

UNIVERSIDADE FEDERAL DE CAMPINA GRANDE  
CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA  
CURSO DE GRADUAÇÃO EM ENGENHARIA ELÉTRICA

Projeto de Engenharia Elétrica

**Sistema de Supervisão e Controle na Configuração *Stand Alone***

**Aluno**

Hugo Tabosa da Silva

**Orientadora**

Prof. Dr. Maria de Fátima Q. V. Turnell

Campina Grande, Paraíba – Dezembro de 2006

Aluno **Hugo Tabosa da Silva** - 20421141

**Sistema de Supervisão e Controle na Configuração *Stand Alone***

Projeto de Engenharia Elétrica,  
apresentado como requisito para a obtenção  
do título de Graduado Pleno em Engenharia Elétrica

Curso	<b>Graduação em Engenharia Elétrica</b>
Área de Concentração do TCC	<b>Sistemas Supervisórios</b>
Orientador	<b>Prof. Dr. Maria de Fátima Q. V. Turnell</b>

Campina Grande, Paraíba – Dezembro de 2006



Biblioteca Setorial do CDSA. Fevereiro de 2021.

Sumé - PB

FICHA CATALOGRÁFICA ELABORADA PELA BIBLIOTECA CENTRAL DA UFCG

S586s Silva, Hugo Tabosa da  
2006 Sistema de Supervisão e Controle na Configuração Stand Alone / Hugo  
Tabosa da Silva . — Campina Grande, 2006.  
36f.:il.

Referências.

Monografia (Graduação em Engenharia Elétrica) – Universidade Federal  
de Campina Grande, Centro de Engenharia Elétrica e Informática.

Orientadora: Maria de Fátima Q. V. Turnell.

1— Interface Homem-Máquina 2— Parametrização 3— Testes de  
Usabilidade 4— Comunicação I— Título

CDU 004.05

**Hugo Tabosa da Silva**

Sistema de Supervisão e Controle na Configuração *Stand Alone*

TCC Aprovado em 21.12.2006

Banca examinadora

Prof. Dr. **Maria de Fátima Q. V. Turnell**

UFCG/CEEI/UAEE

Prof. Dr. **Edmar Candeia Gurjão**

UFCG/CEEI/UAEE

## **DEDICATÓRIA**

Dedico este trabalho, principalmente, a Deus, que sempre me ajudou e guiou nos momentos mais difíceis; dedico este trabalho aos meus pais Cícero e Conceição que sempre apoiaram e deram força as minhas decisões, com muito carinho, para a realização dessa etapa da vida.

## **AGRADECIMENTOS**

Agradeço em particular a professora Maria de Fátima Q. V. Turnell por toda ajuda e conhecimento que alicerçaram esse trabalho. Agradeço a todos os professores desta universidade que contribuíram para o meu desenvolvimento neste curso.

Agradeço, também, aos que me ajudaram na realização deste trabalho.

## RESUMO

O presente trabalho tem como objetivo apresentar os estudos sobre sistemas supervisórios visando a construção de um simulador para treinamento de operadores de subestação de sistemas elétricos, em situações críticas. O simulador se fundamenta em estudo realizado sobre o erro humano na operação dos sistemas, na empresa CHESF. Nesta empresa o ambiente de supervisão é desenvolvido no ambiente SAGE, no entanto como este sistema não foi disponibilizado para o projeto da interface com o simulador, esta foi construída a partir de um outro supervisorio Para tanto foi realizado um estudo sobre sistemas supervisórios afim de se escolher o mais adequado ao projeto. Durante este projeto foram também estudadas as características do sistema real implantado na empresa de modo a compreender os seus aspectos que poderiam estar influenciando na ocorrência de falhas de operação na opinião dos operadores de uma subestação da empresa. O objetivo desta análise foi propor mudanças na interface do supervisorio atual, além de apoiar a construção do simulador no que diz respeito a recriar em ambiente de testes adequado ao treinamento dos operadores em situações críticas de operação, durante as quais terão de encontrar a melhor solução para um problema no menor tempo possível.

*Palavras Chave: Sistema Supervisório, Parametrização, Testes de Usabilidade, Interface Homem-Máquina*

## ABSTRACT

This work shows some studies about supervisory systems being aimed at the construction of a simulator for training of operators of electrical systems, in critical situations. The simulator is based on a study carried through on the human error in the operation of the systems, in company CHESF. In this company the supervision environment is developed in environment SAGE, however as this system was not used for the project of the interface with the simulator, this was constructed from another supervisory. For in such a way was carried through a study on supervisory systems similar of if choosing most adequate to the project. During this project also the characteristics of the real system implanted in the company had been studied in order to understand its aspects that could be influencing in the occurrence of imperfections of operation in the opinion of the operators of the company. The objective of this analysis was to consider changes in the interface of the current supervisory, besides supporting the construction of the simulator in that it says respect to create again in adequate environment of tests to the training of the operators in situations you criticize of operation, during which will have to find the best solution for a problem in the lesser possible time.

Key words: *Supervisory, Parametrization, Tests of Usability, Man-Machine Interface*

## Índice

ÍNDICE .....	9
ÍNDICE DE FIGURAS.....	10
INTRODUÇÃO .....	11
OBJETIVO GERAL.....	11
OBJETIVOS ESPECÍFICOS.....	11
ORGANIZAÇÃO DO TRABALHO .....	12
CAPÍTULO 1 – <i>SISTEMAS SUPERVISÓRIOS</i> .....	13
1.1 - SOFTWARE SAGE.....	15
1.2 – SOFTWARE ELIPSE.....	18
1.3 – SOFTWARE INDUSOFT .....	19
CAPITULO 2 – MODELANDO UMA SUBESTAÇÃO.....	22
2.1 - ESTUDO DE CASO .....	23
2.2 - CRIANDO O MODELO.....	24
2.2.1 – Criação das telas .....	25
2.2.2 – Criação das planilhas de tarefas.....	29
CAPITULO 3 – COMUNICAÇÃO .....	31
CAPITULO 4 – CONCLUSÕES E SUGESTÕES .....	34
REFERENCIAS BIBLIOGRÁFICAS .....	35

## Índice de Figuras

Fig 2.1. 1 – Sistema de supervisão e controle.....	23
Figura 2.2. 1 – Configurando o modo de segurança dos grupos .....	26
Figura 2.2. 2 – Tela Login .....	27
Figura 2.2. 3 – Função Open Screen utilizada para navegação .....	27
Figura 2.2. 4 – Tela Principal .....	28
Figura 2.2. 5 – Tela Aviso.scr.....	28
Figura 2.2. 6 – Tela Sinótico e seu detalhamento .....	29
Figura 2.2. 7 – Planilhas de tarefas: 1 - Trend, 2 - Alarm e 3 -Math .....	30
Figura 3. 1 – Salvando telas em formato HTML .....	32
Figura 3. 2 – Aba Web: Configurando o servidor .....	32

## **Introdução**

Na medida em que o desenvolvimento tecnológico cresce, os equipamentos vão se tornando cada vez mais confiáveis, por conseguinte, a causa das falhas do sistema vai passando a ser do operador. De acordo com levantamentos feitos em algumas empresas mais de 80% das falhas na operação são oriundas do fator humano. O que indica a necessidade do desenvolvimento de interfaces de operação que ofereçam recursos para o aprendizado e para apoio à decisão, o que levaria a operação mais segura destes sistemas.

