sua visão de complexidade, levando em consideração o fator tempo e esforço para pontuar um cartão e após juntos chegar a um denominador comum na equipe através de consenso.

## Referências:

Métricas para estimativa de esforço em projetos de teste de software. Link < <http://pt.slideshare.net/samantacilia/mtricas-para-estimativa-de-esforo-em-projetos-de-teste-de-software> >

Estimativa de Esforço em Teste de Software: Modelos, Fatores e Incertezas. Link<

<http://docplayer.com.br/3913009-Estimativa-de-esforco-em-teste-de-software-modelos-fatores-e-incertezas.html> >

Estimativa x Teste de Software. Revista Java Magazine 92. Link < <http://www.devmedia.com.br/estimativa-x-teste-de-software-revista-java-magazine-92/21385> >