

AValiação DE ESTIMATIVA DE ESFORÇO PARA O DESENVOLVIMENTO DE APLICAÇÕES COM SISTEMAS SUPERVISÓRIOS

Renato Fernandez¹
Valesca Alves Corrêa²

Resumo: A determinação do esforço necessário para o desenvolvimento de um projeto é o fator chave para sua aprovação e sucesso. O esforço contempla não somente o tempo e o custo exigido para o desenvolvimento ou manutenção de projetos de software, mas também a quantidade de pessoas e horas necessárias que cada uma deve dedicar ao projeto. Estimativas eficientes permitem a verificação da viabilidade do projeto, a elaboração de propostas técnicas e comerciais, a confecção de planos e cronogramas detalhados, e o acompanhamento efetivo do projeto. As medidas de riscos têm papéis importantes no processo de desenvolvimento de um projeto. Dessa forma, as estimativas se constituem em uma das principais atividades de planejamento para o desenvolvimento de um projeto de software. A Análise de Pontos de Função (APF) é uma técnica de métrica de projetos que viabiliza um planejamento mais realista dos recursos necessários. Baseado em um levantamento prévio do escopo e da estrutura da aplicação, ela permite uma estimativa bastante confiável, dando uma visão prévia dos prazos, custos e pessoal necessários ao desenvolvimento do projeto. Este estudo tem como objetivo a aplicação da técnica do modelo algorítmico de Análise de Pontos de Função (APF) para determinar o esforço necessário no desenvolvimento de sistemas supervisórios utilizando o aplicativo LabVIEW, onde diversos parâmetros serão aplicados visando quantificar a estimativa de esforços em diversos níveis de aplicação principalmente detectando os elementos de maior complexidade que influenciam o projeto. Como resultados espera-se obter o principal fator na mensuração de riscos e fornecer ainda informações significativas para suporte à tomada de decisões que tornem o trabalho mais claro em termos de tempo, esforço e custo, respeitados suas prioridades.

Palavras Chave: Análise de Pontos de Função, Sistemas Supervisórios, LabVIEW.

¹ Mestrando em Engenharia Mecânica pela Universidade de Taubaté e Professor do Instituto Federal de Educação, Ciência e Tecnologia de São Paulo (renato@cefetsp.br, renato_fernandez@hotmail.com)

² Professora Orientadora da Faculdade de Engenharia Mecânica da Universidade de Taubaté (valescacorrea@unitau.com.br, alvescorreavalesca@gmail.com)

Abstract: The determination of the effort required to develop a project is the key factor in its approval and success. The effort includes not only the time and cost required for the development or maintenance of software projects, but also the number of people and hours required that each one must devote for the project. Estimates efficient for checking the feasibility of the project, preparation of technical and commercial proposals, the preparation of detailed plans and schedules, and effective monitoring of the project. The risk measures have important roles in the development process of a project. Thus, the estimates are on a major planning activities for the development of a software project. The Function Point Analysis (FPA) is a technique that enables metric projects planned in a more realistic resource requirements. Based on a preliminary survey of the scope and structure of the application, it allows a fairly reliable estimate, giving a preview of deadlines, costs and personnel necessary to develop the project. This study aims at applying the technique of algorithmic model of Function Point Analysis (FPA) to determine the effort required in developing supervisory systems using the LabVIEW application, where several parameters are applied to quantify the estimation of effort at various levels application mainly detecting the most complex elements that influence the project. As results we expect to get a major factor in measuring risk and still provide meaningful information to support decision that makes the work more clear in terms of time, effort and cost, respecting their priorities.

Keywords: Function Point Analysis, Supervisory Systems, LabVIEW.

1. Introdução

Em muitos sistemas industriais ainda existem equipamentos que deveriam estar totalmente automatizados, mas que por razões da própria dinâmica da produção acabam sendo usados de forma manual. Tais situações levam a perda de produtividade que é erroneamente compensada pela continuidade ou não interrupção do processo produtivo.

Métricas de software são padrões quantitativos de medidas de vários aspectos de um projeto ou produto de software, e se constitui em uma poderosa ferramenta gerencial, contribuindo para a elaboração de estimativas de prazo e custo mais precisas e para o estabelecimento de metas plausíveis, facilitando assim o processo de tomada de decisões e a subsequente obtenção de medidas de produtividade e qualidade.

Meller (2002), aplicou diversas metodologias para avaliação da estimativa de custos de softwares visando verificar, no estudo de casos, possíveis relações entre as técnicas aplicadas e os modelos estudados relacionando a partir desta análise os modelos mais adequados para cada tipo

de software avaliado.