Neste sentido, um ambiente que emule o sistema real de operação pode ser de grande valia no treinamento de operadores, o que será conseguido a partir da parametrização de um software do tipo SCADA, o qual representará um ambiente real de operação, ligado a um motor de simulação. O ambiente de simulação que será abordado é uma subestação elétrica.

O processo ao qual o software supervisorio estará conectado será representado pelo modelo de uma subestação elétrica, mais precisamente a subestação de CGD. Este modelo, construído em Redes de Petri Coloridas, se comunicará bidirecionalmente com o software supervisorio.

Neste projeto serão abordados além da parametrização do software supervisorio, a implementação de uma estratégia de comunicação entre o supervisorio e o modelo que representa a operação da planta.

## **Objetivo geral**

O objetivo principal deste projeto é a construção de um simulador para o treinamento de operadores de sistemas elétricos, a partir da construção de sua interface com um sistema supervisorio, de modo a representar com o maior realismo possível a o sistema atualmente instalado na empresa CHESF.

## **Objetivos específicos**

Parametrização do supervisorio instalado na subestação de CGD, da empresa CHESF, estabelecendo a comunicação TCP/IP entre um software supervisorio e o modelo da planta, construído em Redes de Petri Hierárquicas (HCPN) de modo a viabilizar uma plataforma de estudos de IHM.

## **Organização do trabalho**

Este trabalho, além desta introdução, é composto de quatro capítulos mais referências bibliográficas.

No primeiro capítulo, Sistemas Supervisórios, são apresentados os três tipos de sistemas supervisórios envolvidos no projeto: SAGE, que é o sistema presente na CHESF de Campina Grande e os dois que foram estudados para a sua parametrização, o ELIPSE e o INDUSOFT. É apresentada uma comparação entre os softwares justificando a opção pela utilização do INDUSOFT no projeto.

No Segundo capítulo, Modelando uma Subestação, é apresentado todo o processo para a criação do modelo, que consta desde o estudo de caso, visitas a subestação, entrevista com os operadores do sistema, até a implementação do projeto no InduSoft Web Studio.

No terceiro capítulo, Comunicação, é apresentada a forma de comunicação via Web, entre a IHM com outras instâncias clientes ou com outro software, como o Excel ou Access.

No quarto capítulo, Conclusões, são apresentadas as análises dos resultados.

## **CAPÍTULO 1 – Sistemas Supervisórios**

Nos últimos anos, o constante crescimento dos setores elétrico e industrial vem sendo amparado e viabilizado por um grande desenvolvimento tecnológico, em conjunto com a modernização dos sistemas de comunicações, formando assim um ambiente ideal para a implementação de sistemas de automação cada vez mais sofisticados.

Neste escopo estão os sistemas de comunicações, o desenvolvimento de protocolos, equipamentos de medição e controle e a utilização de suportes computacionais. Neste sentido, as Interfaces Homem-Máquina – IHM exercem um papel fundamental. Uma interface mal projetada pode causar fadiga durante a operação ou até mesmo induzir a erros, e assim, comprometer o trabalho de operação do sistema em questão.

Software do tipo SCADA (Supervisory Control and Data Acquisition) é um software para supervisão, controle e aquisição de dados, usado em soluções de automação e controle de processos, são conhecidos como supervisórios. São programas muito flexíveis, através dos quais pode-se configurar todo o processo controlado, cadastrando-se todos os pontos controlados no campo em um banco de dados.

Pode-se também criar telas que representem o processo, ou seja, uma Interface Homem-Máquina, por onde se pode iniciar um processo de comando simplesmente com o uso do mouse, o que substitui os controles tradicionais, tais como botoeiras e os painéis de controle que ocupam muito espaço. Claro que atualmente vários processos nos quais foram implantados um sistema de automação com um software supervisório, não foram abandonados ou retirados os controles manuais, a justificativa dos administradores dos sistemas para este fato é na utilização do sistema manual em caso de falha do sistema supervisório, muito embora os sistemas supervisórios venham com um confiável sistema de segurança usando estratégias, tais como: duplicidade de rede, duplicidades de servidores (implementação hot-standby), duplicidade de estações.

Na maioria dos casos os softwares supervisórios vêm acompanhados de um compilador usado na execução de rotinas que podem ser criadas pelos próprios usuários para a aplicação, muitas vezes as linguagens utilizadas são próprias dos

fabricantes do software, muito embora, em alguns casos seja possível implementação em linguagens de programação mais comuns como C, C++ e outras.

É importante salientar que dentre as características dos softwares SCADA está o fato de serem configuráveis pelo usuário e a grande diversidade de dispositivos aos quais podem ser conectados e trocar informações, isso é possível devido ao fato de serem implementados diversos drivers de comunicação e protocolos, sendo os mesmos incorporados ao software, muitas vezes são disponíveis também ferramentas para a implementação de novos protocolos para o caso de inserção de novos dispositivos com protocolos de comunicação diferentes dos já implementados.

Atualmente os sistemas são implementados em plataformas Unix, Windows NT, Windows2000 e OS/2, sendo configurados em intranets com protocolos TCP/IP, o que torna possível visualização e controle do processo em qualquer lugar do mundo onde se tenha disponível uma conexão pra WEB, claro que para isso o sistema deve ter um módulo para tal tipo de conexão.

Para a implementação da IHM foi realizada uma pesquisa com os principais softwares de automação do mercado, aonde foram analisados dez softwares SCADA dedicados a automação de processos bem como a área elétrica.

A análise foi feita segundo os seguintes parâmetros:

- **Módulo de desenvolvimento:** Permite que um usuário desenvolva e teste suas aplicações.
- **Módulo Runtime StandAlone:** Programa que executa em uma máquina e controla um processo.
- **Módulo RunTime Servidor de BDTR:** Programa que executa em uma máquina e controla um processo. Difere do StandAlone por permitir outros módulos clientes.
- **Módulo Master:** Licença para executar em uma única máquina o módulo de desenvolvimento e o "runtime". Desta forma, os dois modos possuem uma única licença.
- **Módulos de Comunicação:** Consiste nos diversos protocolos e canais de comunicação que um *software* possui disponível.
- **Módulo Cliente e Servidor da Base de Dados Corporativa:** Consiste em um módulo que é caracterizado pela presença de um servidor de base de dados onde são guardados os dados históricos.

- **Módulos Clientes de Interface Homem-Máquina (IHM):** Consiste na disponibilidade de configuração de IHM para máquinas que serão utilizadas como clientes.
- **Módulo Servidor de Web:** Esse módulo corresponde a um RunTime Cliente do Servidor de BDTR mas servidor de WEB. Assim, usuários via browser podem acessar as informações disponibilizadas pelo módulo servidor de Web.
- **Banco de Dados relacional para dados de tempo real:** Consiste na inter-relação entre tabelas que constituem a base de dados na qual são armazenados os valores e variáveis utilizadas pelo *Software* em tempo real.
- **Banco de Dados Relacional Para Dados Históricos:** Consistem na inter-relação entre tabelas que constituem o banco de dados onde são armazenados os dados históricos.
- **Possui demo.**
- **Vantagens apresentadas como singulares.**

Com isto foram analisados os seguintes softwares:

- Software SAGE, produzido pela CEPEL.
- Software Elipse, produzido pela Elipse.
- Software Indusoft, produzido pela Indusoft.

