

**PONTIFÍCIA UNIVERSIDADE CATÓLICA DO PARANÁ  
PROGRAMA DE PÓS-GRADUAÇÃO EM ENG. DE PRODUÇÃO E SISTEMAS**

**MARCO TÚLIO CORRÊA DE SIQUEIRA**

**DESENVOLVIMENTO DE UM WEBLAB PARA UMA PLANTA DE PROCESSO  
TÉRMICO EMPREGANDO UMA METODOLOGIA DE PROJETO**

**CURITIBA**

**2013**

**MARCO TÚLIO CORRÊA DE SIQUEIRA**

**DESENVOLVIMENTO DE UM WEBLAB PARA UMA PLANTA DE PROCESSO  
TÉRMICO EMPREGANDO UMA METODOLOGIA DE PROJETO**

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia de Produção e Sistemas. Área de concentração: Concepção de Produtos e Sistemas do Centro de Ciências Exatas e Tecnológicas, da Pontifícia Universidade Católica do Paraná, como requisito parcial à obtenção do título de mestre em Engenharia de Produção em Sistemas

Orientador: Prof. Dr. Luciano Antônio Mendes

**CURITIBA**

**2013**

**MARCO TÚLIO CORRÊA DE SIQUEIRA****DESENVOLVIMENTO DE UM WEBLAB PARA UMA PLANTA DE PROCESSO  
TÉRMICO EMPREGANDO UMA METODOLOGIA DE PROJETO**

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia de Produção e Sistemas. Área de concentração: Concepção de Produtos e Sistemas, do Centro de Ciências Exatas e Tecnológicas, da Pontifícia Universidade Católica do Paraná, como requisito parcial à obtenção do título de mestre em Engenharia de Produção e Sistemas

**COMISSÃO EXAMINADORA**

---

Professor 1  
Instituição 1

---

Professor 2  
Instituição 2

---

Professor 3  
Instituição 3

Curitiba, \_\_\_\_ de \_\_\_\_\_ de 20\_\_.

## **AGRADECIMENTOS**

Agradeço a minha adorável e paciente esposa, que sempre esteve presente nos momentos difíceis desta caminhada.

Agradeço ao professor Dr. Luciano Antonio Mendes que não poupou esforços para me orientar durante a elaboração deste trabalho. Sua dedicação, perseverança e humildade são certamente louváveis.

Agradeço aos coordenadores, professores e funcionários do Programa de Pós-graduação em Engenharia de Produção e Sistemas, em especial ao colaborador Henrique Ferazza, pelo auxílio e atenção prestados.

Agradeço aos alunos Diego Araujo, Karine Bauer, Pedro Casagrande e Rodrigo Delay pelo esforço e contribuição prestados.

Agradeço a empresa B&R Automation pela doação de equipamentos, pelo treinamento e auxílio ofertados.

Agradeço a meus familiares pelo carinho e amor incondicional, que tanto me ajudou a seguir em frente.

Agradeço a meus amigos, em especial a Marcio Debner, que sempre estiveram ao meu lado fornecendo apoio e motivação.

Agradeço a todos que direta ou indiretamente contribuíram para a realização deste trabalho.

Agradeço ao SENAI pelo auxílio e incentivo ofertados.

Agradeço a Deus por mais esta oportunidade.

## RESUMO

WebLabs, ou laboratórios operados remotamente, são sistemas que permitem a realização à distância de experimentos de laboratório reais, em modo *online*, com interatividade e segurança, utilizando recursos de hardware, software e infraestrutura de rede, e que têm aplicações em ensino e pesquisa. Este trabalho apresenta, na forma de um estudo de caso, o projeto e execução de um Weblab interativo para uma planta de processo térmico com base em uma proposta de modelo sistemático de desenvolvimento voltado para WebLabs – processo de desenvolvimento de WebLabs (PDW), em seis fases e contemplando a engenharia concomitante de três camadas: sistema físico, hardware e software. A proposta é baseada em conhecimentos de metodologia de projeto na engenharia e emprega ferramentas como: QFD, Análise Funcional, Matriz Morfológica e Método de Pugh. Os objetivos do trabalho são a realização de um WebLab e a realimentação do modelo PDW com contribuições advindas das experiências e lições aprendidas. Uma planta experimental de temperatura foi modificada para viabilizar a operação remota e atender aos requisitos definidos na etapa inicial do projeto - em particular, permitir a realização de experimentos comuns no ensino de engenharia de controle, tais como o ajuste de parâmetros de controladores On-Off / PID e a determinação do modelo dinâmico de um sistema, neste caso térmico. Foram empregados recursos de instrumentação virtual sobre plataforma PC e linguagem LabVIEW<sup>TM</sup> como base para a implementação e integração das diversas funcionalidades. Para variar a dinâmica do sistema térmico e diferenciar experimentos, a admissão de ar comprimido e a abertura da janela do forno foram tornadas variáveis através de recursos de automação. Foram criadas funcionalidades para agendar sessões de utilização, manter videoconferência com o usuário, gerenciar a segurança e estabilidade da conexão de rede, armazenar e enviar por e-mail dados de sessão, e uma interface gráfica de operação de alto nível acessível desde uma página Web. Para qualificar o sistema WebLab obtido, foram realizados testes-piloto com um

grupo de usuários voluntários. Foi observada uma operação estável de todas as funcionalidades do sistema, e registrada uma opinião predominante por parte desse grupo aprovando o sistema como forma alternativa à presença física no laboratório. A adoção do modelo sistemático de desenvolvimento permitiu organizar as etapas de desenvolvimento do Weblab e as respectivas informações, tendo contribuído significativamente para se alcançar um conjunto final que atendeu aos requisitos propostos. Da experiência do desenvolvimento, foram inferidas contribuições para o aprimoramento do modelo de referência utilizado, na forma de requisitos generalizáveis a outros projetos de WebLabs ou recomendações de caráter geral: admitir o uso obrigatório de soluções que empregam recursos disponíveis para o projeto, mesmo que de capacidade superdimensionada, se necessário para manter a viabilidade técnico-econômica; em WebLabs interativos, assegurar a retomada segura de sessões interrompidas; possibilitar que ações administrativas sobre o sistema sejam tomadas à distância; coletar opiniões dos usuários sobre o sistema; e, adicionar elementos de automação para a operação remota sem criar dificuldades para a manutenção dos componentes físicos pré-existentes.

**Palavras-chave:** WebLab, Planta de Processo de Temperatura, Modelo Sistemático de Desenvolvimento de Projeto, Instrumentação Virtual, Controle de Sistema Térmico.

## ABSTRACT

WebLabs, or remotely operated laboratories, are systems that allow performing real time experiments from a remote computer in a real laboratory, *online*, with interactivity and safety, using hardware and software resources and network infrastructure, with applications in education and research. This work presents, in the form of a case study, the design and implementation of an interactive Weblab for a thermal process plant based on a systematic Weblabs development model proposal – the Weblabs Development Process (PDW), organized in six major phases which take in consideration the concurrent engineering of three layers: physical system, hardware and software. The proposal is based on engineering design and project methodology knowledge, and uses tools such as: QFD, Functional Analysis, Morphological Chart and Pugh's Method. The objectives of this work are the realization of a Weblab and the feedback of the PDW model with the contributions coming from the experiences and lessons learned. An experimental temperature process plant has been adapted to allow remote operation and to comply with the requirements defined in the initial phase of the project - in particular, allowing the performance of common educational control engineering experiments, such as the parameters adjustment of On-Off / PID controllers and the determination of the dynamic model of a thermal system. Resources of virtual instruments over PC platform and LabVIEW™ language have been employed as the base for the implementation and integration of different functionalities. In order to modify the thermal system dynamic and to distinguish experiments, the admission of compressed air and the opening of the oven window have been made variable through the use of automation resources. Functionalities have been created to schedule user sessions, establish video conference with the the remote user, manage the security and stability of the network connection, e-mail session data and provide a high level graphical user interface. In order to qualify the obtained WebLab system, pilot tests were executed with a group of volunteers. It was observed a stable operation of all the system functionalities and all the group's opinions registered point towards approving the system as an alternative to the physical presence at the lab. The adoption of the systematic development model allowed organizing the WebLab

development phases and the respective information, having played an important role to reach a solution attending the proposed requirements. From the present development experience, contributions were deduced for the improvement of the reference model employed, in the form of requirements that can be extended to other WebLab projects or general recommendations: accept certain solutions at some functions as mandatory, by using available resources, even if these are overdimensioned, to keep technical and economical viability; in interactive WebLabs, ensure the safe recovery of interrupted sessions; allow administrative actions to be commanded remotely; collect users feedback on the system usage experience; and, add automation components to allow remote operation without creating difficulties for the maintenance of the existent physical systems.

**Key-words:** WebLab, Temperature Process Plant, Project Development Systematic Model, Virtual Instrumentation, Thermal System Control.

## LISTA DE ILUSTRAÇÕES

Figura 1– Estágios do Processo de Desenvolvimento. ....	23
Figura 2 – Macroestrutura para WebLabs. ....	30
Figura 3 – Figura e diagrama da Estação de Processo de Temperatura LabVolt - Modelo 3504. ....	34
Figura 4 – Arquitetura do WebLab para a Planta de Processo Térmico. ....	46
Figura 5 – Diagrama Técnico da Solução Conceitual Selecionada. ....	46
Figura 6 – Linearização da Área de Abertura do Damper. ....	47
Figura 7 – Variação da área de abertura do <i>damper</i> em função de seu deslocamento linear. ....	48
Figura 8 – Requisitos do sistema para NI LabVIEW Development Systems. ....	50
Figura 9 – Vista do Damper Antes e Após a Automação. ....	51
Figura 10 – Circuito Eletrônico que Executa o Posicionamento do Damper. ....	52
Figura 11 – Fluxograma da Rotina de Posicionamento do Damper. ....	52
Figura 12 – Válvula Pneumática Linear Com Atuador Eletrônico. FESTO, modelo MPP – 3 – 1/8 ....	53
Figura 13 - Diagrama Elétrico do Quadro de Comando. ....	54
Figura 14 – Sistema de Comunicação Entre o PC, o CLP e os Equipamentos da Planta de Temperatura. ....	55
Figura 15 – Conjunto CLP e IHM B&R Automation e Painel de Controle. ....	55
Figura 16 – Webcam para vídeo conferência e seu suporte. ....	57
Figura 17 – Vista Geral do WebLab da Estação de Temperatura. ....	58
Figura 18 – Tela do Cadastro de Usuários. ....	59
Figura 19 – Tela do Cadastro de Usuários. ....	60
Figura 20 – Exemplo de E-mail Enviado ao Usuário Após a Confirmação do Cadastro. ....	60
Figura 21 – Arquitetura do Software de Gerenciamento do WebLab. ....	62
Figura 22 – Arquitetura do Software de Gerenciamento do WebLab. ....	63
Figura 23 – Detalhe do Arquivo Texto Gerado pelo Sistema. ....	65

Figura 24 – Estrutura do arquivo texto gerado durante o processo de avaliação do sistema.....	68
Figura 25 – Interface de Login do software de gerenciamento do WebLab. ....	70
Figura 26 – Interface de Controle do software de gerenciamento do WebLab.....	71
Figura 27 – Interface de Monitoração do Sistema.....	73
Figura 28 – Interface de Comunicação. ....	74
Figura 29 – Imagem do WebLab transmitida através do programa Skype™. ....	75
Figura 30 – Interface de Orientações sobre as Atividades Didáticas. ....	76
Figura 31 – Interface de Avaliação do sistema.....	77
Figura 32 – Interface de Instruções.....	77
Figura 33 – Gráfico contendo o histórico de uma sessão em que foi realizada a Atividade Didática 1.....	79
Figura 34 – Gráfico contendo o histórico de uma sessão em que foi realizada a Atividade Didática 2.....	80
Figura 35 – Gráfico contendo o histórico de uma sessão em que foi realizada a Atividade Didática 3.....	81
Figura 36 – Aba Dados do Programa Microsoft Excel 2010.....	81
Figura 37 – Detalhe dos Dados Formatados no Programa Microsoft Excel 2010.....	81
Figura 38 – Avaliação do Sistema pelo Grupo de Usuários Piloto. ....	82