Drach (2005), realizou uma análise das características computacionais específicas da plataforma Web de forma a permitir que desenvolvedores e gerentes de projeto avaliassem o grau de adequação da Análise de Pontos de Função (APF) a este tipo de ambiente e sua contribuição para extração de requisitos e estimativa de esforço e concluiu que o resultado obtido demonstrou que a contagem de Pontos de Função (PF) com esta metodologia se mostrou suficiente para contabilizar a maioria das funções presentes em aplicações em Web.

Lemos (2006), teve como objetivo o desenvolvimento de uma ferramenta capaz de manter uma base de dados de projetos de software utilizando a metodologia APF como medida de tamanho de software e concluiu que o uso desta ferramenta é viável na realização de inferências estatísticas a fim de obter novos modelos de estimativas de desenvolvimento de projetos de software.

Este trabalho visa qualificar as estimativas de esforços nas diversas ferramentas disponíveis no software LabVIEW se baseando em sistemas supervisórios já desenvolvidos na área de automação industrial utilizando a metodologia APF.

2. Análise de pontos de função

A APF é uma técnica de medição das funcionalidades fornecidas por uma aplicação do ponto de vista de seu usuário. Ponto de função é a unidade de medida desta técnica que tem por objetivo tornar a medição independente da tecnologia utilizada para a construção da aplicação, ou seja, a APF busca medir o que a aplicação faz, e não como ela será construída (VAZQUES, SIMÕES e ALBERT, 2010).

Portanto o processo de medição (também chamado contagem de pontos de função) é baseado em uma avaliação padronizada dos requisitos lógicos do usuário. Este procedimento padrão está descrito pelo grupo de usuários da International Function Point Users Group (IFPUG) em seu Manual de Práticas de Contagem.

As principais técnicas de estimativa de projetos de desenvolvimento de software assumem que o tamanho de um software é um vetor importante para a determinação do esforço para sua construção. Logo, saber o seu tamanho é um dos primeiros passos do processo de estimativa de esforço, prazo e custo.

Assim sendo, é importante destacar que PF não medem diretamente

de software avaliado.

Drach (2005), realizou uma análise das características computacionais específicas da plataforma Web de forma a permitir que desenvolvedores e gerentes de projeto avaliassem o grau de adequação da Análise de Pontos de Função (APF) a este tipo de ambiente e sua contribuição para extração de requisitos e estimativa de esforço e concluiu que o resultado obtido demonstrou que a contagem de Pontos de Função (PF) com esta metodologia se mostrou suficiente para contabilizar a maioria das funções presentes em aplicações em Web.

Lemos (2006), teve como objetivo o desenvolvimento de uma ferramenta capaz de manter uma base de dados de projetos de software utilizando a metodologia APF como medida de tamanho de software e concluiu que o uso desta ferramenta é viável na realização de inferências estatísticas a fim de obter novos modelos de estimativas de desenvolvimento de projetos de software.

Este trabalho visa qualificar as estimativas de esforços nas diversas ferramentas disponíveis no software LabVIEW se baseando em sistemas supervisórios já desenvolvidos na área de automação industrial utilizando a metodologia APF.

2. Análise de pontos de função

A APF é uma técnica de medição das funcionalidades fornecidas por uma aplicação do ponto de vista de seu usuário. Ponto de função é a unidade de medida desta técnica que tem por objetivo tornar a medição independente da tecnologia utilizada para a construção da aplicação, ou seja, a APF busca medir o que a aplicação faz, e não como ela será construída (VAZQUES, SIMÕES e ALBERT, 2010).

Portanto o processo de medição (também chamado contagem de pontos de função) é baseado em uma avaliação padronizada dos requisitos lógicos do usuário. Este procedimento padrão está descrito pelo grupo de usuários da International Function Point Users Group (IFPUG) em seu Manual de Práticas de Contagem.

As principais técnicas de estimativa de projetos de desenvolvimento de software assumem que o tamanho de um software é um vetor importante para a determinação do esforço para sua construção. Logo, saber o seu tamanho é um dos primeiros passos do processo de estimativa de esforço, prazo e custo.

Assim sendo, é importante destacar que PF não medem diretamente

–A complexidade de uma saída externa é calculada a partir da quantidade de arquivos lógicos referenciados (um arquivo lido pela lógica de processamento da saída externa) e da quantidade de dados elementares referenciados (um único campo não recursivo, identificado pelo usuário, que aparece em uma saída externa).

- Consulta Externa (*External Inquiry*): Representa um processo elementar que envia dados ou informações de controle para fora da fronteira da aplicação. O objetivo principal é apresentar a informação para o usuário por intermédio da recuperação de dados. O processamento lógico não contém fórmulas matemáticas, nem cálculos e não cria dados derivados. Pode-se dizer também que é uma atividade que, através de uma requisição on-line de dados gera uma resposta imediata.