### **1.1 - Software SAGE**

O Sistema implementado na Chesf de Campina Grande foi desenvolvido no ambiente SAGE, assim se fez necessário estudar o software para conhecer todas as suas singularidades e desta forma, realizar a melhor escolha sobre o supervisor que será utilizado para a parametrização.

Os procedimentos para a configuração do Sistema Aberto de Gerenciamento de Energia – SAGE, que permite supervisionar e controlar um processo de produção, transmissão e distribuição de energia elétrica, inclui as seguintes etapas:

- **Banco de Dados:** carregamento dos dados em um ambiente *off-line*, gerando a Base de Dados Fonte e a partir desta a Base de Dados Referência (Imagem da Base de Dados Tempo Real em disco); também *off-line*;
- **Telas:** edição de telas e associação dos seus pontos dinâmicos com a Base de Dados Referência;
- **Interface Gráfica:** configuração do controle de acesso, dos fontes, das cores, das telas e dos relatórios.

O SAGE pode ser configurado para um modelo de dados EMS (Energy Management System) ou simplesmente para um modelo de dados SCADA (Supervisory Control and Data Acquisition System).

Para se configurar o Banco de Dados do SAGE são necessários os seguintes passos:

- Modelagem dos dados para preenchimento da Base Fonte.
- Carregamento da Base Fonte.
- Carregamento da Base de Referência.

A Base Fonte do SAGE utiliza um modelo de dados relacional. Uma base de dados relacional é modelada através de entidades, atributos e relacionamentos entre entidades.

- **Entidade:** é um ente abstrato que caracteriza um ponto fundamental para a organização das informações em uma Base de Dados Relacional, ou seja, é algo sobre o qual se deseja armazenar informações;
- **Atributo:** é o nome dado a uma característica da entidade, isto é, os atributos de uma entidade definem as informações que se deseja armazenar sobre ela;
- **Relacionamento:** é a maneira pela qual, duas entidades estão associadas.

Para se mapear entidades, atributos e relacionamentos, utilizam-se tabelas, aonde uma unidade corresponde a uma tabela e cada atributo desta unidade corresponde a um campo dessa tabela. O relacionamento é a forma como os campos entre as tabelas estão relacionados.

A configuração do SAGE está organizada segundo os subsistemas que o compõem. Para o modelo SCADA é configurado o Subsistema de Suporte

Computacional (SSC) e o Subsistema de Comunicação e Pré-processamento de Dados (SCD).

O SSC é responsável pelo gerenciamento do processo da base de dados distribuída, seletivamente replicada, residente em memória e utilizada em tempo real (módulo CMCD); é também responsável pela distribuição e monitoração dos processos nos diversos nós da rede (módulo GCD) e ainda oferece um serviço de alarmes e eventos (módulo ALR).

O SCD implementa a comunicação de dados com Centros de Controle, Unidades Terminais Remotas (UTR), Controladores Lógicos Programáveis (CLP) e outros equipamentos e sistemas diretamente conectados ao sistema de gerenciamento de energia.

A Interface Gráfica utiliza tecnologia Full Graphics, através do sistema X-Window e biblioteca de objetos gráficos OSF/Motif, apresentando controle de acesso baseado nos conceitos de Ações e Privilégios, adaptável a política da empresa. Suporta múltiplas janelas com diagramas unifilares, tabulares, gráficos de tendências, lista de alarmes e eventos, etc. Recursos gráficos de scrolling, panning, zooming, decluttering, camadas e filtragem. Apresenta ainda, Editor Interativo de Telas (com associação de objetos gráficos a endereços na base de dados), SAGE Shell (para ativação de programas diretos a partir da interface gráfica), além de Flexibilidade de configuração.

As telas do SAGE apresentam-se divididas nos seguintes visores:

- Visor de acesso: Envolve as atividades de controle de entrada e saída de usuários da interface do SAGE: ativação de programas, verificação dos privilégios do usuário e dos estudos dos programas.
- Visor de alarmes: Mostra textualmente os alarmes. Pode-se reconhecer um determinado alarme, eliminá-lo e inserir uma anotação.
- Visor de logs: Apresenta o histórico dos alarmes e eventos. Pode-se definir as datas de início e fim para ser visualizado o arquivo e também o tipo de informação que será exibida (alarme, evento).
- Visor de telas: É o sinótico da planta. Todo elemento tem uma janela de confirmação para cada elemento em que se atua.
- Visor de histórico: É como o visor de Log, soh que de forma gráfica.
- Visor de cálculos: Definição e monitoração de pontos calculados dinamicamente na planta.

- Visor de base: Visualização e alteração de dados do banco de dados do SAGE.
- Visor de processos: Controle e monitoração de processos.
- Visor de tendências: Mostra gráficos de tendências para as variáveis do processo.
- TsrMonitor: Monitoração e gerenciamento de canais de comunicação.

## 1.2 – Software ELIPSE

O Elipse SCADA é configurável pelo usuário, permitindo a monitoração de variáveis em tempo real, através de gráficos e objetos que estão relacionados com as variáveis físicas de campo. É possível fazer acionamentos e enviar ou receber informações para equipamentos de aquisição de dados. Apresenta uma linguagem de programação denominada Elipse Basic, que permite automatizar diversas tarefas.

O Elipse encontra-se disponível em quatro versões, que se diferenciam na sua funcionalidade.

- **Versão View:** É indicada para aplicações simples. As informações recebidas pelo View estão disponíveis também para outras aplicações que possam trabalhar com DDE (Dynamic Data Exchange). Apresenta funções de monitoramento e controle, comunicação com PLCs e outros equipamentos via drivers DLL, inclusive em blocos. Possui objetos de tela para a produção de interfaces e permite a importação de imagens de editores gráficos. Pode ser utilizado como servidor para aplicações remotas.
- **MMI (Man Machine Interface):** É indicada para aplicações de médio porte, onde é necessário o armazenamento de dados, tratamento de informações e criação de relatórios complexos. Possui além das características presentes na versão View, históricos, receitas, relatórios, CEP (Controle Estatístico de processos), novos objetos de tela e log de alarmes em disco.
- **Pro (Professional):** É indicada para aplicações de qualquer porte, que envolvam comunicação em rede, seja local ou remota ou ainda que necessitem a troca de informações com bancos de dados. Possui além das características presentes na versão MMI, suporte ODBC (Open Database Connectivity), suporte DAO (Data Access Objects).

- **Power:** Versão destinada a supervisão de subestações e sistemas elétricos. Permite conexão IEDs (Intelligent Electronic Device) e RTU (Remote Terminal Units) através de qualquer protocolo de comunicação.