## LISTA DE TABELAS

Tabela 1 - Lista Comparativa das Vantagens e Desvantagens entre Laboratórios Reais, Virtuais e Remotos.....	18
Tabela 2 - Fontes de Requisitos em Projetos de WebLabs. ....	29
Tabela 3 - Fontes Comuns de Especificações Técnicas em Projetos de WebLabs. .	30
Tabela 4 - Funções de Primeira Ordem no Desenvolvimento de WebLabs.....	32
Tabela 5 - Requisitos didáticos, de Usuário e Técnicos.....	38
Tabela 6 - Especificações Técnicas.....	39
Tabela 7 – Análise Funcional e Matriz Morfológica.....	41
Tabela 8 - Aplicação do Método de Pugh na seleção da melhor alternativa.....	42
Tabela 9 – Pseudocódigo representando a programação do CLP.....	56
Tabela 10 – Resultados Obtidos Versus Especificações Técnicas.....	83

**LISTA DE ABREVIATURAS E SIGLAS**

CLP	Controlador Lógico Programável
DIN	<i>Deutsches Institut für Normung</i>
EAD	Ensino a Distância
FPGA	<i>Field Programmable Gate Array</i>
HOQ	<i>House of Quality - Casa da Qualidade</i>
IEEE	<i>Institute of Electrical and Electronics Engineers</i>
IES	Instituição de Ensino Superior
IHM	Interface Homem Máquina
I/O	<i>Input / Output</i>
MIT	<i>Massachusetts Institute of Technology</i>
No.	Número
PBSA	Abordagem Sistemática de Pahl & Beitz
PDW	Processo de Desenvolvimento de WebLabs
PID	Proporcional, Integral, Derivativo
Pt-100	Termoresistência de Platina
QD	Desdobramento da Qualidade
QFD	Desdobramento da Função Qualidade
QFDr	Desdobramento da Função Qualidade no Sentido Restrito
QVGA	<i>Quarter Video Graphics Array</i>
RL	Laboratório Remoto
RTD	<i>Resistance Temperature Detectors</i>
TIC	Tecnologia da Informação e Comunicação
TQC	Controle de Qualidade Total
TCP/IP	Protocolo de Controle de Transmissão / Protocolo de Interconexão
USB	<i>Universal Serial Bus</i>
Vol.	Volume
XML	<i>Extensible Markup Language</i>

## SUMÁRIO

1	INTRODUÇÃO .....	16
1.1	CONTEXTUALIZAÇÃO DO TRABALHO .....	16
1.2	DESENVOLVIMENTOS RECENTES NA ÁREA.....	18
1.3	MÉTODOS E FERRAMENTAS NO DESENVOLVIMENTO DE PRODUTOS E SISTEMAS TÉCNICOS .....	21
1.4	TEMA E QUESTÃO DE PESQUISA .....	24
1.5	OBJETIVOS .....	25
1.5.1	Objetivo geral.....	25
1.5.2	Objetivos específicos .....	25
1.6	JUSTIFICATIVA.....	25
1.7	SOBRE METODOLOGIA DE PESQUISA EM TRABALHOS DE NATUREZA APLICADA.....	26
1.8	ESTRUTURAÇÃO DAS ETAPAS DO TRABALHO .....	27
2	METODOLOGIA .....	28
2.1	DEFINIÇÃO DOS REQUISITOS TÉCNICOS, DIDÁTICOS E DE USUÁRIO 28	
2.2	ESPECIFICAÇÕES TÉCNICAS DE PROJETO.....	29
2.3	SÍNTESE CONCEITUAL.....	30
2.4	ANÁLISE, SIMULAÇÃO E DIMENSIONAMENTO .....	32
2.5	DETALHAMENTO E DOCUMENTAÇÃO .....	32
2.6	INTEGRAÇÃO, TESTE E INICIALIZAÇÃO.....	33
3	DESENVOLVIMENTO DO WEBLAB .....	33
3.1	DESCRIÇÃO DA PLANTA DE TEMPERATURA.....	33
3.2	REQUISITOS E ESPECIFICAÇÕES .....	34
3.2.1	Atividades didáticas propostas para o WebLab .....	34
3.2.2	Determinação dos requisitos técnicos, didáticos e de usuário.....	38
3.2.3	Especificações técnicas de projeto .....	39
3.3	SÍNTESE CONCEITUAL.....	40
3.4	ANÁLISE, SIMULAÇÃO E DIMENSIONAMENTO .....	47
3.4.1	Sistema eletromecânico de acionamento do damper .....	47

3.4.2	Circuito eletrônico para controlar o damper .....	48
3.4.3	Sistema de vídeo chamada .....	49
3.4.4	Equipamentos auxiliares .....	49
3.4.5	Servidor do experimento .....	49
3.4.6	Resolução do CLP .....	50
3.5	DETALHAMENTO E DOCUMENTAÇÃO DA SOLUÇÃO .....	50
3.5.1	Sistema de automação do <i>damper</i> .....	50
3.5.2	Válvula proporcional para controle da vazão de ar comprimido .....	52
3.5.3	Sistema de proteção e intertravamento .....	53
3.5.4	Sistema de comunicação PC-CLP-Planta .....	54
3.5.5	Videoconferência via Skype™ e fixação da webcam.....	56
3.5.6	Painel de controle .....	57
3.5.7	Software para cadastrar usuários .....	58
3.5.8	Resumo dos arquivos gerados pelo sistema .....	61
3.5.9	Estrutura do software de gerenciamento do WebLab .....	61
3.5.10	Aspectos de segurança .....	68
3.5.11	Tela de Login .....	69
3.5.12	Tela de Controle .....	70
3.5.13	Tela de Monitoração .....	72
3.5.14	Tela de Comunicação .....	73
3.5.15	Tela de Atividades Didáticas.....	75
3.5.16	Tela de Avaliação ( <i>feedback</i> ) pelo Usuário .....	76
3.5.17	Tela de Instruções .....	77
4.0	INTEGRAÇÃO E TESTES .....	78
4.1	EXEMPLOS DE REALIZAÇÃO DAS ATIVIDADES DIDÁTICAS .....	78
I)	ATIVIDADE DIDÁTICA 1: Controle ON-OFF .....	79
II)	ATIVIDADE DIDÁTICA 2: Controle PID .....	79
III)	ATIVIDADE DIDÁTICA 3: Aplicando um degrau ao sistema.....	80
5	RESULTADOS E DISCUSSÃO .....	82
5.1	TESTES DE UTILIZAÇÃO POR UM GRUPO DE USUÁRIOS PILOTO .....	82
5.2	ANÁLISE DO ATENDIMENTO DAS ESPECIFICAÇÕES .....	82
5.3	ASPECTOS GERAIS DO DESENVOLVIMENTO DE SOFTWARE .....	84

6	CONCLUSÃO E CONSIDERAÇÕES FINAIS .....	85
6.1	SUGESTÕES PARA TRABALHOS FUTUROS .....	87
4	REFERÊNCIAS.....	88
	ANEXOS .....	95
1.	CASA DA QUALIDADE .....	95
2.	ESQUEMÁTICO DA PLACA DE POSICIONAMENTO DO DAMPER.....	96
3.	ESQUEMÁTICO DA PLACA DE RELÉS .....	97
4.	PROGRAMAÇÃO DO CLP .....	98
5.	FIRMWARE DO MICROCONTROLADOR ZILOG .....	104
6.	FLUXOGRAMAS.....	113

## 1 INTRODUÇÃO

As novas tecnologias desenvolvidas nas últimas décadas têm revolucionado os meios de comunicação, ampliando as possibilidades de acesso à informação através da Web. A crescente popularização da internet e o contínuo desenvolvimento de novas ferramentas ligadas a área da tecnologia da informação têm transformado e influenciado a sociedade em que vivemos, lançando para o governo e para a sociedade civil o desafio de traçar estratégias de educação sintonizadas com os modernos paradigmas de acesso à informação, Castro (2008). É neste contexto que vemos o surgimento de um novo perfil de usuário da internet, que se adapta e faz uso destas novas ferramentas, de modo a otimizar o seu tempo e flexibilizar a maneira como suas tarefas diárias são executadas. A busca por soluções adequadas, neste novo cenário, tem despertado o interesse das Instituições de Ensino Superior (IES). Encontrar meios que facilitem o acesso à educação, de modo a contornar problemas como acessibilidade à escola, problemas orçamentários e rotinas pessoais são objetivos contemporâneos das IES. Em parte, estas dificuldades estão sendo contornadas com as ofertas na modalidade de Ensino à Distância (EAD), que vêm se popularizando em função das novas tecnologias de informação e comunicação (TIC) disponíveis no mercado. Entretanto, lidar com diferentes áreas de conhecimento através do EAD não tem sido uma tarefa muito fácil em função da não uniformidade entre as disciplinas abordadas. Ko et al. (2000) afirmou que uma das barreiras ao EAD seria a sua impossibilidade de ofertar atividades práticas laboratoriais. Mendes et al. (2010) retrata esta situação da seguinte forma: “se a interação entre professor e aluno pode ser aproximada com o uso de câmeras, microfones e projetores na área humanística, as mesmas capacidades de videoconferência não seriam suficientes para lidar com as atividades práticas existentes em cursos de engenharia.”

### 1.1 CONTEXTUALIZAÇÃO DO TRABALHO

O emprego dos laboratórios virtuais tem sido adotado como alternativa para suplantando parte das carências relativas às atividades laboratoriais faltantes na educação à distância em áreas tecnológicas. Com criatividade e recursos de

software e hardware oriundos do campo das TIC's, podem-se construir alternativas virtuais de interação que atendam, ainda que de forma simulada, à diversas situações encontradas nas práticas laboratoriais. Entretanto, experimentos virtuais não conseguem reproduzir em sua totalidade as características encontradas nos laboratórios reais, não sendo, portanto, um bom substituto à experimentação prática, conforme Nedic et al. (2007). Ainda segundo este autor, laboratórios reais na forma tradicional apresentam elevados custos de manutenção, além de terem acesso restrito e ficarem ociosos a maior parte do tempo. Em meio a estas questões e aos avanços tecnológicos vivenciados nas últimas duas décadas, os laboratórios operados remotamente surgem como uma nova opção, seja em complementação ou mesmo em substituição, das aulas laboratoriais tradicionais.

WebLabs, ou laboratórios remotos, são sistemas que permitem a realização à distância de experimentos de laboratório reais, *online*, com interatividade e segurança, utilizando recursos de hardware, software e infraestrutura de rede, e que têm aplicações em ensino e pesquisa. Um experimento remoto é um ensaio executado em equipamentos físicos localizados em um laboratório, onde o controle é feito à distância (tipicamente através da internet), em contraste com um experimento em laboratório convencional, onde os alunos estão diante do equipamento e sob acompanhamento direto de um instrutor. A aplicação dos WebLabs potencializa diversas áreas da educação à distância, de modo a reforçar o ensino de conceitos teóricos e a permitir que estudantes e pesquisadores possam ter acesso a equipamentos, muitas vezes complexos e caros, localizados em diferentes instituições de ensino. Gravier et al. (2008) relata diferentes áreas de desenvolvimento de WebLabs abordadas pela comunidade científica, tais como: hidráulica, mecânica dos fluídos, mecatrônica, astronomia e redes de computadores, sendo que o maior número de publicações se concentra nos campos da automação, eletrônica, física e robótica. Esta flexibilidade de utilização abre um canal de comunicação entre os usuários, promovendo novas parcerias e o desenvolvimento de projetos conjuntos em diferentes comunidades científicas, tais como aqueles reportados em Andria (2007), Maga (2008), Cmok (2009), TIDIA-KyaTera (2011) e Alliet-Gaubert et al. (2012).