A complexidade de uma consulta externa é calculada a partir da quantidade de arquivos lógicos referenciados (um arquivo lido quando a consulta externa é processada) e da quantidade de dados elementares referenciados (um campo não recursivo, identificado pelo usuário, que aparece em uma consulta externa).

Deve-se utilizar apenas a maior complexidade encontrada entre as partes de entrada e de saída.

As funções contribuem para o cálculo de PF com base na quantidade (número de funções) e na complexidade funcional relativa atribuída a cada uma delas.

Determina-se o número de PF de uma aplicação em três etapas de avaliação:

- PF não-ajustados: Refletem as funções específicas e mensuráveis do negócio, providas ao usuário pela aplicação;
- Fator de Ajuste: Representam a funcionalidade geral provida ao usuário pela aplicação;
- PF Ajustados: Refletem a aplicação do Fator de Ajuste sobre o resultado apurado na primeira etapa.

2.1 Cálculo dos PF Não-Ajustados ou PF Brutos

Uma função específica do usuário em uma aplicação é avaliada em termos do que é fornecido pela aplicação e não como é fornecido. *Somente componentes solicitados e visíveis ao usuário são contados.*

Cada função, através de critérios próprios, deverá ser classificada segundo a sua complexidade funcional relativa, em: Simples, Média ou

Complexa.

A cada função será atribuído um número de pontos, segundo seu tipo e complexidade funcional relativa conforme mostra a Tabela 1 (ANDRADE, 2004):

Tabela 1. Dados para cálculo dos PF brutos

Tipo de Função	Complexidade Funcional Relativa		
	Simples	Média	Complexa
Arquivo Lógico Interno (ILF)	7	10	15
Arquivo de Interface Externa (EIF)	5	7	10
Entrada Externa (EI)	3	4	6
Saída Externa (EO)	4	5	7
Consulta Externa (EI)	3	4	6

2.2 Cálculo do Fator de Ajuste

Segundo Mecenas (2009), o valor do Fator de Ajuste é calculado a partir de 14 características gerais do sistema, que permitem uma avaliação geral da funcionalidade da aplicação.

As características gerais de um sistema são:

- Comunicação de Dados: Quando são utilizados recursos de Comunicação de Dados para o envio ou recebimento de dados e informações de controle, utilizados pela aplicação;
- Processamento Distribuído: Quando a aplicação prevê a distribuição de dados ou de processamento entre várias CPUs de instalação;
- Performance: Esta característica identifica os objetivos de performance da aplicação, estabelecidos e aprovados pelo usuário, que influenciaram (ou irão influenciar) o desenho, desenvolvimento, implantação e suporte da aplicação;
- Utilização do Equipamento: Representa a necessidade de se fazer considerações especiais na arquitetura dos sistemas para que a configuração do equipamento não sofra degradação;
- Volume de Transações: Avalia o impacto no desenho da aplicação, do volume de transações previsto para ela;
- Entrada de Dados "on-line": Avalia o volume de transações que são entradas de dados interativas;
- Eficiência do Usuário Final: Analisa as funções "on-line" desenhadas e disponibilizadas voltadas para a eficiência do usuário final;
- Atualização "on-line": Verifica o volume de arquivos lógicos internos que sofrem manutenção "on-line" e o impacto do processo de recuperação de seus dados;
- Processamento Complexo: Considera o impacto, sobre o desenho da

aplicação, causado pelo tipo de complexidade do processamento;

- Reutilização de Código: Avalia se a aplicação e seu código foram especificamente projetados e desenvolvidos para serem reutilizados em outras aplicações;
- Facilidades de Implantação: Considera o esforço despendido para o atendimento dos requerimentos de conversão de dados para a implantação da aplicação;
- Facilidade Operacional: Avalia o desenho da aplicação quanto aos requisitos estabelecidos para inicialização, "backup" e recuperação voltados à minimização da intervenção manual do operador;
- Múltiplos Locais: Quando a aplicação for especificamente projetada e desenvolvida para ser instalada em múltiplos locais ou para múltiplas organizações;
- Facilidades de Mudanças: Quando os requisitos da aplicação prevêm o projeto e desenvolvimento de mecanismos que facilitem mudanças operacionais, tais como: capacidade de emissão de relatórios genéricos, de consultas flexíveis ou de alterações nos dados de controle do negócio (parametrização)

Segundo Hazan (2009), a cada característica será atribuído um peso variando de 0 (zero) a 5 (cinco), de acordo com o nível influência na aplicação, observando-se os critérios estabelecidos para cada característica, representando: 0 (zero) - Nenhuma influência, 1 (um) - Influência Mínima, 2 (dois) - Influência Moderada, 3 (três) - Influência Média, 4 (quatro) - Influência Significativa e 5 (cinco) - Grande Influência.