O Elipse SCADA possui três módulos de operação, Configurador, Runtime e Master. Os módulos Configurador e Master são destinados para a criação e desenvolvimento de aplicativos, e o módulo Runtime permite apenas a execução destes. O módulo ativo é definido a partir de um dispositivo de proteção (hardkey) que é acoplado ao computador, na ausência deste dispositivo o software pode ser executado em modo de demonstração (Demo), no qual possui todos os recursos existentes no módulo Configurador, com exceção de que trabalha com um máximo de 20 tags (variáveis de processos) e permite a comunicação com equipamentos de aquisição de dados por até 10 minutos.

O Elipse possui dois Plug-ins (ferramentas adicionais adquiridas separadamente) os quais tornam possível a monitoração de sistemas através de recursos de captura, registro e transmissão digital de imagens em tempo real (Plug-in Elipse Watcher) e a supervisão processos através da Internet (Plug-in Elipse Web).

### 1.3 – Software INDUSOFT

As aplicações IWS rodam em microcomputadores conectados em tempo real com máquinas ou processadores através de controladores programáveis, equipamentos remotos de I/O e outros equipamentos para aquisição de dados.

As aplicações desenvolvidas pelo Indusoft Web Studio consistem de telas de operação animadas, drivers de comunicação (para PLCs e protocolos de E/S (I/O) tais como Profibus, DF1, etc.), banco de dados de tags e módulos adicionais, tais como monitores de alarme (Alarm), lógica (Math), gráficos de tendência (Trend), receitas (Recipes), ações baseadas em eventos (Scheduler), e sistema de segurança (Security System). Estas aplicações interfaceiam com sistemas de E/S industriais e outras aplicações em Windows no ambiente de runtime utilizando protocolos ODBC, DDE, NetDDE, OPC, ou TCP/IP. Pode-se operar a aplicação através de um navegador comum de internet. A interface da aplicação do IWS com sistemas I/O industriais e outras aplicações em modo runtime utilizam os seguintes protocolos:

- ODBC (Open Database Connectivity)
- DDE (Dynamic Data Exchange)
- NetDDE (Network Dynamic Data Exchange)
- OPC (Open Connectivity)
- TCP/IP (Transmission Control Protocol / Internet Protocol)

O produto consiste de dois componentes, o software de desenvolvimento e o software runtime, que pode rodar em qualquer estação de interface com o sistema operacional Windows NT/2000/XP/2003 Server ou Windows CE. Após o desenvolvimento de uma aplicação, pode-se rodá-la em sua estação de trabalho de desenvolvimento ou fazer um download da aplicação para uma estação de trabalho em runtime (utilizando uma conexão TCP / IP ou em série) e assim rodá-la com a utilização de softwares Indusoft Web Studio ou CEView. A estação runtime processa os dados dos equipamentos conectados a ele (PLCs, E/S, etc) de acordo com parâmetros definidos na aplicação e reagem, exibem e armazenam os dados.

O IWS apresenta os seguintes recursos:

- Interface Web que permite visualizar telas em um navegador Web (thin client) através da Internet/Intranet e trocar dados com o servidor através de um protocolo TCP/IP.
- Configuração e gerenciamento online de aplicações remotas.
- Biblioteca com mais de 100 símbolos e objetos dinâmicos.
- Suporte a objetos Active-X.
- Uma API de arquitetura aberta que permite troca de valores de tags do IWS com produtos de softwares externos com suporte à arquitetura Microsoft .Net.
- Mais de 150 drivers de comunicação com dispositivos diversos.
- Expressões lógicas e linguagem com mais de 200 funções.
- Integração com pacotes baseados em PC tais como ASAP, ISaGRAF, SteepleChase, e Think&Do.

O software apresenta a versão Demo do produto em seu site para Download de forma totalmente gratuita. Esta versão permite ao usuário utilizar o software por 40 horas, incluindo também o Indusoft's CEView para CE v3.0 e CE.Net. A empresa IWS do Brasil, representante dos produtos Indusoft no Brasil, disponibilizou uma licença FULL para a realização do projeto. Esta licença permite a utilização de 512 mil tags, 24

drivers, 32 Web Clients, módulos de engenharia e runtime. Foi feita então a escolha por este software para a parametrização do supervisório na subestação de CGD.

## Capítulo 2 – Modelando uma subestação

Em muitas situações, o sistema de automação auxilia o operador em situação de operação normal, no entanto é pouco eficiente em situações críticas, nestas situações, além da complexidade inerente à busca de soluções, as tarefas são mais complexas e os procedimentos ditados pelas normas são mais estritos do ponto de vista do erro humano.

Em particular, na interação com sistemas de supervisão industrial, o volume de informações apresentado ao operador da planta é tipicamente elevado. Nos sistemas supervisórios, a representação do sinótico de uma planta industrial permite que o usuário (operador) obtenha informações oriundas de diferentes tipos de equipamento e sistemas, na forma de diagramas, controles para ajuste de set-points, listas de alarmes, entre outras de complexidade semelhante. Algumas destas informações demandam uma resposta do usuário em um prazo estrito, além de impor restrições na navegação entre opções do sistema.

A concepção de Interfaces por muito tempo foi considerada como uma atividade de menor prioridade que as questões relacionadas às funcionalidades (Scherer, 2004). A partir dos anos 70, com a introdução e popularidade dos terminais de visualização em ambientes de trabalho, cada vez mais pesquisadores são motivados a estudarem a concepção de interfaces. Essa atenção se traduziu em 1983 na criação do comitê de ergonomia da ISO, ISO/TEC 159 ao qual foi destinada a tarefa de elaboração de um padrão relativo à ergonomia.

A interface de um sistema é o ponto no qual o usuário interage com o sistema. Desta forma dois produtos que implementam as mesmas funcionalidades podem ser diferenciados de acordo com a facilidade de uso, com a clareza das informações que auxiliam o usuário durante a execução das tarefas e com a eficácia de suas operações.

O projeto consiste na implementação de um sistema de supervisão e controle na configuração StandAlone. O processo a ser controlado consiste na subestação (SE) Campina Grande II (CGD) do sistema CHESF, aqui a subestação será simulada através de um modelo hierarquico em Redes de Petri Colorida (HCPN). O protocolo de comunicação a ser utilizado pela estação de trabalho é o TCP-IP.

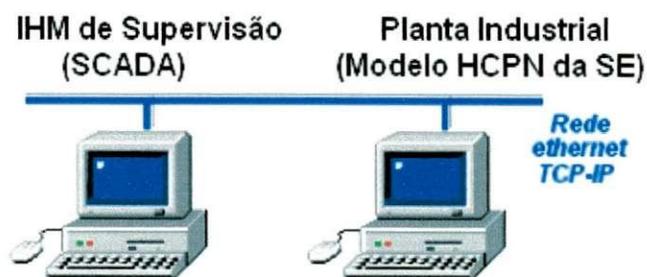


Fig 2.1. 1 – Sistema de supervisão e controle

## 2.1 - Estudo de caso

O cenário desejado para o projeto era de o operador se conectar ao sistema, abrir o sinótico, verificar os alarmes, retornar ao sinótico, atuar em um disjuntor, ir para o gráfico de tendências, voltar para o sinótico, desfazer a operação no disjuntor e atuar em outro, ir para a tela de alarmes, voltar para o sinótico, desfazer a atuação anterior, atuar na terceira chave, observar as janelas de alarme e tendências, e finalizar com um logout, ou retornando a tela de sinótico.