Atualmente diversas universidades espalhadas pelo mundo estão desenvolvendo laboratórios remotos de modo a ampliar e tornar mais versátil sua capacidade de ensino, para que – idealmente – em qualquer tempo e em qualquer lugar seus alunos possam praticar as teorias adquiridas em sala de aula, tornando-se assim mais competitivos em um mercado global. Como referências de projetos em estágio adiantado de desenvolvimento, tem-se o WebLab Deusto, da Universidade de Deusto, na Espanha, o Remotelab, na Faculdade de Engenharia da Universidade do Porto, em Portugal, e o iLabs, no Massachusetts Institute of Technology (MIT), nos Estados Unidos.

Um estudo comparativo entre os laboratórios reais, virtuais e remotos pode ser encontrado em Corter et al. (2004) e Nedic et al. (2003), conforme Tabela 1, onde são abordadas as principais vantagens e desvantagens acerca do uso destes três tipos de laboratórios .

Tabela 1 - Lista Comparativa das Vantagens e Desvantagens entre Laboratórios Reais, Virtuais e Remotos.

Adaptado de Nedic et al. (2003).

<b>LISTA COMPARATIVA SOBRE AS VANTAGENS E DESVANTAGENS ENTRE LABORATÓRIOS REAIS, VIRTUAIS E REMOTOS</b>		
<b>TIPO DO LABORATÓRIO</b>	<b>VANTAGENS</b>	<b>DESVANTAGENS</b>
<b>Real</b>	<ul style="list-style-type: none"> <li>- Dados realísticos</li> <li>- Interação com equipamento real</li> <li>- Trabalho colaborativo</li> <li>- Interação com o supervisor</li> </ul>	<ul style="list-style-type: none"> <li>- Restrições de tempo e lugar</li> <li>- Exigem agendamento</li> <li>- Caros</li> <li>- Necessidade de supervisão</li> </ul>
<b>Virtual</b>	<ul style="list-style-type: none"> <li>- Bom para explicação conceitual</li> <li>- Sem restrição de tempo e lugar</li> <li>- Média interatividade</li> <li>- Custo baixo</li> </ul>	<ul style="list-style-type: none"> <li>- Dados idealizados</li> <li>- Falta de Colaboração</li> <li>- Nenhuma interação com o equipamento real</li> </ul>
<b>Remoto</b>	<ul style="list-style-type: none"> <li>- Interação com equipamento real</li> <li>- Calibração</li> <li>- Dados realísticos</li> <li>- Sem restrição de tempo e lugar</li> <li>- Custo médio</li> </ul>	<ul style="list-style-type: none"> <li>- Apenas "presença virtual" no laboratório</li> </ul>

## 1.2 DESENVOLVIMENTOS RECENTES NA ÁREA

Os WebLabs são idealizados de modo a permitir a realização de experimentos remotamente sem a necessidade de acompanhamento de um instrutor ou auxiliar de laboratório. Zubia et al. (2009) abordam aspectos relevantes a se considerar durante o desenvolvimento de laboratórios remotos, tais como

mantenabilidade, escalabilidade e segurança, e realizam uma comparação entre a arquitetura XML e outras arquiteturas existentes. Santos (2013) apresenta uma proposta de um modelo de referência para o desenvolvimento de projetos de WebLabs.

Diversos trabalhos sobre o desenvolvimento de WebLabs tem sido relatados na última década. Pereira et al. (2008) apresentam o uso de sistemas hápticos em laboratórios remotos como recurso para enriquecer experimentos cuja compreensão é facilitada pela percepção de propriedades táteis. Pradarelli et al. (2009) tratam do acesso remoto a equipamentos de testes de circuitos integrados (CIs), com fins educativos. Gomes et al. (2009) abordam a experimentação remota aplicada ao ensino de lógica digital. Kaminski et al. (2009) relatam o sucesso do emprego de um laboratório remoto para o ensino de técnicas de Modulação por Largura de Pulso (PWM) em conversores de tensão trifásicos. Zalewski et al. (2009) discutem uma abordagem para o projeto e implementação de laboratórios remotos voltados ao ensino de engenharia de software com foco em sistemas embarcados e sistemas de tempo real, tais como microcontroladores e placas FPGA (*Field Programmable Gate Array*).

No trabalho apresentado por Brito et al. (2009) tem-se o caso do projeto e desenvolvimento de um experimento remoto para controlar o nível de um sistema de dois tanques. O experimento é controlado por meio de um controlador On-off e de um algoritmo PID, e monitorado através de um ambiente customizado através da plataforma de desenvolvimento de software LabVIEW™. Em Wolbank et al. (2009) tem-se a descrição do projeto de um laboratório remoto aplicado aos estudos da eletrônica de potência e do controle de movimento. O WebLab foi projetado de modo a permitir: a identificação do modelo estudado (servo motor), o projeto das malhas de controle de velocidade e de posição, e a sintonia dos parâmetros de controlador. Khalil et al. (2009) utilizaram a plataforma LabVIEW para implementar um laboratório remoto voltado ao desenvolvimento de experimentos realizados em cursos de engenharia elétrica da Universidade do Qatar. O WebLab permite que o usuário realize dois experimentos: levantamento da curva de condução de um diodo e estudo das características de um pêndulo invertido. Yusof (2009) apresenta o desenvolvimento de um laboratório remoto aplicado ao ensino de controle de

processos. A estrutura implementada permite que o usuário realize simulações e experimentos em tempo real utilizando como técnicas de controle a lógica *fuzzy*, o controle PID e o controle MIMO (*multi-input multi-output system*).

No *Massachusetts Institute of Technology* (MIT), os laboratórios operados remotamente são denominados "iLabs" e foram desenvolvidos principalmente para o estudo e caracterização de dispositivos microeletrônicos, tais como diodos e transistores. Do lado do cliente (usuário) utiliza-se um navegador Web padrão, compatível com Java. O sistema permite, através de um gerenciamento de fila, que múltiplos usuários executem diferentes experimentos de forma coordenada. Este laboratório remoto tem sido utilizado no ensino de assuntos relativos a dispositivos eletrônicos dos cursos de graduação e pós-graduação do MIT, bem como de outras instituições com concessão de acesso.

O laboratório remoto NetLab, desenvolvido pela *University of South Australia*, foi criado com o objetivo de permitir ao aluno o acesso remoto ao experimento por meio de qualquer computador conectado à internet durante 24 horas por dia. A ideia é permitir que os alunos realizem experimentos relacionados à área de eletrônica de modo a se sentirem tão próximos quanto possível de um laboratório real. Para isto, o sistema possui um software configurador de circuitos que permite a alteração do diagrama elétrico e a seleção dos componentes a serem utilizados, aumentando assim a versatilidade e conservando uma analogia com a forma original de se trabalhar no laboratório.

O laboratório de experimentação remota RExLab, na Universidade Federal de Santa Catarina (UFSC), conta com uma rede de 12 Universidades (RexNet) em 5 diferentes países tendo como um de seus objetivos atender a necessidade de apropriação social da ciência e da tecnologia, popularizando conhecimentos científicos e tecnológicos.

Dentre outros os artigos e dissertações recentes e relevantes sobre WebLabs, pode-se citar: Gravier (2008), Carnevali (2003), Rocha (2009), Ngolo (2009), Farias (2008), Paladini (2008), Moreira (2008), Cruz (2007).

A pesquisa bibliográfica abordou ainda diversas aplicações dos WebLabs com o intuito de identificar as características comuns e relevantes para o desenvolvimento destes sistemas. Características como acessibilidade,

operacionalidade e segurança são recorrentes em sistemas já implantados e disponíveis na internet, tais como: iLAB (MIT), WEBLAB-DEUSTO (Universidade de Deusto na Espanha), REMOTELAB (Faculdade de Engenharia da Universidade do Porto) e Projeto GigaBOT (Universidade Estadual de Campinas, Brasil) e NETLAB (University of South Australia).

### 1.3 MÉTODOS E FERRAMENTAS NO DESENVOLVIMENTO DE PRODUTOS E SISTEMAS TÉCNICOS

Encontrar boas soluções para desenvolvimento de produtos é uma tarefa que pode ser melhor conduzida com o uso de um procedimento sistematizado de trabalho que seja planejável, flexível e que permita otimizações. Uma metodologia de trabalho inclui procedimentos para interligação de etapas de trabalho e fases do projeto que, de maneira flexível, são adaptados ao respectivo problema e objetivam incentivar a criatividade e habilidades individuais das pessoas envolvidas no processo de desenvolvimento. Assim como referenciado em Pahl & Beitz (2005), o emprego de uma metodologia de trabalho tende a minimizar o caráter subjetivo presente nas questões relativas ao problema que se quer resolver, minimizando a dependência de soluções encontradas ao acaso ou em ciclos de tentativa e erro, facilitando o planejamento e o controle do trabalho em equipe num processo integrado e multidisciplinar de geração de um produto. Alguns exemplos de propostas metodológicas para o desenvolvimento de sistemas técnicos, bem como de novos produtos, são: Pahl & Beitz (1996), VDI 222 (1985), Asimov (1962), ASME(1986), Back (1983) e Back (2008).

Segundo a abordagem sistemática proposta por Pahl & Beitz (1996), o processo de projeto divide-se em 4 etapas que são resumidas a seguir :

- clarificação das tarefas: etapa de coleta dos requisitos de projeto e definição das restrições associadas, resultando na definição do problema a ser resolvido;
- projeto conceitual: etapa de determinação da estrutura funcional e de busca por princípios de solução, resultando em um conceito;
- projeto preliminar: etapa onde se desenvolve o leiaute (configuração e montagem) do produto segundo critérios técnicos e econômicos;
- projeto detalhado: etapa de documentação do projeto.

Pahl & Beitz (2005) destacam que para garantir a eficácia de uma metodologia esta deve ser auxiliada por medidas organizacionais, devendo a equipe de projeto estar preparada para trabalhar de forma sistematizada.

Back (2008) apresenta um modelo de referência conhecido como Processo de Desenvolvimento Integrado de Produtos (PRODIP) proposto com base em pesquisas e experimentos realizados pelo Núcleo de Desenvolvimento Integrado de Produtos (NeDIP / UFSC). O PRODIP apresenta três macros fases: planejamento do projeto; elaboração do projeto do produto e planejamento da implementação. A macro fase de elaboração do produto é decomposta em: projeto informacional; projeto conceitual; projeto preliminar e projeto detalhado. A macro fase de implementação é decomposta em: fase de preparação da produção, fase do lançamento e fase da validação do produto.

Uma vez que os modelos metodológicos existentes tendem a ter um caráter bastante generalista, Mendes et al. (2010) propuseram um modelo com prescrições mais profundas direcionadas a sistemas automatizados de teste, que com algumas adaptações pode ser utilizado na abordagem de desenvolvimento de WebLabs, resultando em um modelo direcionado especificamente para este tipo de sistema. Este modelo, que será referenciado nesta dissertação com a sigla PDW, é também derivado da abordagem sistemática de Pahl & Beitz (PBSA), sendo composto por seis etapas, conforme detalhado na Figura 1. As duas primeiras etapas deste modelo abordam um planejamento conciso do projeto, através das quais se obtém um conjunto de especificações técnicas, com base em requisitos qualitativos e quantitativos, enquanto as etapas seguintes preveem a síntese progressiva de uma solução integrando três camadas, denominadas: sistema físico, hardware e software (que representam a característica de uma engenharia integrada).

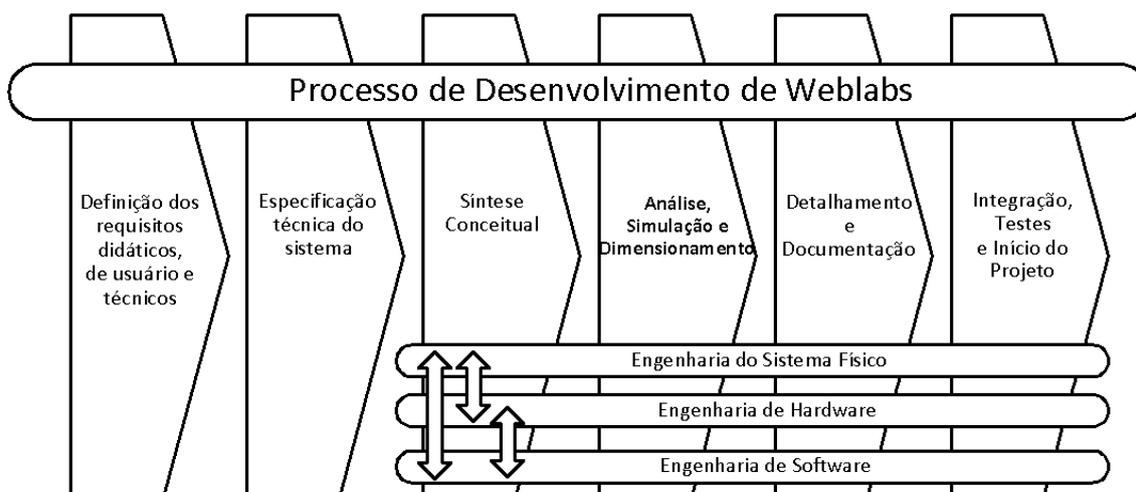


Figura 1– Estágios do Processo de Desenvolvimento.