O Nível de Influência Geral é obtido pelo somatório do nível de influência de cada característica e o Fator de Ajuste é obtido pela expressão 1

$$\text{Fator de Ajuste} = 0,65 + (\text{Nível de Influência Geral} * 0,01)$$

Expressão 1. Fator de Ajuste

O fator de ajuste é aplicado sobre os PF Brutos para permitir o cálculo dos PF Ajustados. Esse valor pode variar de 0,65 até 1,35, uma vez que o fator de ajuste, quando aplicado aos PF não ajustados, pode produzir uma variação de mais ou menos 35% e cada ponto atribuído ao nível de influência afeta o resultado final em 1%.

2.3 Cálculo dos PF Ajustados

O total de PF da aplicação será encontrado através da multiplicação do número de PF Não Ajustados (PFB) pelo Fator de Ajuste (FA).

O processo de estimativa de projetos de software envolve quatro atividades, sendo necessário estimar o tamanho do produto a ser desenvolvido, o esforço a ser empregado para sua implementação e a duração do projeto e o custo à organização.

Após analisar os requisitos para garantir a qualidade do produto e o tamanho do projeto de software, o próximo passo é calcular o esforço necessário e então derivar as estimativas de prazo e custo com base nas estimativas de tamanho. Desta forma, a partir do cálculo do tamanho do projeto, é possível calcular todas as outras estimativas, de forma a identificar as necessidades de recursos financeiros e de pessoal.

3. Sistemas supervisórios

Atualmente, os sistemas de automação industrial utilizam tecnologias de computação e comunicação para automatizar a monitoração e controle dos processos industriais, efetuando coleta de dados em ambientes complexos, eventualmente dispersos geograficamente, e a respectiva apresentação de modo amigável para o operador, com recursos gráficos elaborados (interfaces homem-máquina) e conteúdo multimídia.

Os Sistemas Supervisórios permitem que sejam monitoradas e rastreadas informações de um processo produtivo ou instalação física. Tais informações são coletadas através de equipamentos de aquisição de dados e, em seguida, manipulados, analisados, armazenados e, posteriormente, apresentados ao usuário. Estes sistemas também são chamados de SCADA (*Supervisory Control and Data Acquisition*).

Os componentes físicos de um sistema de supervisão podem ser resumidos, de forma simplificada, em: sensores e atuadores, rede de comunicação, estações remotas (aquisição/controle) e de monitoração central (sistema computacional SCADA).

O processo de controle e aquisição de dados se inicia nas estações remotas, PLCs (*Programmable Logic Controllers*) e RTUs (*Remote Terminal Units*), com a leitura dos valores atuais dos dispositivos que a ele estão associados e seu respectivo controle. Os PLCs e RTUs são unidades computacionais específicas, utilizadas nas instalações fabris (ou qualquer outro tipo de instalação que se deseje monitorar) para a funcionalidade de ler entradas, realizar cálculos ou controles, e atualizar saídas. A diferença entre os PLCs e as RTUs é que os primeiros possuem mais flexibilidade na linguagem

de programação e controle de entradas e saídas, enquanto as RTUs possuem uma arquitetura mais distribuída entre sua unidade de processamento central e os cartões de entradas e saídas, com maior precisão e seqüenciamento de eventos.

A rede de comunicação é a plataforma por onde as informações fluem dos PLCs/RTUs para o sistema SCADA e, levando em consideração os requisitos do sistema e a distância a cobrir, pode ser implementada através de cabos Ethernet, fibras ópticas, linhas dial-up, linhas dedicadas, rádio modems, etc.

As estações de monitoração central são as unidades principais dos sistemas SCADA, sendo responsáveis por recolher a informação gerada pelas estações remotas e agir em conformidade com os eventos detectados, podendo ser centralizadas num único computador ou distribuídas por uma rede de computadores, de modo a permitir o compartilhamento das informações coletadas (DANEELS e SALTER, 2000).

4. Software Labview

O software LabVIEW é um aplicativo desenvolvido pela National Instruments baseado em linguagem G (linguagem de programação gráfica ou visual) que emprega ícones ao invés de textos para criar aplicações. Este tipo de programação é baseada no fluxo de dados que definem a execução desses dados. (LabVIEW User Manual, 1998)

O LabVIEW permite criar aplicações de teste e medição, aquisição de dados, controle de instrumento, registro de dados, análise de medição e geração de relatório, além de aplicações executáveis e bibliotecas compartilhadas

Os programas em LabVIEW são chamados de instrumentos virtuais (VIs - *Virtual Instruments*). Os VIs contêm três componentes principais: o painel frontal, o diagrama de bloco e o painel de ícones e conectores. A Figura 1 mostra uma representação do ambiente computacional do software LabVIEW.