Para este cenário portanto, fez-se necessário a construção de cinco telas:

- Tela de Login: Possui a parte de login (no modelo é apenas uma transição), que é nome e senha, botão de confirmação e botão de limpar. O login pode ser correto ou errado. Tem um botão para encerrar a aplicação. Os outros botões são irrelevantes para o cenário.
- Tela Principal: Permite ao usuário se desconectar do sistema, e através do menu pode-se chamar os outros 3 visores necessários para o estudo de caso (sinótico, gráfico de tendências e alarmes/log).
- Sinótico: Possui o sinótico da planta.
- Alarmes/log: Mostra textualmente os alarmes.
- Visor de tendências: Mostra gráficos de tendências para as variáveis do processo.

Terminado o estudo de caso fez-se necessário uma visita a subestação Campina Grande Dois (CGD), localizada na cidade de Campina Grande – PB, afim de observar a forma de operação do software e seus recursos. Na ocasião, aproveitou-se para conversar com os operadores sobre o sistema, analisando suas singularidades e possíveis falhas. Detectou-se que algumas chaves ainda são operadas manualmente, apesar de serem monitoradas pelo supervisor. Na ocorrência de alguma falha é

gravado em um arquivo de histórico o horário da ocorrência, o operador que estava atualmente responsável, e o procedimento que foi adotado para sua solução. O fato de todas as falhas ficarem registradas em arquivo, cria nos operadores uma preocupação extra para que nenhuma falha ocorra durante o seu turno de trabalho.

Uma singularidade observada foi o recurso de zoom presente na tela de sinótico, o qual permite ao operador ter um detalhamento da tela.

## **2.2 - Criando o modelo**

Um modelo é uma representação das principais características de um objeto ou sistema e pode consistir em uma descrição matemática ou uma representação física. A utilização de métodos formais na modelagem de sistemas se baseia no uso de notações com semântica bem definida, favorecendo o estabelecimento de métricas de desempenho e possibilitando a discussão de propriedades do sistema modelado.

Neste trabalho, a utilização de modelos é motivada pela necessidade de investigar propriedades relacionadas à ergonomia da interface com o usuário sem necessariamente lidar com ela de uma maneira direta. Frequentemente, no setor industrial, lidar com o sistema real é inviável ou mesmo impraticável. A partir da utilização de modelos é possível constatar a existência de propriedades desejáveis bem como propriedades indesejáveis do sistema modelado, isto possibilita o estudo da viabilidade de implementação soluções para eliminá-las.

Para a criação do modelo procurou-se seguir fielmente as telas presentes no sistema SAGE, não só a sua aparência mas também a sua forma de navegação para que o operador sinta-se completamente familiarizado com a arquitetura. Assim, embora para a simulação do cenário fosse suficiente a criação de apenas cinco telas, foram criadas telas adicionais com o único objetivo de manter fidelidade a navegação, evitando que o operador “pule” algumas telas durante a manipulação do modelo. As telas apresentam desabilitados os botões que estão fora do escopo do projeto, e ao serem pressionados apresentam a mensagem: “Opção indisponível”.

Estando definidas todas as telas necessárias à parametrização, iniciou-se o estudo do supervisor Indusoft. Foram estudados dois documentos disponibilizados pelo fabricante:

- **Getting Started Guide:** Guia prático, que informa sobre todas as funcionalidades do software, modo de criação e customização de um projeto.
- **Apostila de treinamento:** Apostila utilizada para treinamento de pessoal, que contém informações passo-a-passo da criação de um projeto do zero, inserção de todas as suas ferramentas até a comunicação via Web.

Terminado o treinamento com o supervisor iniciou-se a construção do modelo, foi criado então um novo projeto denominado "SimuladorChesf", no qual começaram a ser produzidas as telas. O IWS disponibiliza a importação de Bitmaps, o que tornou possível a construção de objetos gráficos por softwares externos e depois simplesmente importá-los, o que permitiu tornar a aparência do modelo a mais próxima possível com a do real. O processo de criação foi dividido em três estágios:

- **Criação das telas e biblioteca de símbolos:** Nesta etapa preocupou-se com a parte gráfica, desenvolvendo a biblioteca de símbolos elétricos a ser utilizada no projeto, criando botões, e montando toda a interface. Para inserir um comando a algum objeto da tela utilizou-se a propriedade "Command", a qual permite modificar o valor das tags (tipo de variável, que pode ser booleana, inteira, real, string, ou classe) através de uma lógica baseada em planilhas. Através da propriedade "Position" é feito o gerenciamento da exibição dos objetos de acordo com os valores das tags.
- **Criação de planilhas de tarefas (Alarmes, Matemáticas e Trend):** Nas planilhas de alarmes e de trend são inseridos as tags que deseja-se monitorar e suas propriedades. As planilhas matemáticas (math) são utilizadas para a realização das lógicas inerentes ao projeto.
- **Criação da comunicação (TCP/IP):** Desenvolveu-se um sistema de comunicação via Web que permite operar o modelo em qualquer uma estação Web Thin Client.

### **2.2.1 – Criação das telas**

A primeira tela a ser produzida foi a tela "Login" que permite ao usuário informar seu nome e senha para entrar no modelo. O IWS possui um modo de segurança que permite ao desenvolvedor criar grupos de usuários e atribuir a cada um destes grupos níveis de segurança. Para o projeto foram criados três grupos:

- Engenharia: Permite ao usuário acesso total ao sistema. Destinado para o desenvolvedor do produto.
- Operação: Permite ao usuário acesso a toda parte de operação do sistema, mas restringe acesso ao modo de engenharia, não permitindo assim customizações no produto.
- Guest: Permite apenas a visualização do sistema. Usuários deste grupo não necessitam de senha.

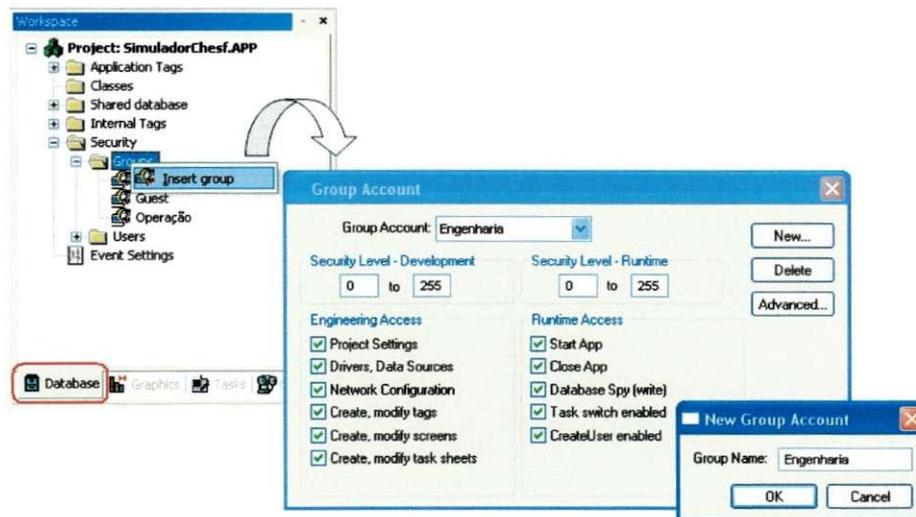


Figura 2.2. 1 – Configurando o modo de segurança dos grupos

Para a construção da tela foram utilizadas as ferramentas gráficas do IWS (Rectangle, Line, Button, Fill color, etc.). No botão Login, através da propriedade Command, foi inserida a função interna Logon(), que permite especificar o usuário e a senha, identificando desta forma a que grupo o mesmo pertence, definindo assim suas permissões. O botão de Logout foi habilitado com a função Logoff() que desconecta o usuário do sistema.