Fonte: Mendes et al. (2010).

Segundo Mendes et al. (2010), o modelo proposto auxilia na organização dos elementos presentes no desenvolvimento de uma arquitetura de WebLab logo nas etapas iniciais, reduzindo assim a necessidade de retrabalho ou até mesmo profundas modificações no projeto por conta de um planejamento mal elaborado. A consideração das fontes de requisitos presente no modelo auxilia na condução do trabalho, de modo a reduzir erros devidos à tomadas de decisão equivocadas ou a não consideração de elementos relevantes, eventualmente por inexperiência, pela equipe de desenvolvimento.

Mendes et al. (2010) sugerem algumas ferramentas que podem ser úteis, tais como a matriz da “Casa da Qualidade” do método QFD (*Quality Function Deployment*), traduzido para o português como “Desdobramento da Função Qualidade” (ver Carnevali et al. (2003)), a Análise Funcional (ver Back (2008)), a Matriz Morfológica (ver Cross (1994)) e o método de Pugh (ver Pugh (1991)).

O método QFD surgiu no final década de 60, em meio à franca expansão das indústrias japonesas, conforme formulado pelos professores Akao e Mizuno, encaixando-se no contexto do controle de qualidade total (TQC). A partir de sua introdução o método sofreu o acréscimo de vários elementos conceituais e metodológicos, resultando em um modelo amplo do QFD. Segundo Cheng (2007), o QFD pode ser conceituado como “uma forma de comunicar sistematicamente

informação relacionada com a qualidade e de explicitar ordenadamente trabalho relacionado com a obtenção da qualidade, que tem como objetivo alcançar o enfoque da garantia da qualidade durante o desenvolvimento de produto e é subdividido em Desdobramento da Qualidade (QD) e Desdobramento da Função Qualidade no sentido restrito (QFDr).” O QD tem por objetivo desdobrar a qualidade através da lógica da causa e efeito, de forma estruturada, hierarquizada e priorizada, e o QFDr é o desdobramento da função trabalho. Para Ohfujii et al. (1997), o QFD é uma metodologia de desenvolvimento de produto ou serviço, por meio da qual pode-se traduzir as exigências e expectativas dos clientes em características mensuráveis, garantindo a qualidade a partir da fase inicial do projeto. Carnevali et al. (2003) cita algumas vantagens no uso do QFD: melhoria da comunicação e relação entre os departamentos envolvidos; redução do tempo de desenvolvimento e do número retrabalho em projetos; redução dos custos e aumento da satisfação dos clientes.

A Casa da Qualidade, do inglês *House of Quality* (HOQ), é uma técnica do QFD muito utilizada para o planejamento sistematizado de produto (serviço), consistindo de uma matriz que contém a relação entre os requisitos de clientes e os meios para obtê-los, Cmok et al. (2009).

Cmok et al. (2009) identificou os requisitos para sistemas de aprendizagem aplicados em sistemas de e-learning (RL-Miracle). Santos et al. (2010) apresentaram uma proposta, direcionada a um experimento em WebLab relacionado a uma planta de processo térmico, que permite selecionar e desdobrar requisitos de usuários e requisitos de projeto tendo como base as características técnicas e as necessidades dos usuários.

A seguir serão descritas as seis etapas do modelo PDW.

#### 1.4 TEMA E QUESTÃO DE PESQUISA

O tema desta dissertação contempla o modelo sistemático para o planejamento e desenvolvimento de WebLabs proposto por Mendes et al. (2010), PDW, aplicado ao caso de projeto de um WebLab para uma planta de processo térmico. O tema está inserido na linha de pesquisa Concepção de Produtos e Sistemas do PPGEPS.

*Questão de pesquisa:* O modelo PDW cobre suficientemente bem as particularidades e requisitos que devem ser considerados no desenvolvimento de um WebLab, ou pode ainda incorporar contribuições advindas de experiências e lições aprendidas em estudos de caso?

## 1.5 OBJETIVOS

### 1.5.1 Objetivo geral

Desenvolver o estudo de caso correspondente ao projeto e execução de um WebLab para uma planta de temperatura, e identificar possíveis contribuições para o modelo proposto por Mendes et al. (2010).

### 1.5.2 Objetivos específicos

- Planejar o WebLab para a planta de processo térmico com base no modelo de Mendes et al. (2010);
- Implementar o WebLab, observando as limitações de capacidade física de realização ditadas pela disponibilidade efetiva de recursos;
- Avaliar os resultados obtidos ante o conjunto inicial de especificações de projeto;

## 1.6 JUSTIFICATIVA

O desenvolvimento de um WebLab está ligado ao contexto do experimento e aos objetivos didáticos envolvidos. Sua complexidade pode variar de sistemas simples a sofisticados, e de acordo com as características e requisitos de projetos necessários a sua implementação. Encontrar as soluções adequadas para tal desenvolvimento é uma tarefa melhor conduzida com o uso de um modelo de referência para orientar as etapas de concepção e desenvolvimento. Segundo Mendes et al. (2010), atender aos requisitos para o desenvolvimento de um projeto de WebLab compreende, de forma integrada, criar um sistema que contempla três diferentes domínios, ou camadas: sistema físico, hardware e software. Ainda, segundo Santos et al. (2010), o processo de desenvolvimento de um WebLab

assemelha-se ao desenvolvimento de projetos especiais, como os de máquinas e equipamentos, em oposição aos sistemas fabricados em série ou produtos de massa. Lidar com as incertezas iniciais de um projeto desta natureza pode não ser uma tarefa muito clara para uma equipe pouco experiente. Ademais, processos mal conduzidos tendem a gerar retrabalhos que implicam em novos custos e maior tempo de desenvolvimento. A priori, quanto maior a compreensão sobre o problema de pesquisa, logo nos estágios iniciais de desenvolvimento, menores serão os riscos citados acima. Desta forma, a adoção de uma sistematização provida por uma metodologia de projeto tende a reduzir o caráter empírico de desenvolvimento, de modo a estruturar, orientar e integrar as etapas de construção e implementação de tais sistemas.

Considerando-se que até o momento não foram encontrados na literatura trabalhos reportando o emprego de processos sistemáticos para o desenvolvimento de WebLabs, o presente trabalho posiciona-se sobre essa lacuna e enquadra-se como um estudo de caso da aplicação do modelo PDW. Com a adoção deste processo sistemático, são esperados benefícios tais como menor tempo de desenvolvimento e menor quantidade de loops de projeto. Os resultados obtidos com a aplicação da referida abordagem sistematizada no presente caso poderão servir para propor melhorias do modelo utilizado.

Com base no exposto acima, justifica-se a realização do presente projeto tanto pela realização de um sistema que será efetivamente utilizado para os fins a que se propõe, como pela possibilidade de uso das experiências e lições aprendidas no refinamento de um modelo de referência para o projeto de sistemas semelhantes.

## 1.7 SOBRE METODOLOGIA DE PESQUISA EM TRABALHOS DE NATUREZA APLICADA

Minayo (1993) descreve a pesquisa como “atividade básica das ciências na sua indagação e descoberta da realidade. É uma atitude e uma prática teórica de constante busca que define um processo intrinsecamente inacabado e permanente. É uma atividade de aproximação sucessiva da realidade que nunca se esgota, fazendo uma combinação particular entre teoria e dados”. Na condução desta pesquisa, de natureza aplicada, será utilizada uma abordagem qualitativa, uma vez

que os requisitos envolvidos têm, muitas vezes, conteúdo de ordem subjetiva. Do ponto de vista do objetivo envolvido, neste caso exploratório, pretende-se proporcionar maior familiaridade com o tema tornando-o explícito à luz do experimento. Gil (1991) caracteriza os objetivos de uma pesquisa como exploratórios, descritivos ou explicativos. Yin (2001) descreve-os como explanatórios, descritivos ou exploratórios quando relacionados a estudos de casos. Quanto aos procedimentos técnicos, adotou-se o estudo de caso como forma de verificar, dentro do objeto de estudo, quais variáveis ou fatores são capazes de influenciá-lo e os efeitos destas sobre o mesmo. Ainda para Gil (1991), os aspectos técnicos de uma pesquisa podem ser: Pesquisa Bibliográfica; Pesquisa Documental; Pesquisa Experimental; Levantamento; Estudo de caso; Pesquisa Expost-Facto; Pesquisa-Ação e Pesquisa Participante. Assim, a pesquisa dar-se-á com a aplicação do modelo proposto por Mendes et al. (2010) na elaboração e construção de um WebLab para uma planta de processo térmico, levando em conta as restrições físicas e orçamentárias existentes. A avaliação será feita de forma qualitativa, confrontando os resultados obtidos com os requisitos de projeto levantados à priori, de modo a apontar quais características foram relevantes ao processo de desenvolvimento, e se existem ou não possíveis melhorias que possam levar à otimização do modelo e de que modo podem ser implementadas. Não se pretende validar o modelo com este único estudo de caso, uma vez que as prescrições deste modelo devem ser adaptadas à natureza de cada projeto, podendo encontrar maior ou menor aplicabilidade conforme o caso; logo, somente uma parte destas prescrições encontrará aplicação neste estudo.

## 1.8 ESTRUTURAÇÃO DAS ETAPAS DO TRABALHO

Este trabalho contém três capítulos, enumerados e sintetizados a seguir:

Capítulo 1 – Introdução

Capítulo 2 – Metodologia

Capítulo 3 – Desenvolvimento do WebLab

Capítulo 4 – Resultados e Discussão

Capítulo 5 – Conclusão e Considerações Finais

O primeiro capítulo apresenta uma introdução ao tema de pesquisa de modo a contextualizá-lo e justificá-lo mediante o uso de uma metodologia de pesquisa e discorre sobre métodos de desenvolvimento de produtos / sistemas técnicos comumente encontrados na literatura.

O segundo capítulo apresenta os detalhes do modelo sistemático de desenvolvimento de WebLabs (PDW), proposto por Mendes et al. (2010).

O terceiro capítulo descreve o desenvolvimento do WebLab para a planta de processo de temperatura com a aplicação das etapas do modelo PDW, e apresenta os testes de utilização do sistema.

No quarto capítulo discutem-se os resultados obtidos com o sistema WebLab, com base nas especificações inicialmente propostas.

O quinto capítulo, de conclusão e considerações finais, apresenta as contribuições da dissertação.

## **2 METODOLOGIA**

Os tópicos seguintes correspondem a uma revisão das etapas do modelo PDW, neste trabalho empregado no projeto e execução de um Weblab interativo para uma planta de processo térmico.

### **2.1 DEFINIÇÃO DOS REQUISITOS TÉCNICOS, DIDÁTICOS E DE USUÁRIO**

Nesta primeira etapa busca-se definir os requisitos do projeto por meio de um levantamento. Tais requisitos estão divididos em três grupos: didáticos, de usuário e técnicos, sendo que na Tabela 2 são apresentadas fontes para sua busca, ou seja, aspectos que ao serem examinados, podem sugerir elementos e ideias que levem à definição de requisitos. A sua elaboração tem caráter multidisciplinar, devendo envolver os pontos de vista do usuário e da equipe de desenvolvimento. Os requisitos compreendem informações de caráter qualitativo.

Tabela 2 - Fontes de Requisitos em Projetos de WebLabs.