As vantagens de utilização deste ambiente computacional se resumem na diversidade de drivers e suporte para acesso a diferentes periféricos/instrumentos e hardware, diversas bibliotecas de funções como: Data Acquisition – Aquisição de dados, entradas e saídas analógicas e digitais; Signal generation – Geração de sinais periódicos e outros; Mathematics – Funções e instruções matemáticas; Statistics – Funções e instruções estatísticas; Signal conditioning – Acondicionamento de sinal; Analysis – Análise (VELOSA, 2009).

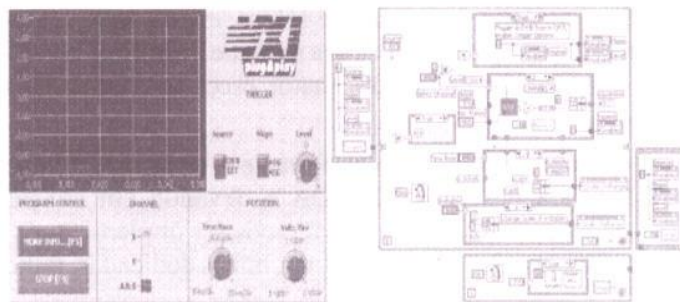


Figura 1. Painel Frontal e Diagrama de Blocos do software LabVIEW

5. Resultados

Com o objetivo de ilustrar como se utiliza a metodologia APF para se estimar o tamanho de um software, será considerada uma aplicação demonstrada por Lopes (2007), referente ao desenvolvimento de um Sistema Supervisório utilizando o Software LabVIEW para Aquisição, Tratamento, Visualização e Armazenamento de Sinais de um Mini Sistema de Potência, onde foram avaliadas as seguintes interfaces: Medições Simultâneas, Estado do Mini Sistema e Instrumentação Virtual, conforme Diagrama de Bloco da Figura 1.

A seguir serão analisadas nas Tabelas 2, 3 e 4 as entradas e saídas para os três casos propostos:

Tabela 2. Identificação dos componentes de entrada e saída para instrumentação virtual

Instrumentação Virtual	
Entradas	1 menu do mini sistema
	1 diagrama de fasores
	1 osciloscópio virtual
	1 sincronoscópio virtual
Saídas	1 sinal de tensão no início da Linha de Transmissão
	1 sinal de tensão no final da Linha de Transmissão
	1 sinal de corrente
	1 forma de onda do sinal de tensão
	1 diagrama de fasores do sinal de tensão
	1 valor de defasagem entre o sinal de corrente e o sinal de tensão
	1 número de ciclos de tensão
	1 número de ciclos de corrente
	1 fator multiplicativo para sinal de corrente
	1 filtro digital
	1 sequência de fase diferente
	1 sequência de fase igual
	3 fasores de tensão na saída da Linha de Transmissão
	3 fasores de tensão na concessionária

Tabela 3. Identificação dos componentes de entrada e saída para medições simultâneas

Medições Simultâneas	
Entradas	1000 amostras
Saídas	3 valores de tensão na saída do gerador
	3 valores de corrente eficaz na Linha de Transmissão
	3 valores de tensão no fim da Linha de Transmissão
	1 frequência no mini sistema (em Hertz) (cálculo matemático)
	1 valor de potencia ativa (cálculo matemático)
	1 valor de potencia reativa (cálculo matemático)
	1 valor de potencia aparente (cálculo matemático)
	1 fator de potência (cálculo matemático)

Tabela 4. Identificação dos componentes de entrada e saída para estado do mini sistema

Estado do Mini Sistema	
Entradas	4 chaves
Saídas	16 imagens para representação dos estados

Uma vez definidas as características de cada interface proposta de supervisão, a Tabela 5 demonstra a identificação da complexidade para a entrada de dados através da tela inicial do mini sistema, apresentada no Diagrama de Bloco da Figura 1:

Tabela 5. Identificação da complexidade para entrada externa (tela inicial)

Registros Lógicos Referenciados	Dados Elementares Referenciados		
	01 até 04	05 até 15	16 ou mais
00 ou 01	Simple	Simple	Média
02	Simple	Média	Complexa
03 ou mais	Média	Complexa	Complexa

A Tabela 6 demonstra as quantidades de Entradas Externas e as respectivas Complexidades encontradas para cada interface proposta:

Tabela 6. Valores da complexidade para entrada externa para as três interfaces propostas (tela inicial)

Interface	Quantidade de Entradas Externas (Tela Inicial)	Complexidade
Medições Simultâneas	1	Simple
Estado do Mini Sistema	4	Simple
Instrumentação Virtual	4	Simple

De forma análoga a Tabela 7 mostra os valores de referência para o cálculo da complexidade em relação à saída externa.