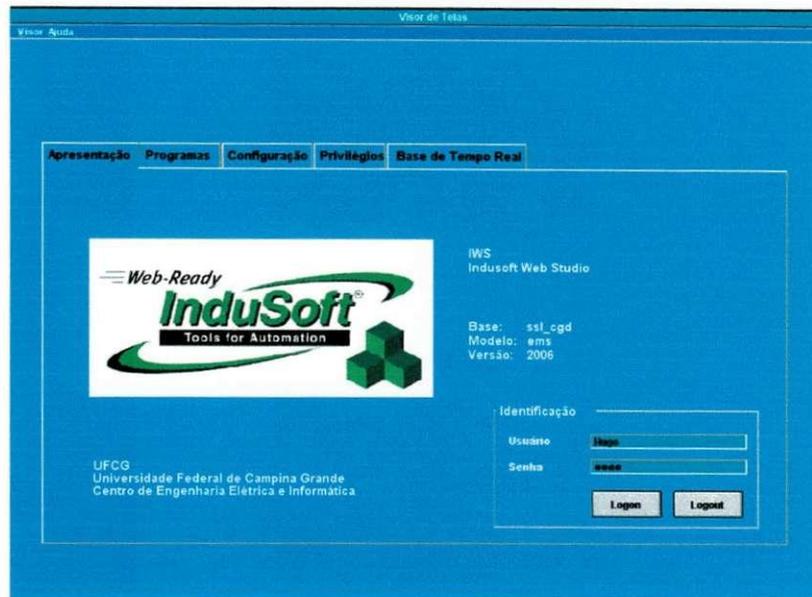


Figura 2.2. 2 – Tela Login

O sistema de abas foi construído através de retângulos habilitados com a função Open Screen, que permitiu realizar a navegação entre as telas.

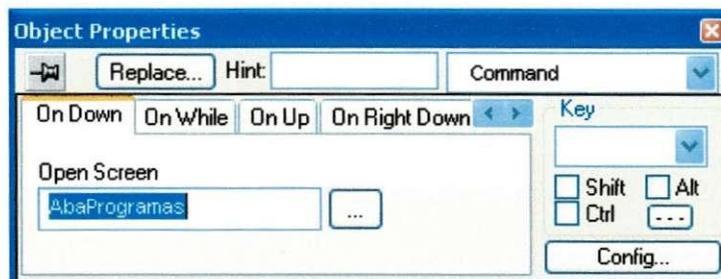


Figura 2.2. 3 – Função Open Screen utilizada para navegação

A segunda tela produzida foi a “Principal”, que permite acesso as demais telas de interesse. Esta tela permite ao usuário ter acesso as telas de Logout, Sinótico, a tela Campina Grande II (CGD) e a tela CGD - CGU - BVT, que proporciona uma visão geral das subestações Campina Grande I (CGU), Campina Grande II (CGD) e Bela Vista (BVT). Os demais botões presentes na tela encontram-se desabilitados, pois não fazem parte do escopo do projeto, ao serem selecionados abrem a tela Aviso.scr.

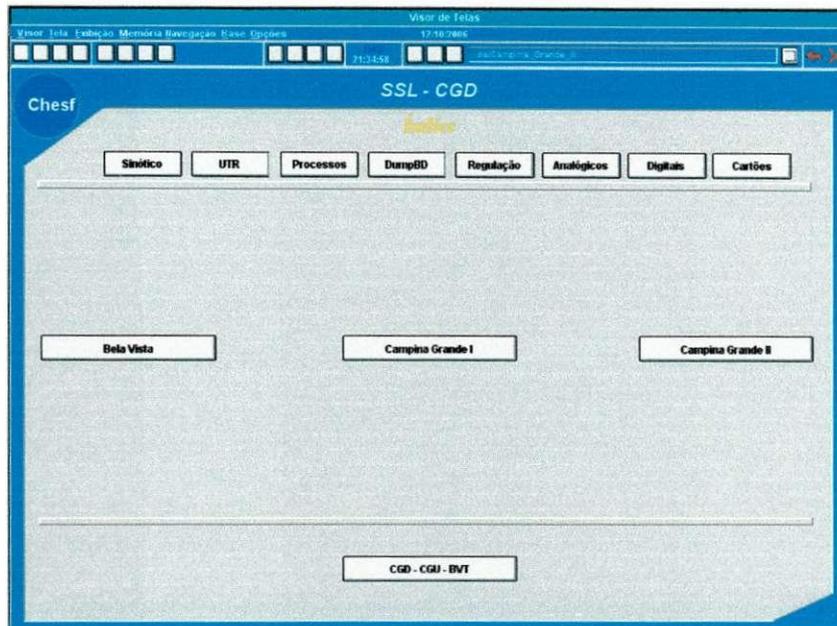


Figura 2.2. 4 – Tela Principal

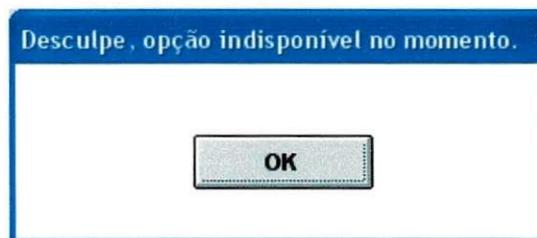
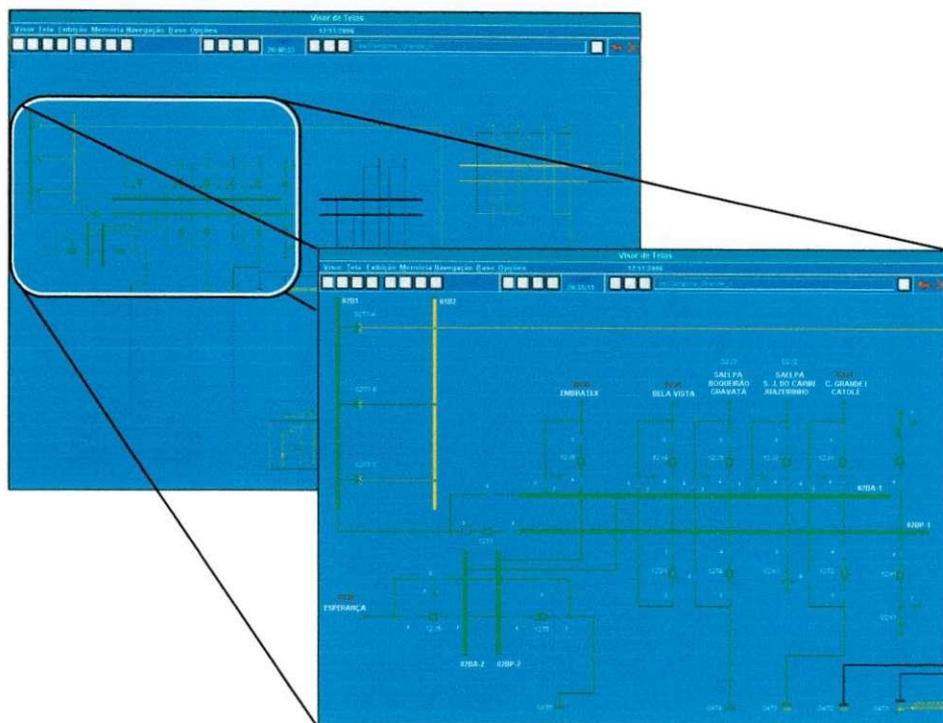