Adaptado de Mendes et al. (2010).

GRUPOS	FONTES DE REQUISITOS
Didático	<ul style="list-style-type: none"> <li>- Conhecimentos sobre o experimento</li> <li>- Contexto e objetivos do experimento</li> <li>- Sequenciamento didático</li> <li>- Comandos e cognição resultante</li> <li>- Resultados esperados do experimento</li> </ul>
Usuário Final	<ul style="list-style-type: none"> <li>- Interface cognitiva</li> <li>- Visualização da dinâmica do sistema físico</li> <li>- Modos e elementos de interação</li> <li>- Orientação de ações do usuário</li> <li>- Performance do sistema constante (e suficiente)</li> <li>- Opções de linguagem</li> <li>- Acesso Multi-plataforma</li> <li>- Compatibilidade com outros sistemas de e-learning</li> </ul>
Técnico	<ul style="list-style-type: none"> <li>- Configuração remota do experimento</li> <li>- Restrições do laboratório</li> <li>- Procedimentos e normas de segurança aplicáveis</li> <li>- Formatação e apresentação dos dados do experimento</li> <li>- Calibração, manutenção e testabilidade</li> <li>- Escalabilidade e capacidade de upgrade</li> <li>- Manutenção (local e remota)</li> <li>- Paralelismo de tarefas</li> <li>- Disponibilidade de horário e de agendamento</li> <li>- Gerenciamento de acesso e permissões de usuário</li> <li>- Suporte à simultaneidade de usuários</li> <li>- Suporte ao trabalho colaborativo</li> <li>- Intervenções técnicas locais</li> </ul>

## 2.2 ESPECIFICAÇÕES TÉCNICAS DE PROJETO

A Tabela 3 exibe um conjunto de fontes de especificações técnicas de projeto aplicáveis, conforme a proposta do PDW. Especificações técnicas de projeto devem expressar quantitativamente a performance desejada e não devem ter nenhum compromisso com qualquer solução em particular. Salienta-se que este conjunto de fontes, que podem ser usadas para sugestionar especificações (de modo semelhante à obtenção dos requisitos, no item 2.1), não é fechado ou definitivo, devendo-se observar criticamente o contexto de cada projeto. Ao final do projeto, espera-se que a performance do sistema seja compatível com as especificações técnicas projetadas.

Tabela 3 - Fontes Comuns de Especificações Técnicas em Projetos de WebLabs.

Adaptado de Mendes et al. (2010).

FONTES DE ESPECIFICAÇÕES TÉCNICAS
Largura de banda de conexão
Recursos de computadores locais e remotos
Medidas de qualidade de videoconferência
Intervalos variáveis e taxas de atualização
Incertezas admissíveis para sinais medidos e de saídas
Limitações de experimentos e prazos de execução
Usuários simultâneos, independentes ou colaborativos
Restrições e limitações do sistema físico
Consumo de energia
Segurança geral
Disponibilidade do WebLab

### 2.3 SÍNTESE CONCEITUAL

A Figura 2, tomada do PDW, mostra uma macroestrutura tipicamente encontrada em um WebLab, detalhando as relações existentes entre as camadas de software, hardware e sistema físico. Esta composição permite que se tenha uma visão geral das soluções necessárias ao estabelecimento da interconexão entre usuário e experimento, de modo a auxiliar na determinação da estrutura funcional do WebLab que se quer implementar.

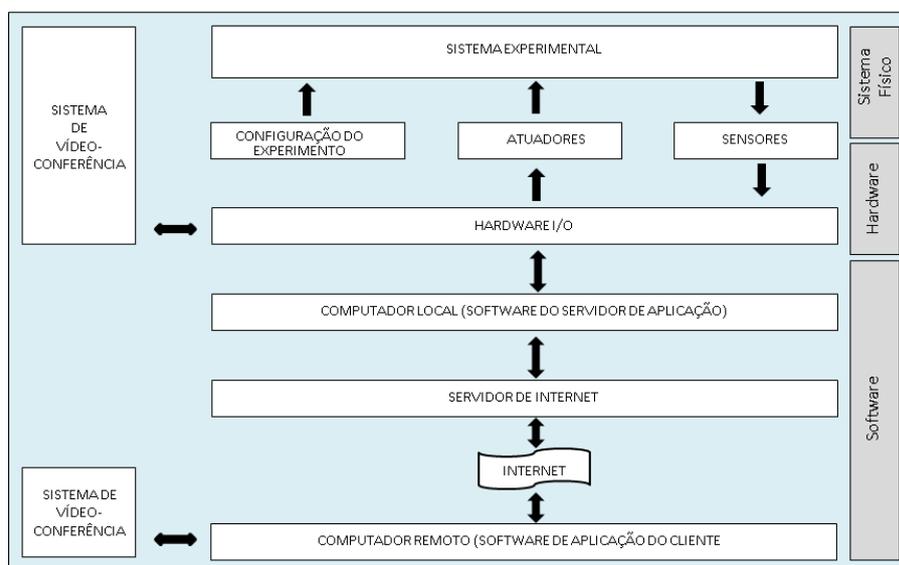


Figura 2 – Macroestrutura para WebLabs.

Adaptado de Mendes et al. (2010).

De acordo com Back (2008), o método da síntese funcional trata-se de um dos métodos de geração de concepções mais citados nas literaturas, tendo sua origem na década de 1970. A síntese funcional permite que se extraiam funções essenciais ao desenvolvimento de produtos, dispositivos ou sistemas durante a abordagem de um problema. As funções podem ser divididas em sub-funções que caracterizam os limites de decisão da abordagem do problema. A essência do método tem foco em estabelecer as características desejadas em um novo projeto e não como obtê-las (Cross, 1994).

A macroestrutura exibida na Figura 2 é um ponto de partida para a definição da estrutura funcional do sistema, em uma hierarquia de funções e sub-funções. Tipicamente, a nomenclatura de funções é feita através de uma expressão contendo uma combinação verbo+substantivo.

As sub-funções, ou funções parciais, ensejam a busca por soluções parciais que lhes atendam individualmente, formando um leque de opções. A seguir, observadas as condições para compatibilidade, são integradas de modo a compor a solução geral do sistema. Esta visão geral de possibilidades de combinação de soluções parciais é possível com o emprego da chamada Matriz Morfológica. Como resultado tem-se soluções conceituais alternativas, elaboradas através da combinação de princípios de solução encontrados para cada função.

Estas soluções conceituais alternativas podem então ser ordenadas em termos de escores de mérito através do método de Pugh, usando um conjunto de critérios ponderados, que pode ser tomado, por exemplo, da matriz do QFD elaborada na etapa inicial. O método de Pugh, criado por Stuart Pugh na década de 90 (ver Pugh (1991)), é um método que compara diferentes conceitos com um conceito de referência, dando objetividade à escolha da melhor opção – no caso, melhor solução conceitual alternativa (geral) para o sistema WebLab.

Diversas funções que podem ser usadas como ponto de partida no estabelecimento das funções principais presentes em um WebLab, tomadas do PDW, são listadas na Tabela 4. Não obstante, estas podem/devem ser desmembradas em funções parciais e oportunamente acrescidas de outras pertinentes ao contexto do projeto.

Tabela 4 - Funções de Primeira Ordem no Desenvolvimento de WebLabs.

Adaptado de Mendes et al. (2010).

CAMADAS	FUNÇÕES DE NÍVEL SUPERIOR
Sistema Físico	<ul style="list-style-type: none"> <li>- (Prover) Sustentação mecânica/Posicionamento dos sensores/atuadores e outros dispositivos de interação</li> <li>- (Prover) Fornecimento de energia</li> <li>- (Prover) Fornecimento de materiais/substâncias</li> <li>- (Prover) Sistema de proteção contra potenciais riscos</li> <li>- (Prover) Tratamento (manual / automático) de emergências</li> <li>- Posicionar a tomada visual do sistema físico</li> </ul>
Hardware	<ul style="list-style-type: none"> <li>- Condicionar e transmitir os sinais dos sensores</li> <li>- (Prover) Amplificação e saída dos sinais de atuação</li> <li>- (Prover) Reprodução, captura e transmissão de áudio e vídeo ao vivo</li> <li>- Exibir o estado do sistema localmente e remotamente</li> <li>- (Prover) Capacidade computacional para executar os softwares do servidor local</li> </ul>
Software	<ul style="list-style-type: none"> <li>- Inicializar e configurar os sistemas</li> <li>- Gerenciar os sinais de I/O com componentes de hardware e instrumentos de interface</li> <li>- Executar algoritmo(s) de experimento(s)</li> <li>- Gerenciar a exibição, armazenamento e recuperação dos dados do experimento</li> <li>- Transformar os dados adquiridos em informações significativas</li> <li>- Gerenciar a transferência de dados via Web</li> <li>- Calibrar sensores</li> <li>- Garantir operação segura</li> </ul>

## 2.4 ANÁLISE, SIMULAÇÃO E DIMENSIONAMENTO

Esta é a etapa de refinamento da solução conceitual obtida, que deve passar por um processo de análise, simulação e dimensionamento, para que se localizem as otimizações cabíveis nas camadas de hardware, software e sistema físico. Conhecimentos e ferramentas dos campos da engenharia são aplicáveis nesta etapa, considerando-se as áreas tecnológicas presentes. A construção de um protótipo (piloto) pode de auxiliar nestas tarefas. Eventualmente, os resultados obtidos nesta fase podem aprimorar o conceito definido na fase anterior, em um ou mais loops de projeto.

## 2.5 DETALHAMENTO E DOCUMENTAÇÃO

Esta etapa tem por finalidade preparar e organizar a documentação e o detalhamento de todo o projeto, de modo a garantir sua correta execução e facilitar a manutenção e readaptação do sistema quando necessário.

## 2.6 INTEGRAÇÃO, TESTE E INICIALIZAÇÃO

Nesta última etapa passa-se à implementação dos subsistemas envolvidos na construção do WebLab. Os mesmos devem ser testados exaustivamente de modo a serem suficientemente confiáveis quanto à operacionalidade e à segurança.

## 3 DESENVOLVIMENTO DO WEBLAB

Este capítulo detalha as etapas de desenvolvimento do WebLab para uma planta experimental de temperatura, segundo o modelo PDW descrito anteriormente. Para a realização do projeto, foram projetados acréscimos e adaptações da planta, principalmente com o intuito de viabilizar a operação remota e permitir a realização de experimentos comuns no ensino de engenharia de controle, tais como o ajuste de parâmetros de controladores On-Off / PID e a determinação do modelo dinâmico de um sistema, neste caso térmico. Ao final deste capítulo são apresentados os testes de utilização do sistema implementado.

### 3.1 DESCRIÇÃO DA PLANTA DE TEMPERATURA

O do WebLab desenvolvido neste trabalho teve como base a planta didática 'Estação de Processo de temperatura LabVolt - Modelo 3504', existente no Laboratório Controle de Processos (LAS-8) da Pontifícia Universidade Católica do Paraná, do curso de Engenharia de Controle e Automação. Esta planta, exibida na Figura 3, compreende um forno elétrico com faixa de operação entre 20 e 200°C (70-400°F) que pode ser operado manualmente (modo On-Off, através do chaveamento de um relé 24 Vdc), ou proporcionalmente (por meio de um sinal de entrada de 4 a 20mA que opera a resistência de aquecimento). Mudanças nas características da dinâmica do processo podem ser realizadas através de uma válvula, que permite a injeção em quantidade variável de ar comprimido no interior do forno. A geração de distúrbios pode ser feita manualmente mediante o ajuste (manual) da posição de um *dampner* localizado em uma das paredes do forno. A instrumentação de processo é composta por um transmissor de temperatura para termopar tipo J, um termômetro de bulbo capilar e um transmissor de temperatura

para sensor RTD tipo Pt-100. Ambos os transmissores operam com sinal de saída na faixa 4 a 20mA. A configuração da planta, conexão dos equipamentos e sinais de controle que depende da atividade a ser executada, é realizada através de conectores tipo banana localizados no painel de controle principal. A planta dispõe de um CLP e de um registrador gráfico em rolo de papel, que não serão utilizados.