Tabela 7. Identificação da complexidade para saída externa (cálculos matemáticos)

Registros lógicos referenciados	Dados Elementares Referenciados		
	01 até 05	06 até 19	20 ou mais
00 ou 01	Simple	Simple	Média
02 ou 03	Simple	Média	Complexa
04 ou mais	Média	Complexa	Complexa

A Tabela 8 demonstra as quantidades de Saídas Externas e as respectivas Complexidades calculadas para cada interface.

Tabela 8. Valores da complexidade para saída externa (cálculos matemáticos)

Interface	Quantidade de Saídas Externas	Complexidade
Medições Simultâneas	5	Simple
Estado do Mini Sistema	0	Simple
Instrumentação Virtual	0	Simple

A seguir serão considerados na Tabela 9 os valores de referência para a identificação da complexidade para consulta externa:

Tabela 9. Identificação da complexidade para consulta externa - parte de saída

Registros Referenciados	Lógicos	Dados Elementares Referenciados		
		01 até 05	06 até 19	20 ou mais
00 ou 01		Simple	Simple	Média
02 ou 03		Simple	Média	Complexa
04 ou mais		Média	Complexa	Complexa

A Tabela 10 demonstra as quantidades de Consultas Externas e as respectivas Complexidades em relação às saídas para cada interface:

Tabela 10. Valores da complexidade para consulta externa - parte de saída

Interface	Quantidade de Consultas Externas	Complexidade
Medições Simultâneas	09	Simple
Estado do Mini Sistema	16	Simple
Instrumentação Virtual	18	Simple

De acordo com os valores de referência da Tabela 1, serão demonstrados a seguir na Tabela 11, os valores calculados de complexidade funcional para cada interface:

Tabela 11. Valores de complexidade funcional

Interface	Entrada Externa	Saída Externa	Consulta Externa	Complexidade Funcional Relativa
Medições Simultâneas	Simple / 3	Simple / 4	Simple / 3	10
Estado do Mini Sistema	Simple / 3	Simple / 4	Simple / 3	10
Instrumentação Virtual	Simple / 3	Simple / 4	Simple / 3	10

Apesar da existência de 14 características gerais do sistema, conforme citado anteriormente no item Cálculo do Fator de Ajuste, a seguir serão demonstradas na Tabela 12 as características gerais inerentes ao sistema de Aquisição, Tratamento, Visualização e Armazenamento de Sinais de um Mini Sistema de Potência e seus respectivos níveis de influência, uma vez que as características referentes a Processamento Distribuído, Utilização de Equipamento, Entrada de Dados “On-Line”, Atualização “On-Line”, Reutilização de Código, Múltiplos Locais e Facilidades de Mudanças não fazem parte do escopo dos sistemas avaliados. A soma do Grau de Influência das características, citadas abaixo, afetam o esforço necessário para desenvolver a aplicação.

Como cada característica pode ter um Grau de Influência variando de 0 a 5, o Grau de Influência Total (GIT) pode variar de 0 a 70 ($14 * 0$ a $14 * 5$).

Tabela 12. Características gerais do sistema e níveis de influência

Características	Nível de Influência					
	Nenhuma[0]	Mínima[1]	Moderada[2]	Média[3]	Significativa[4]	Grande[5]
Comunicação de dados					4	
Performance						5
Volume de transações				3		
Eficiência usuário final						5
Processamento complexo				3		
Facilidades de implantação				3		
Facilidade operacional						5
Grau de influência total (GIT)						28

O cálculo do Fator de Ajuste (FAT) será obtido a partir do Grau de Influência Total das características da aplicação (GIT) aplicando a expressão 2:

$$FAT = 1,35 + (0,01 * GIT) = 1,63$$

Expressão 2. Fator de Ajuste

Como o Grau de Influência Total (GIT) varia de 0 a 70, o Fator de Ajuste (FAT) varia de 0,65 a 1,35, uma vez que o fator de ajuste, quando aplicado aos PF não ajustados, pode produzir uma variação de mais ou menos 35% e cada ponto atribuído ao nível de influência afeta o resultado final em 1%. O fator de ajuste é aplicado sobre os PF Brutos para permitir o cálculo dos PF Ajustados.