Figura 2.2. 5 – Tela Aviso.scr

Para a construção da tela de sinótico fez-se necessário algumas modificações com relação a tela original apresentada no SAGE. O IWS não apresenta a funcionalidade de zoom, então para a obtenção deste tipo de efeito foram construídas mais cinco telas, uma para o detalhamento de cada barramento. Estas telas são acessadas com um simples clicar do mouse sobre a área que se deseja visualizar o detalhamento. A construção das chaves foi feita através da superposição de dois bitmaps habilitados com as ferramentas position e command.



**Figura 2.2. 6 – Tela Sinótico e seu detalhamento**

Para as telas de alarmes e Visor de tendências foi utilizada uma funcionalidade do IWS denominada Background Tasks, a qual é responsável pela execução de lógicas (scripts) contidos nas planilhas matemáticas (math) e pelo gerenciamento dos módulos configurados, como nas planilhas de Alarmes, Receitas, Tendências e Relatórios. Através desta ferramenta foram introduzidas as tags de interesse e suas respectivas propriedades (Valor Máximo, mínimo, etc.).

### **2.2.2 – Criação das planilhas de tarefas**

Através das planilhas de tarefas são configuradas todas as lógicas necessárias a aplicação. Para o projeto foram utilizadas as planilhas Alarm, Trend e Math, afim de atenderem respectivamente a criação de grupos de alarmes, grupos de tendências e para a inserção de matemáticas.



Figura 2.2. 7 – Planilhas de tarefas: 1 - Trend, 2 - Alarm e 3 -Math

As planilhas estão localizadas na aba Tasks no Indusoft e foram inseridas clicando-se com o botão direito do mouse e selecionando a opção "Insert". A planilha matemática (Math) apresenta duas colunas, a primeira é denominada Tag Name, aonde são colocadas as tags que irão receber o valor do cálculo realizado na segunda coluna, Expression. As expressões matemáticas são criadas a partir das funções da Linguagem Indusoft Web Studio, que diferentemente de outras linguagem baseadas em textos estruturados, não se faz atribuição de valores na coluna Expression.

Na planilha de alarmes foram inseridas as tags a serem monitoradas, o tipo de alarme (Hi, Lo, HiHi ou LoLo) e os limites estabelecidos. Foram inseridas também mensagem a serem informadas no momento que o alarme é disparado.

Na planilha Trend inseriu-se as tags de interesse e foi selecionado o Check Box "Save On Tag Change", para que o sistema salve os valores das tags toda vez que elas mudarem de valor.

### Capítulo 3 – Comunicação

É importante salientar que dentre as principais características dos *softwares* SCADA está o fato de serem configuráveis pelo usuário e a grande diversidade de dispositivos aos quais podem ser conectados e com os quais podem trocar informações (Alves, 2004). Isso é possível devido ao fato de incorporarem diversos protocolos de comunicação. São disponíveis também ferramentas para a implementação de novos protocolos, possibilitando a inserção de novos dispositivos com protocolos de comunicação diferentes dos já implementados.

Atualmente os sistemas são implementados em plataformas Unix, Windows NT, Windows2000 e OS/2, sendo configurados em *intranets* com protocolos TCP/IP, o que torna possível a visualização e controle do processo de qualquer local conectado a WEB. Para isso o sistema deve ter um módulo agregado que possibilite tal tipo de conexão.

Antes dos sistemas OPC se tornarem realidade, a obtenção de informação de CLP's era bem mais complexa. Cada SCADA deveria possuir drivers específicos para cada CLP, assim sendo, um SCADA como o Intouch, por exemplo, deveria ter um driver para a Allen-Bradley, outro para Siemens, etc., e além das diferentes marcas de controladores, muitas vezes era preciso mais de um driver para cada fabricante, pois esses diversificam a comunicação criando diferentes famílias de CLP's. Essa comunicação era baseada em DDE – "*Dynamic Data Exchange*", "Troca de Dados Dinâmica".

Para a parametrização do supervisor da subestação de CGD existia a necessidade de se estabelecer uma comunicação entre o software supervisor e um modelo HCPN em TCP/IP, afim de viabilizar uma plataforma de estudos para testes relacionados a construção de IHM ergonômicas e avaliar o grau de conformidade da plataforma montada com o sistema real. Foi criada então uma interface Web que permite visualizar as telas em um navegador Web (thin client) através da Internet/Intranet e trocar dados com o servidor através de um protocolo TCP/IP.

Primeiramente as telas foram salvas em formato HTML, através do menu File > Save as HTML.



Figura 3. 1 – Salvando telas em formato HTML

Após este comando ter sido executados os arquivos Web foram armazenados na subpasta \web no diretório da aplicação. Em seguida configurou-se a estação servidor, para isto através do menu Project > Settings selecionou-se a guia Web, na qual especificou-se o endereço IP da estação do servidor no campo Data Server IP Adress. A estação Web Thin Client troca dados on-line (valores de tag) com a estação que foi especificada neste campo.