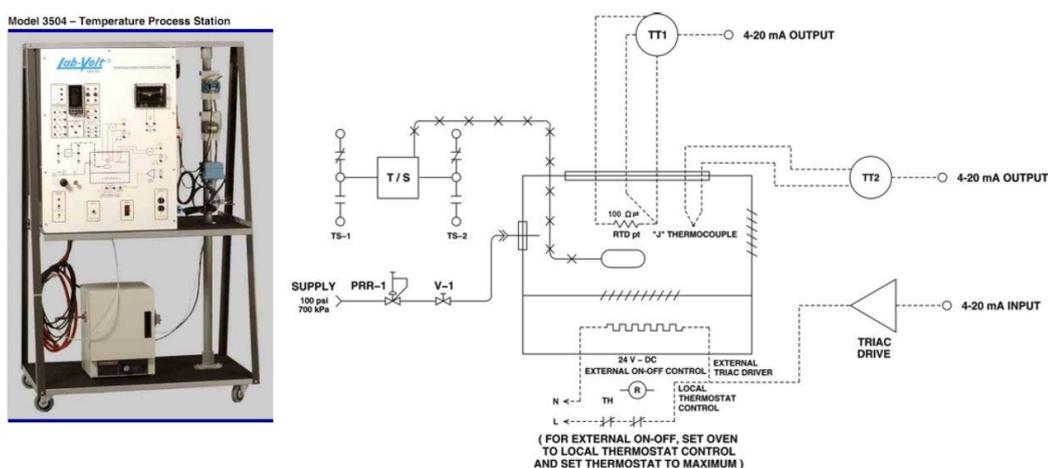


Figura 3 – Figura e diagrama da Estação de Processo de Temperatura LabVolt - Modelo 3504.

Fonte: Lab-Volt®.

## 3.2 REQUISITOS E ESPECIFICAÇÕES

### 3.2.1 Atividades didáticas propostas para o WebLab

As rápidas mudanças e avanços tecnológicos ocorridos na área de instrumentação de processos têm aumentado a variedade de tarefas executadas pelos instrumentistas responsáveis pelo projeto, implantação, manutenção e operação das plantas industriais. Tarefas como parametrizar e sintonizar controladores de processo podem se tornar um grande desafio frente às diferentes condições de uso e operação. O controlador é o subsistema que gera um sinal de saída dependente do erro ou desvio entre o sinal de entrada advindo do processo medido e o valor desejado (*set point*). O sinal de saída é enviado ao elemento final de controle que regula o processo de modo a eliminar o erro. O algoritmo de controle

deve ser executado em uma plataforma computacional adequadamente dimensionada.

Para este projeto, foram selecionadas três atividades didáticas relacionadas com a engenharia de sistemas de controle: controle de temperatura On-Off, controle PID (proporcional, integral, derivativo) e identificação do modelo dinâmico da planta de processo de temperatura. Estas atividades têm por objetivo oportunizar ao aluno a exploração na prática de conceitos de um sistema de controle em malha fechada de processo de temperatura, determinando a dinâmica do processo e variando estratégias de controle.

- **ATIVIDADE DIDÁTICA 1: CONTROLE DE TEMPERATURA ON-OFF**

*Objetivo:* Experimentar o controle do sistema com atuação a duas posições.

*Discussão:* Um controlador On-Off, ou controlador a duas posições, tem apenas dois estados de operação: ligado ou desligado. Por serem relativamente simples e baratos, os controladores de duas posições são comumente usados na indústria de processos. Sua maior limitação é a característica inerente de ciclagem contínua, tornando-os adequados a processos lentos, tais como os que necessitam de controle de temperatura.

*Procedimentos do usuário:*

1. Ajustar a posição do damper (janela variável do forno);
2. Ajustar o fluxo de ar comprimido;
3. Iniciar a gravação dos dados;
4. Ajustar o set point de temperatura;
5. Selecionar modo de controle On-Off;
6. Verificar através do indicador de temperatura se o forno é mantido na temperatura desejada;
7. Variar a carga térmica do sistema;
8. Avaliar as mudanças ocorridas no ciclo de trabalho On-Off;

9. Finalizar gravação dos dados;

- ATIVIDADE DIDÁTICA 2: CONTROLE PID (PROPORCIONAL, INTEGRAL, DERIVATIVO)

*Objetivos:* Sintonizar um controlador PID para que tenha bom desempenho; analisar os efeitos decorrentes de mudanças de carga e de *set point* sobre a variável controlada.

*Discussão:* Entender como um processo reage a mudanças de carga e distúrbios permite projetar melhor um sistema de controle para o processo. Um método para observar como os ajustes do controlador afetam o comportamento da variável de processo é o de se gerar um distúrbio intencional no processo e monitorar a resposta do sistema para cada variação individual realizada nos parâmetro PID do controlador. Desta forma pode-se observar, através dos dados armazenados, como a mudança dos parâmetros PID do controlador interfere no controle do processo. O controle PID é amplamente utilizado na indústria.

*Procedimentos do usuário:*

1. Ajustar a posição do damper;
2. Ajustar a abertura da válvula de ar comprimido de modo a injetar uma carga pequena;
3. Iniciar a gravação dos dados;
4. Ajustar o *set point* de temperatura;
5. Selecionar o modo de controle Manual;
6. Ajustar manualmente a saída do controlador (variável manipulada MV) até que a temperatura no interior do forno (variável do processo PV) se iguale a referência de temperatura desejada (*setpoint*).;
7. Selecionar o modo de controle PID;
8. Aguardar o sistema estabilizar;
9. Gerar um distúrbio no processo, aumentando o ajuste de abertura da válvula de ar comprimido;

10. Analisar a resposta do controle PID à mudança de carga ocorrida no processo;
11. Reduzir a abertura da válvula de ar e aguardar o processo estabilizar;
12. Finalizar a gravação dos dados;

- ATIVIDADE DIDÁTICA 3: IDENTIFICAÇÃO DO MODELO DINÂMICO DA PLANTA DE PROCESSO DE TEMPERATURA:

*Objetivo:* Esta atividade tem por objetivo familiarizar o aluno com as técnicas utilizadas para realizar o modelamento de um processo.

*Discussão:* Muitas são as publicações sobre a teoria e projeto de controladores PID, dentre as quais pode-se citar Astrom (1995). Como etapas do projeto dos parâmetros de um controlador temos: determinação dos requisitos de desempenho, identificação do processo e análise de malha fechada. Os requisitos de desempenho podem estar definidos em termos de: requisitos no tempo (regime transitório, regime permanente e rejeição a perturbações); requisitos na frequência (largura de faixa) e robustez (estabilidade frente a mudanças no processo). A derivação de uma descrição relevante de um sistema dinâmico a partir de dados observados é chamada de identificação de sistemas e o resultado desta derivação é o modelo do sistema. A análise de malha fechada é a análise da resposta do sistema operando em malha fechada a partir do modelo calculado. Para modelar o sistema pode-se utilizar o método de Ziegler-Nichols (1942) a partir da resposta ao degrau do sistema ou o método dos mínimos quadrados, tal como descrito em Bi et al. (1999).

*Procedimentos do usuário:*

1. Ajustar a abertura do damper;
2. Ajustar a abertura da válvula de ar comprimido de modo a injetar uma carga pequena;
3. Iniciar a gravação dos dados;

4. Selecionar o modo de controle Manual;
5. Aplicar um degrau na saída do controlador ajustando o valor da variável manipulada (MV);
6. Aguarda a estabilização sistema;
7. Finalizar gravação;
8. Identificar o modelo a partir de um dos métodos sugeridos.

### 3.2.2 Determinação dos requisitos técnicos, didáticos e de usuário

A determinação dos requisitos didáticos, técnicos (quantitativos) e de usuário (qualitativos), adotados para o desenvolvimento deste trabalho, teve como base as fontes de requisitos fornecidas por Mendes et al. (2010), considerando:

- Os propósitos didáticos relacionados a três atividades a serem executadas no WebLab;
- Os dados levantados à partir da revisão da literatura relativa às publicações sobre laboratórios remotos relatadas na última década.

A Tabela 5 exibe um resumo dos requisitos técnicos, didáticos e de usuários adotados no desenvolvimento do WebLab para a planta de temperatura.

Tabela 5 - Requisitos didáticos, de Usuário e Técnicos

GRUPOS	REQUISITOS
Didático	<ul style="list-style-type: none"> <li>- Conter procedimentos de orientação ao usuário</li> <li>- Possibilitar a realização do Experimento 1</li> <li>- Possibilitar a realização do Experimento 2</li> <li>- Possibilitar a realização do Experimento 3</li> </ul>
Usuário	<ul style="list-style-type: none"> <li>- Permitir a interação do usuário com o equipamento</li> <li>- Proporcionar a sensação de realismo</li> <li>- Permitir a monitoração e a execução dos comandos</li> <li>- Conter meios (ferramentas) que permitam avaliar o sistema</li> <li>- Ser fácil de usar ou operar</li> <li>- Permitir o acesso a partir do Windows</li> </ul>
Técnico	<ul style="list-style-type: none"> <li>- Permitir a configuração remota do experimento</li> <li>- Não impor limitações físicas que impeçam a manutenção</li> <li>- Realizar o gerenciamento de usuários</li> <li>- Permitir a análise dos dados experimentais</li> <li>- Estar disponível 24h</li> <li>- Permitir a simultaneidade de usuários</li> <li>- Permitir agendamento</li> <li>- Ter arquitetura flexível</li> <li>- Salvar arquivos, gerar relatórios e gráficos</li> <li>- Impor restrições de acesso a parâmetros</li> <li>- Contemplar as normas de segurança</li> </ul>

### 3.2.3 Especificações técnicas de projeto

A Tabela 6 exibe um conjunto de requisitos de projeto, na forma de características quantificáveis. Acrescidos de valores-meta e unidades próprias, constituem as especificações técnicas de projeto.

Tabela 6 - Especificações Técnicas.

ESPECIFICAÇÕES TÉCNICAS	CARACTERÍSTICAS	METAS	MEIO DE AVALIAÇÃO
Quantidade de operações do usuário	Número de etapas para utilizar o sistema considerando: <i>login</i> e senha; escolha do experimento; leitura de instruções; configuração dos parâmetros e execução do experimento	Máximo 6 etapas	Contagem
Quantidade de pontos de visualização por câmeras	Possibilidade de visualizar o experimento por mais de uma câmera ou múltiplos pontos de visualização	1	(Não se aplica)
Quantidade de Pontos de interação	Parâmetros configuráveis e elementos de leitura (indicadores, gráficos, mensagens de texto)	Máx. 30	Contagem
Grau de detalhamento das instruções	Grau de informação dada ao usuário para orientar suas ações durante a realização de uma atividade didática. (Escala de 1 a 3 significando : somente texto; texto e figuras; texto, áudio e vídeo).	Mínimo 1	Inspeção
Instrumentos de coleta de opinião	Número de recursos que permitam o usuário avaliar o sistema. (Questionário on-line, E-mail, Telefone, gravação de voz)	Mínimo 1	Contagem
Número de sistemas operacionais	Acesso multi-plataforma (Windows, Linux, IOS, etc.)	1	(não se aplica)
Espaço local de armazenamento de dados	Tamanho do espaço local disponibilizado para armazenamento de informações geradas pelo usuário durante o experimento	Mínimo 10GBytes	(não se aplica)
Número de repetições do experimento	Quantas vezes o usuário poderá repetir a realização do experimento no período de tempo agendado	Mínimo 1	(não se aplica)
Largura de banda de rede	Quantidade de dados (bits/s) que podem ser transmitidos numa comunicação em rede, em um determinado intervalo de tempo	Mínimo 300 kbps	Medição da velocidade da rede
Índice de disponibilidade do sistema	Tempo que o sistema estará disponível podendo chegar às 24h/dia	58%	(não se aplica)
Número de telas	Número de interfaces que o usuário terá que transpor para acessar o experimento	Máximo 2	Contagem
Índice de segurança	Fator de segurança representando o risco, danos materiais ou pessoais, que o experimento oferece no ambiente em que se encontra (1=Baixo, 2=Médio ou 3=Alto)	1	(não se aplica)

Tipos de dispositivos diferentes de acesso	Tipos de dispositivos que poderão acessar os experimentos como: PC's, Tablets, celulares, smartphones, etc	Mínimo 1	Não se aplica
Bloco de tempo alocado para o usuário	Extensão do período de utilização por usuário e por sessão	1h	Cronometragem
Tempo de resposta do sistema	Desempenho do sistema em responder às alterações dos parâmetros, atualização de imagens, a execução de um acionamento, etc	Máximo 2s	Cronometragem
Resolução da medida de temperatura	Grau de variação de resultados da medição	1°C	Medição
Resolução de atuação de potência (resfriamento/aquecimento)	Grau de variação da atuação de potência	1%	Medição

Para mapear as relações entre os requisitos e as especificações técnicas de projeto, utilizou-se a técnica da Casa da Qualidade do método QFD, conforme Anexo 1. Como resultado, tem-se uma ordenação por prioridade de importância das especificações, o que facilita tomadas de decisão sobre a utilização dos recursos disponíveis.