A Tabela 13 mostra os valores de referência para o cálculo do total de PF brutos:

Tabela 13. Nível de complexidade para cálculo do Total de PF Brutos

Complexidade	Arquivo	Interface	Entrada	Saida	Consulta
Simples	7	5	3	4	3
Média	10	7	4	5	4
Complexa	15	10	6	7	6

Considerando-se os pesos demonstrados na Tabela 13, as quantidades de entradas demonstradas na Tabela 6, as quantidades de saídas demonstradas na Tabela 8 e as quantidades de consultas demonstradas na Tabela 10, os PF Brutos são apresentados na Tabela 14 para cada interface avaliada.

Tabela 14. Total de PF Brutos para cada interface

Interface	nº entradas / peso	nº saídas / peso	nº consultas / peso	PF Brutos
Medições Simultâneas	1 Simples / 03	5 Simples / 20	9 Simples / 27	50
Estado do Mini Sistema	4 Simples / 12	0 Simples / 00	16 Simples / 48	60
Instrumentação Virtual	4 Simples / 12	0 Simples / 00	18 Simples / 54	66

A determinação do Total de PF Ajustados, que reflete o porte da aplicação considerando as características técnicas que afetam o seu projeto e implementação, é obtida através da expressão 3:

$$PFA = PFB * FAT$$

Expressão 3. Pontos de Função Ajustados

Onde:

PFA = Total de PF Ajustados

PFB = Total de PF Brutos

FAT = Fator de Ajuste

A Tabela 15 mostra os valores calculados para cada interface do total de PF ajustados:

Tabela 15. Total de PF Ajustados para cada interface

Interface	PFA
Medições Simultâneas	81,50
Estado do Mini Sistema	97,80
Instrumentação Virtual	107,58

Foi considerado um ajuste de 35% para mais, pois apesar das funções utilizadas serem exclusivas do Software LabVIEW, a quantidade de funções é bastante grande para o desenvolvimento das interfaces, conforme ilustra a Figura 2.

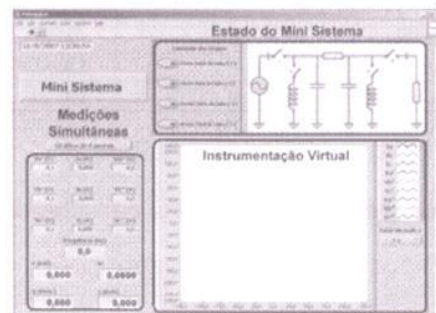


Figura 2. Diagrama de blocos do mini sistema

-A seguir serão estimados os recursos necessários para desenvolver a aplicação, a partir do Total de PF Ajustados e do esforço necessário para construir cada Ponto de Função.

O esforço para construir cada Ponto de Função é medido em Homens.Hora por Ponto de Função, representando o Índice de Produtividade (IPD) da instalação apresentado na Tabela 16, que reflete o IPD em quantidade de Homens.Hora por Ponto de Função em função da tecnologia empregada.

Tabela 16. Índice de produtividade

Tecnologia	IPD
Linguagens de 3a. Geração	12 a 16
Linguagens de 4a. Geração e ambiente relacional	8 a 12
Geradores de aplicação	4 a 8

Para cálculo dos Recursos Estimados em Homens.Hora, utiliza-se a expressão 4:

$$REH = PFA * IPD$$

Expressão 4. Recursos Humanos Hora

Onde:

REH = Recursos estimados em Homens.Hora

PFA = PF Ajustados

IPD = Índice de Produtividade (considerado 4 (quatro) devido a grande quantidade de funções utilizadas do gerador de aplicação)

A seguir serão mostrados na Tabela 17 os valores calculados para cada modelo proposto, considerando-se IPD igual a seis, devido a grande quantidade de funções do software LabVIEW para o desenvolvimento das interfaces avaliadas do mini sistema:

Tabela 17. Recursos Humanos.Hora para cada interface

Interface	REH
Medições Simultâneas	326,00
Estado do Mini Sistema	391,20
Instrumentação Virtual	430,32

Para cálculo dos Recursos Estimados em Homens.Mês, utiliza-se a expressão 5:

$$REM = REH \div 120$$

Expressão 5. Recursos Humanos Mês

Onde:

REM = Recursos estimados em Homens.Mês

REH = Recursos estimados em Homens.Hora

120 = Quantidade. Média de Horas Mensais.

Considera-se que um recurso dedica efetivamente 6 horas por dia, e 20 dias por mês ao projeto.