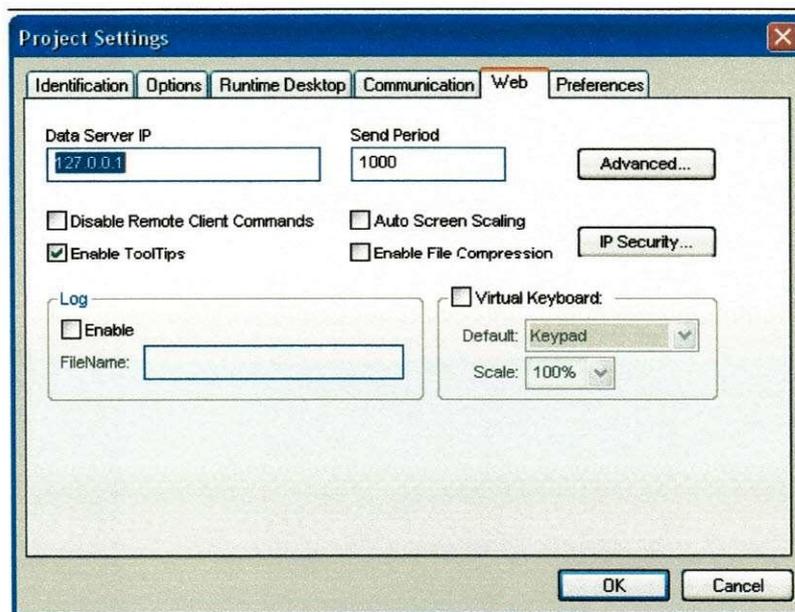


Figura 3. 2 – Aba Web: Configurando o servidor

A Microsoft fornece Web servers (servidores Web) para qualquer sistema operacional da Microsoft (IIS). Para efeito de testes foi copiado o arquivo NTWebServer.exe localizado no subdiretório \BIN do Indusoft e colado no diretório Web da aplicação e em seguida foi executado. Para atualizar os ajustes Web selecionou-se o menu Tools > Verify Application.

Selecionou-se ícone Run Application para executar a aplicação localmente na estação do servidor e disponibilizar deste modo para ser acessada via Web em qualquer estação Web Thin Client, bastando-se abrir um navegador de internet e digitar o endereço URL para abrir a TelaVisor.html ([http://IP do Server/TelaVisor.html](http://IP_do_Server/TelaVisor.html)). Pode-se alterar o status de qualquer chave localmente (estação do servidor) utilizando o módulo run-time Viewer, ou remotamente (Web Thin Client), utilizando o navegador.

## Capítulo 4 – Conclusões e Sugestões

Apesar da automação nos processos industriais, inclusive no setor elétrico, a intervenção humana ainda é indispensável e se dá mediante a interação com a planta através de sua interface. Neste sentido a utilização de regras de concepção de interfaces com princípios ergonômicos leva o projetista ao desenvolvimento de interfaces menos susceptíveis a falhas de projeto. Neste contexto, a utilização de modelos vem demonstrando ser uma forma vantajosa de representação das interfaces industriais.

Como trabalho futuro pode ser inserido o modelo em Redes de Petri Coloridas que se comunicará bidirecionalmente com o software supervisor, e desta forma poderiam ser realizados os testes de usabilidade do sistema. Poderiam ser incluídos gráficos de tendências, os quais teriam como parâmetros o tempo levado para a realização de tarefas versus o nível de experiência do usuário. O tempo para a solução de problemas é cobrado com bastante veemência no ambiente real, o que entra em confronto com a responsabilidade das decisões a serem tomadas, gerando no operador uma atmosfera de pressão e nervosismo, sendo, portanto um agravante para uma possível falha humana.

Na interação com um simulador o operador pode não se sentir imerso no ambiente se a visualização das telas não for realista, assim uma decisão errada pode ser percebida como sem consequências tão graves quanto em um ambiente real. Portanto os tempos envolvidos na interação podem tornar o operador mais suscetível a erros. Uma sugestão para aumentar o realismo da interação seria a atribuição de um tempo máximo para a resolução de cada tarefa, visualizado pelo operador através de um contador regressivo.

## Referencias Bibliográficas

Duarte, A. N.; Sauv , J. P.; Cirne Filho, W; Figueiredo, J. A.; Rocha Neto, E.; Melo, M. S. de; Ara jo, A. S. de. *Desenvolvimento de Uma Ferramenta de Tratamento de Eventos em Redes El tricas*. V Simp sio De Automa o De Sistemas El tricos, 2003, Recife. p. 210-214.

Farias, G. F. *Modelo de Funcionalidades de Interface Homem-M quina Industriais*. Relat rio de Projeto de Pesquisa, P s-Gradua o em Engenharia El trica da UFPB, Campina Grande, PB, 1999.

Farias, G. F., *Diretrizes para Projeto de Interfaces Homem-M quina Aplicadas a Sistemas de Supervis o de Processos Industriais*, Disserta o de Mestrado, Curso de P s-Gradua o em Engenharia El trica, UFPB, 1994.

Guerrero, C. V. S., *MEDITE – Uma Metodologia Orientada a Modelos para Concep o de Interfaces Ergon micas*. Disserta o de Mestrado, Universidade Federal de Campina Grande, fevereiro, 2002.

Nakatani, Y., Nakagawa, T., Terashita, N., Umeda, Y., *Human Interface Evaluation by Simulation*, IEEE SIXTH ANNUAL FACTORS MEETING, 1997.

Ogata, Katsuhiko, *Engenharia de Controle Moderno*. Editora Prentice Hall do Brasil LTDA, Segunda Edi o. Rio de Janeiro, RJ. 1993, pp 74-77.

Queioz, J. E. R. *Abordagem H brida para Avalia o da Usabilidade de Interfaces com o Usu rio*. Tese de Doutorado, Universidade da Para ba, Junho, 2001.

SAGE - Sistema Aberto de Gerenciamento de Energia - *Guia do Usu rio*. Dispon vel em <http://www.sage.cepel.br/documentacao/manusistema.html> agosto de 2006.

Scaico, A., Turnell, M. F. Q. V. *Análise do Tempo na Interação do Usuário com Sistemas de Automação Industrial*. Congresso Brasileiro de Automação, setembro de 2002.

Scaico, A. *Aplicação de um Modelo de Navegação de IHM ao Contexto de Sistemas Industriais*. Dissertação de Mestrado, Universidade Federal da Paraíba, agosto de 2001.

Scaico, A., Turnell, M. F. Q.V., Perkusich, A. *Modelagem da Navegação de Interfaces com o Usuário de Sistemas de Automação Industrial*. V SBAI – Simpósio Brasileiro de Automação Inteligente, novembro de 2001.

Sousa, M. R. F. *Avaliação Iterativa de Especificação de Interfaces com Ênfase na Navegação*. Tese de Doutorado, Pós-Graduação em Engenharia Elétrica da UFPB, Campina Grande, PB, 1999.

Turnell, M. F. V. Q., *Conceitos e Projeto de Interfaces Usuário-Computador, Notas de Aula da Disciplina Projeto de Interface Homem-Máquina*, Curso de Pós-Graduação em Engenharia Elétrica, UFPB, 2000.

Zeilmann, R. P., *Uma estratégia para controle e supervisão de processos industriais via internet*. Dissertação de Mestrado Universidade federal do Rio Grande do Sul, 2002.

Elipse - *Manual do Usuário*. Disponível em [http://www.elipse.com.br/elipse/suporte\\_manuais.aspx?id=1](http://www.elipse.com.br/elipse/suporte_manuais.aspx?id=1) agosto de 2006.

Scaico, A., Turnell, M. F. Q.V., Perkusich, A. *Modelagem da Navegação de Interfaces com o Usuário de Sistemas de Automação Industrial*. V SBAI – Simpósio Brasileiro de Automação Inteligente, novembro de 2001.

Alves, J., Turnell, M. F. Q.V., Perkusich, A. *Modelagem da Interface Homem Máquina de uma Subestação Elétrica*. Dissertação de Mestrado, Universidade Federal de Campina Grande, agosto de 2004.