### 3.3 SÍNTESE CONCEITUAL

A partir da função global “Uso remoto da planta de processo de temperatura” foram derivadas e classificadas as funções parciais de acordo com a camada a qual pertencem (hardware, software e sistema físico). De modo a melhor organizar as informações, entendeu-se por bem estratificar a estrutura funcional em duas categorias: Local e Remoto.

Os princípios de solução aplicáveis a cada função parcial foram organizados na matriz morfológica, de acordo com a Tabela 7. A coleção dos princípios de solução reflete a experiência da equipe e os esforços de estudo e pesquisa acerca das tecnologias importantes para o projeto, incluindo elementos e conceitos observados em outros WebLabs existentes.

Tabela 7 – Análise Funcional e Matriz Morfológica

CATEGORIA	FUNÇÃO		SUBFUNÇÃO		CAMADA	1	2	3
Remoto	F1	Acessar o WebLab *1	F1.1	Conectar à internet (dispositivo)	Hardware	desktop/laptop	Desktop/laptop e Smartphone, Tablet	
			F1.2	Conectar à internet (navegador)	Software	Microsoft Internet Explorer	IE e Mozilla Firefox	IE e Google Chrome
	F2	Visualizar o experimento			Hardware	1 tela de visualização + Alto-falantes	2 telas de visualização (desktop estendido) + Alto-falantes	
	F3	Estabelecer interface gráfica com o usuário			Software	Painel em LabVIEW + Plug-in	Java	
Local (planta + servidor)	F4	Variar o fluxo de ar no interior do forno	F4.1	Variar a vazão	Sist. Físico	Atuador proporcional elétrico	Atuador proporcional pneumático	
			F4.2	Sustentar o atuador	Sist. Físico	Com suporte mecânico e aparafusado na estrutura da planta	Aparafusado diretamente na estrutura da planta	
			F4.3	Controlar o atuador	Hardware	I/O multifuncional	Microcontrolador	CLP
			F4.4	Intercomunicar Servidor e Atuador	Hardware	I/O multifuncional	Microcontrolador	CLP
	F5	Variar a abertura do damper	F5.1	Realizar movimentação mecânica	Sist. Físico	Atuador elétrico	Atuador pneumático	
			F5.2	Sustentar o atuador	Sist. Físico	Com suporte mecânico e aparafusado na estrutura da planta	Aparafusado diretamente na estrutura da planta	
			F5.3	Controlar o atuador	Hardware	I/O multifuncional	Microcontrolador	CLP
			F5.4	Comunicar Servidor e Atuador	Hardware	I/O multifuncional	Microcontrolador	CLP
			F5.5	Delimitar o curso de abertura	Sist. Físico	Chave fim de curso	Sensor ótico	
			F5.6	Sustentar os delimitadores de curso	Sist. Físico	Parafusos	Cola	Fita adesiva
	F6	Ler Temperatura *2			Hardware	I/O multifuncional	Microcontrolador	CLP
	F7	Comandar o TRIAC *3			Hardware	I/O multifuncional	Microcontrolador	CLP
	F8	Estabelecer videoconferência	F8.1	Capturar áudio e vídeo	Hardware	Sistemas de alta resolução	Sistemas de baixa resolução	
			F8.2	Sustentar sistema de capt. de áudio/vídeo	Sist. Físico	Suporte fixo aparafusado na estrutura da planta	Suporte retrátil aparafusado na estrutura da planta	
			F8.3	Transmitir dados	Software	Funções de imagem do LabVIEW™	Skype™	Soft. proprietário
F9	Coletar, armazenar e processar os dados do experimento e do usuário			Software	LabVIEW™	Supervisório Elipse	C++ Builder	
F10	Prover segurança da rede			Software	Antivírus	Firewall	Firewall e Antivírus	
F11	Prover proteção de acesso			Software	Login	Verificação de IP	Login e IP	
F12	Realizar comunicação de dados (Servidor / Cliente remoto)			Soft/Hard	LabVIEW WebServer + Remote Panel	WebServer + Web Services		
F13	Sustentar o sistema de proteção / controle / intertravamento			Sist. Físico	Quadro de comando	Suporte aparente		
F14	Proteger contra falha elétrica			Sist. Físico	Dispositivo de proteção residual (DR)	Disjuntor	Fusível	
F15	Intertravar modos de operação (local/planta ou remoto/weblab)			Sist. Físico	Circuito dedicado com Relés(bobina 5V)	Relés com bobina 127V, fixados em trilhos padrão DIN		

Notas:

\* 1 – Meios e dispositivos utilizados pelo usuário para acessar o WebLab. Procurou-se elencar as soluções padrões comumente utilizadas pelos usuários.

\* 2 – Dispositivo pelo qual será realizada a comunicação entre o servidor e os transmissores de temperatura - TT's.

\* 3 – Dispositivo pelo qual será enviado o sinal de comando de disparo do TRIAC. Este tiristor faz parte do circuito responsável pelo aquecimento do forno, já existente na planta de temperatura.

Duas alternativas de solução geral (S1 e S2) foram montadas e levadas à comparação através do método de Pugh, conforme a Tabela 8. Os critérios adotados correspondem aos requisitos anteriormente delineados, constantes da matriz de QFD. A escala utilizada para o grau de desempenho de cada solução ante os critérios adotados foi: 1 (baixo), 3 (médio) e 9 (alto).

Tabela 8 - Aplicação do Método de Pugh na seleção da melhor alternativa.

Ordem	Especificações Técnicas	Importância %	S1	S2
1	Qtde. de operações do usuário	6,1	9	9
2	Qtde. de pontos de visualização por câmeras	4,4	3	3
3	Qtde. de pontos de interação	8,2	3	3
4	Grau de detalhamento das instruções	7,3	9	9
5	Instrumentos de coleta de opinião	5,4	9	9
6	Nº de sistemas operacionais	5,7	3	9
7	Espaço para armazenar dados	4,3	9	9
8	Nº de repetições do experimento	8,7	9	9
9	Largura de banda de rede	6,8	9	3
10	Índice de disponibilidade do sistema	4,9	9	9
11	Nº de telas p/ acessar um experimento	2,9	9	3
12	Índice de segurança	8,2	9	9
13	Tipos de dispositivos de acesso	10,8	3	9
14	Duração do bloco de tempo de utilização	4,9	9	9
15	Tempo de resposta do sistema	8,1	9	3
16	Resolução da medida de temperatura	1,6	9	9
17	Resolução de atuação de potência	1,6	9	9
<b>Total</b>		<b>100</b>	<b>725</b>	<b>717</b>

Em termos gerais, as duas soluções poderiam ser adotadas para o desenvolvimento do projeto integrado do WebLab; porém, a disponibilidade de recursos para o projeto favorece ligeiramente a Opção 1.

A solução S1 prevê o acesso ao WebLab através de um computador (*desktop* ou *notebook*) levando em conta a versatilidade e capacidade de processamento disponíveis nestes equipamentos. A solução S2 toma por base a mobilidade. Com a disponibilidade crescente de novas redes de operação 3G e 4G os *smartphones* e os *tablets* oferecem um grande potencial para aplicações em diferentes áreas tecnológicas. Porém, um fator ainda limitante é a sua baixa capacidade de

processamento e armazenamento de dados quando comparados às soluções *desktop*.

Outro fator relevante na definição das soluções S1 e S2 trata-se da escolha do hardware e software que irão gerenciar e processar os dados do usuário e do experimento, pois um está atrelado ao outro. A definição do uso do CLP como gerenciador do processo implica na utilização de um software supervisor enquanto a utilização do software LabVIEW™ não requer software adicional.

A implementação da solução S1 sugerida requer que adaptações sejam feitas na planta de temperatura existente para que a mesma possa operar remotamente ou localmente de modo a garantir a confiabilidade e segurança do sistema.

Através da solução S1 o usuário terá acesso ao WebLab à partir de um PC conectado à internet. O acesso à página *web* do experimento é realizado através do *browser* Internet Explorer fazendo-se necessário a instalação do *plug-in* LabVIEW Run-Time Engine 2012, encontrado no site da National Instruments. Como medida de segurança é necessário que se faça *login*, com entradas de usuário e senha. Uma vez conectado, o usuário tem acesso aos parâmetros de configuração do experimento selecionado bem como à monitoração do experimento via *webcam*. Por tratar-se de uma aplicação que não envolve a necessidade de transmissão de áudio e vídeo com alta definição, optou-se pela adoção de dispositivos de baixa resolução para realizar esta tarefa. A sustentação da *webcam* será implementada através de um suporte retrátil fixado junto a estrutura da planta. Esta solução permite que a *webcam* seja recolhida quando não estiver sendo utilizada.

O servidor *Web (Labview WebServer + Remote Panel)*, contemplado na solução S1, hospeda a página da aplicação e tem as seguintes funções:

- Garantir o acesso seguro ao experimento;
- Armazenar os dados temporários bem como informações de usuário;
- Gerenciar o agendamento dos usuários;
- Gerenciar a operação da planta de temperatura através do recebimento e envio de sinais de controle aos atuadores, à válvula proporcional e ao sistema de aquecimento;
- Realizar a comunicação de dados (Servidor / Cliente remoto).

A aquisição e envio dos sinais de entrada e saída pelo servidor é realizada através de comunicação TCP/IP. Um CLP auxiliar, contendo entradas e saídas analógicas (4 a 20mA, 0 a 10V) e digitais (0 ou 24V), comunica-se com o servidor através da rede local ethernet.

O CLP terá as seguintes funções:

- Receber os sinais (4-20mA) dos transmissores de temperaturas (TT's);
- Enviar os sinais dos TT's ao servidor via TCP/IP;
- Receber os sinais de controle do servidor (% Abertura Damper, % Abertura Válvula, % Aquecimento do Forno);
- Enviar sinal de controle (4-20mA) de aquecimento do forno ao TRIAC;
- Enviar o sinal de controle (0-10V) de posicionamento do damper ao microcontrolador;
- Enviar o sinal de controle (0-10V) de variação da vazão de ar à válvula proporcional;

A válvula proporcional de ar comprimido (0-10V) entra em substituição à válvula manual (registro de ar) de forma a permitir que o usuário realize mudanças nas características do processo.

Para possibilitar a geração de distúrbios acrescentou-se um atuador elétrico (0-10V) junto ao damper. A IHM do CLP foi utilizada para possibilitar a visualização local dos sinais provenientes dos transmissores de temperatura. O LabVIEW™ foi a solução de software adotada para o desenvolvimento da interface *web* bem como das ferramentas que executam o gerenciamento e processamento dos dados. Os softwares e aplicativos baseiam-se no sistema operacional Microsoft.

Para garantir que não ocorra operação local ou remota simultâneas especificou-se um dispositivo de intertravamento eletromecânico junto a porta frontal do quadro de comando, constituído por um contator e uma botoeira liga/desliga. Estes dispositivo, em conjunto com um hardware (placa de relés) especialmente projetado para realizar o chaveamento da operação, bloqueia o acesso aos conectores do painel da planta de temperatura impedindo que conexões indesejadas ocorram enquanto o experimento estiver em modo remoto. Em caso de pane a

alimentação elétrica poderá ser cortada localmente por meio de uma botoeira de emergência localizada na porta do painel frontal.