A Tabela 18 mostra os valores dos Recursos Estimados em Homem.Mês para cada interface:

Tabela 18. Recursos Humanos.Mês para cada interface

Interface	REM
Medições Simultâneas	2,72
Estado do Mini Sistema	3,26
Instrumentação Virtual	3,59

A seguir será determinado o Prazo Ótimo (PZO), ou seja, prazo ideal para desenvolver a aplicação sem acréscimo no total de recursos, conforme a expressão 6:

$$PZO = 2,5 * REM$$

Expressão 6. Prazo Ótimo

Onde:

PZO = Prazo Ótimo em Meses.

REM = Recursos estimados em Homens.Mês.

A Tabela 19 mostra os melhores prazos calculados para cada interface proposta:

Tabela 19. Melhor prazo para cada interface

Interface	PZO
Medições Simultâneas	6,80
Estado do Mini Sistema	8,15
Instrumentação Virtual	8,98

A partir da análise das três interfaces avaliadas (Medições Simultâneas, Estado do Mini Sistema e Instrumentação Virtual) e das regras de identificação de PF, teve-se como resultado, os valores de PF estimados de 81,50, 97,80 e 107,58 respectivamente.

6. Conclusão

Conclui-se que com a aplicação da APF em processos de automação que envolve sistemas supervisórios, será possível aos engenheiros diagnosticar o tempo necessário e os custos na implementação de processos automatizados permitindo um melhor planejamento.

Dessa forma, através dos PF obtidos, nota-se que para se desenvolver o Sistema de Aquisição, Tratamento, Visualização e Armazenamento de Sinais de um Mini Sistema de Potência, contemplando as interfaces de Medições Simultâneas, Estado do Mini Sistema e Instrumentação Virtual, com fator de complexidade funcional 10 em uma escala de 10 a 19, foi necessário um período de 24 meses para desenvolvimento das funções de monitoração de variáveis elétricas, cálculo e análise de potência, energia e fator de potência de um mini sistema de energia em baixa tensão e das interfaces para disponibilização das informações através de diagrama de fasores de sinais de tensão e corrente, forma de onda dos sinais de tensão e corrente no mini sistema com as respectivas defasagens e um sincronoscópio virtual que possibilita a análise dos ajustes de tensão e frequência necessários na operação de paralelismo do mini sistema com a concessionária de energia.

7. Referências

- ANDRADE, Edméia Leonor Pereira de. **Pontos de Casos de Uso e pontos de Função na Gestão de Estimativa de Software Orientado a Objetos**. Dissertação de Mestrado, Universidade Católica de Brasília, 2004.
- DANEELS, Alex; SALTER, Wayne. **What is SCADA? CERN - European Organization for Nuclear Research**, CNL-2000-003, Vol. XXXV, issue no 3.
- DRACH, Marcos Dávid. **Aplicabilidade de Métricas por Pontos de Função em Sistemas Baseados em Web**. Dissertação de Mestrado, Universidade Estadual de Campinas, 2005.
- HAZAN, Claudia. **Análise de Pontos de Função** - Uma aplicação nas estimativas de tamanho de Projetos de Software. Engenharia de Software Magazine. Rio de Janeiro, n. 02, p. 25-31, 2009.
- IFPUG. **Function Point Counting Practices Manual: Release 4.3**. International Function Point Users Group, jan 2010.
- LabVIEW User Manual. National Instruments I Corporate Headquarters, 11500 North Mopac Expressway Austin, Texas, 1998.
- LEMOS, Julio Cesar de. **Desenvolvimento de uma Ferramenta Automatizada para Suporte à Análise de Pontos de Função** Formando uma Baseline de Projetos de Software. Dissertação de Mestrado, Universidade Metodista de Piracicabã, 2006.

LOPES, Vinícius José Santos. **Instrumentação Virtual** Aplicação ao Ensino Experimental de Engenharia Elétrica, Dissertação de Mestrado, Universidade de São Paulo, 2007.

MECENAS, Ivã. **Análise de Pontos de Função**: Estudo Teórico, Crítico e Prático. Rio de Janeiro: Alta Books, 2009

MELLER, Maristela Corrêa. **Modelos para Estimar Custos de Softwares: Estudo Comparativo com Softwares de Pequeno Porte**. Dissertação de Mestrado, Universidade Federal de Santa Catarina, 2002.

VAZQUEZ, Carlos Eduardo, SIMÕES, Guilherme Siqueira, ALBERT, Renato Machado. **Análise de Pontos de Função: Medição, Estimativas e Gerenciamento de Projetos de Software**, São Paulo, Ed. Érica, 2010.

VELOSA, José Eduardo Corrêa. **Controle Automático de um Interferômetro para Monitorização e Caracterização de Sensores Interferométricos**. Dissertação de Mestrado, Universidade da Madeira, Portugal, 2009.