A segurança da rede será realizada através do sistema de firewall e antivírus já existente na PUCPR, em conformidade com a política de segurança por esta adotada.

Os recursos adicionais a serem implementados em complemento aos dispositivos já existentes na planta de temperatura descrita anteriormente são:

Servidor compartilhado contendo:

- LabVIEW™ ;
- Firewall e antivírus;
- Recursos de vídeo conferência.

Planta de temperatura:

- Dispositivo de proteção e intertravamento (Disjuntor e Contator);
- Dispositivo de intertravamento (local / remoto) por relés;
- Indicador de temperatura (IHM);
- Atuador elétrico para posicionamento do damper;
- Válvula pneumática linear com posicionador eletrônico para controle da vazão de ar comprimido.

Do ponto de vista do usuário os recursos necessários ao acesso, configuração de parâmetros, exibição e armazenamento de dados, visualização e execução do experimento são:

- Computador com dispositivos multimídia (áudio e vídeo) e recursos de acesso à internet;
- Browser Internet Explorer;
- Plug-in do LabVIEW™.

O esquema da arquitetura do WebLab para a planta de processo térmico é exibido através da Figura 4.

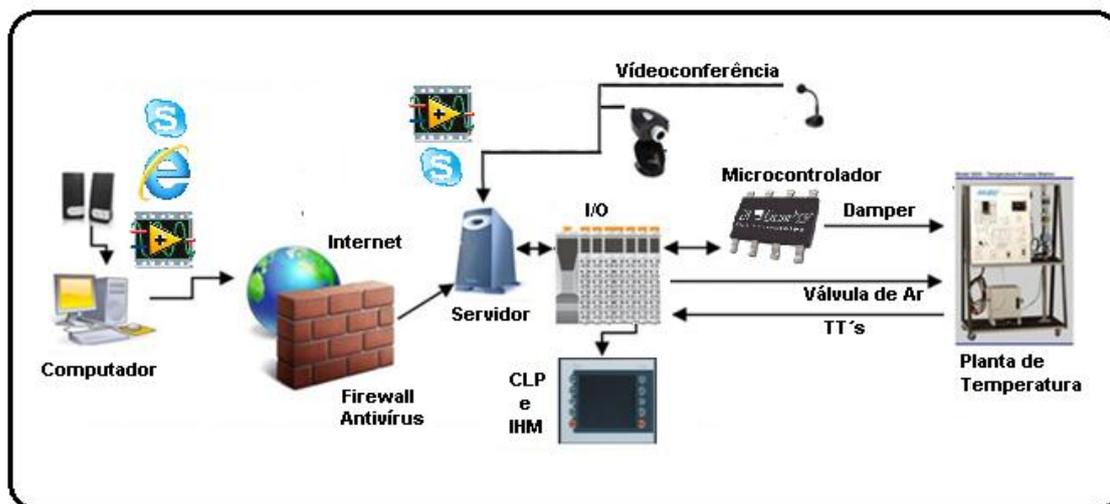


Figura 4 – Arquitetura do WebLab para a Planta de Processo Térmico.

Adaptado de Siqueira et al. (2012).

Um diagrama esquemático representando a solução conceitual selecionada é apresentado na Figura 5.

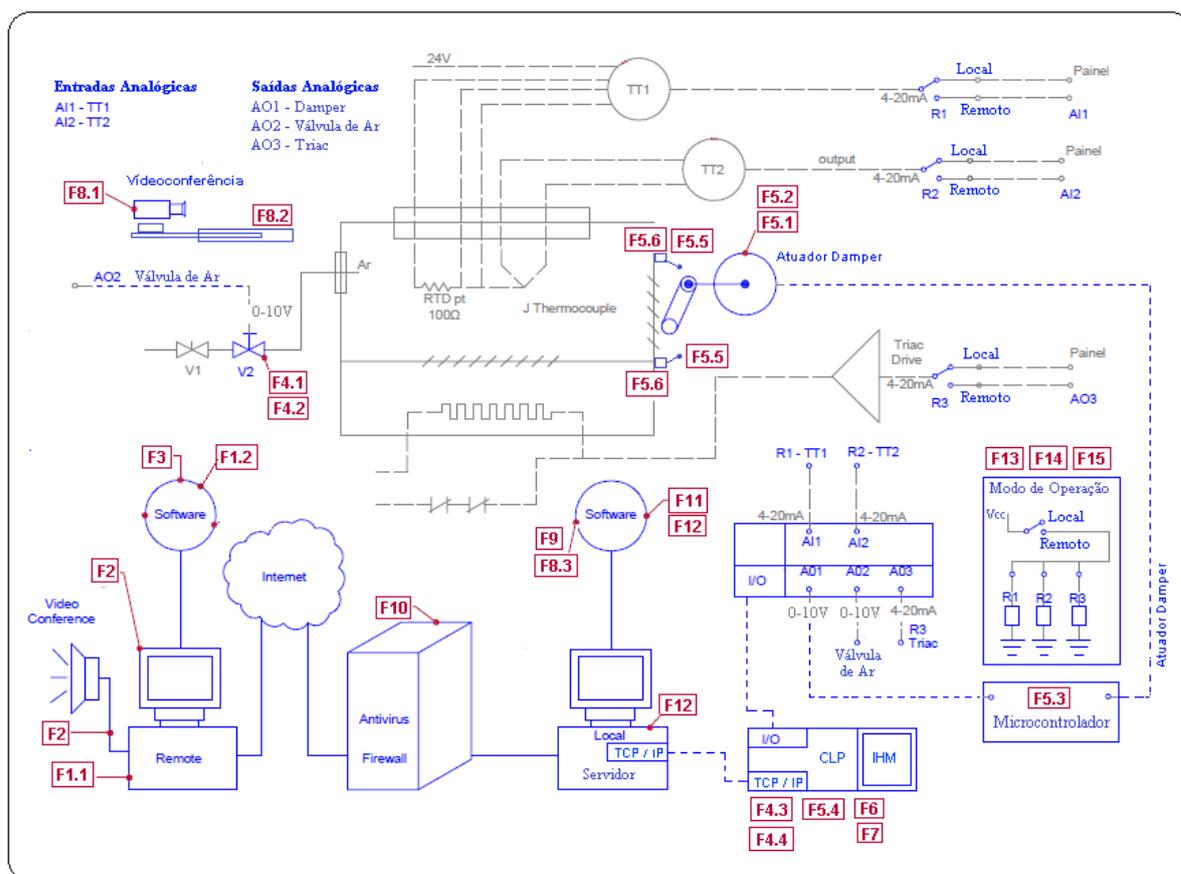


Figura 5 – Diagrama Técnico da Solução Conceitual Selecionada.

Adaptado de Siqueira et al. (2012).

### 3.4 ANÁLISE, SIMULAÇÃO E DIMENSIONAMENTO

Os itens seguintes descrevem as características dos principais equipamentos empregados na solução do sistema.

#### 3.4.1 Sistema eletromecânico de acionamento do damper

- A movimentação do damper será realizada através de motor de passo retirado de um *scanner* de mesa. O motor tem as seguintes características: unipolar (1,8° por passo); corrente de aproximadamente 100mA por fase (obtido através de ensaio em bancada); tensão de alimentação 5V.
- A sustentação mecânica do sistema de acionamento foi implementada através de peças retiradas de um escâner de mesa.
- A linearização da área de abertura da janela circular do damper, em função de seu deslocamento, conforme Figura 6, pode ser realizada de acordo com as seguintes equações:

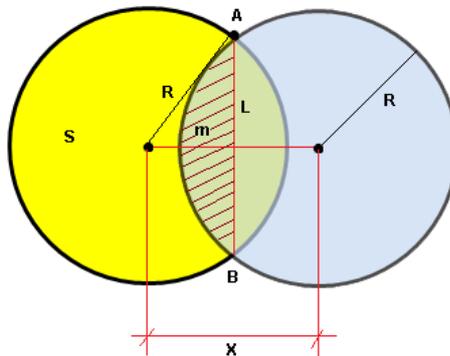


Figura 6 – Linearização da Área de Abertura do Damper.

$$S = \pi R^2 - \left( 2 \cdot \arctg \left( \frac{L}{m} \right) \cdot R^2 - 2 \cdot m \cdot L \right)$$

onde,

$$L = \frac{1}{2} \sqrt{4R^2 - X^2}$$

$$m = \frac{X}{2}$$

$$0 < X < 2R$$

Deslocamento linear do damper:

$$D = \frac{Pm \cdot \pi \cdot Rp}{180^\circ}$$

Onde: Pm = passo do motor = 1,8° e Rp = raio da polia = 0,5 cm

$$D = \frac{1,8^\circ \times 3,1415 \times 0,5\text{cm}}{180^\circ} = 0,0157\text{cm (por passo do motor)}$$

A Figura 7 exibe a variação da área de abertura do *damper* em função de seu deslocamento linear.

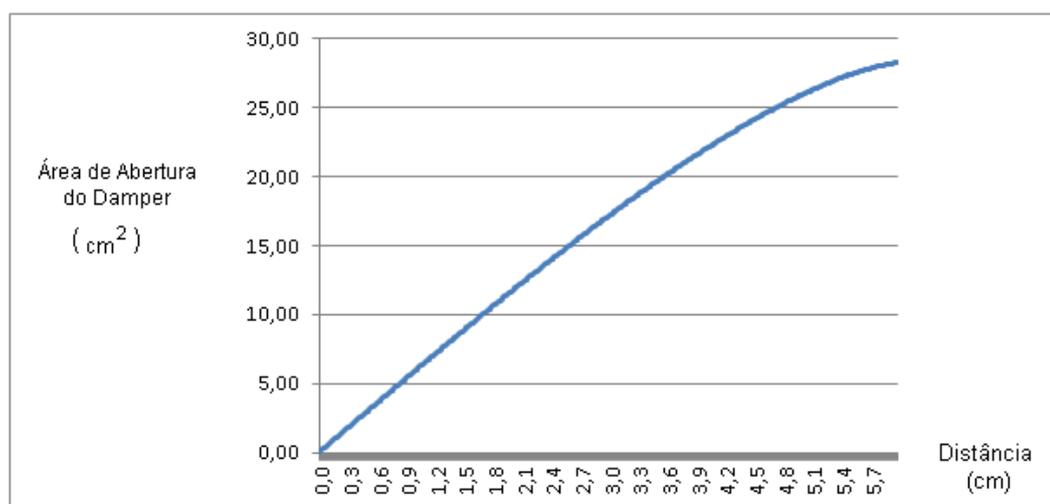


Figura 7 – Variação da área de abertura do *damper* em função de seu deslocamento linear.

### 3.4.2 Circuito eletrônico para controlar o damper

- Microcontrolador adotado: Zilog Z8 encore F0822; CPU 20 MHz; 2 conversores analógico-digital (ADC) de 10 bits; 8 KB de memória flash; tensão de alimentação de 2,7 V a 3,6 V.
- Conversão dos sinais da saída analógica do CLP (0-10V, abertura do damper) para a porta de entrada do microcontrolador (0-3V): Circuito implementado com amplificador operacional (AO) LM358, conforme Anexo 2.
- Circuito transistorizado de potência: Par Darlington com BC547 e TIP31C, conforme Anexo 2.

TIP31C:  $I_c = 3A$ ,  $V_{CBO} = 100V$ ,  $\beta_{sat} = 10$  (Obs:  $I_c=3A > I(\text{motor})=100mA$ )

BC547:  $I_c = 100mA$ ,  $V_{CBO} = 50V$ ,  $\beta_{sat} = 20$ .

### 3.4.3 Sistema de vídeo chamada

- Uma webcam com microfone embutido e conexão USB 2.0;
- Cabo de extensão USB 2.0 (5m) amplificado;
- Haste retrátil para sustentação da webcam;
- Programa Skype™.

### 3.4.4 Equipamentos auxiliares

- Uma fonte de alimentação DC: +12V, +5V , 300W/20A  
Obs:  $I(\text{fonte}) = 20A > I(\text{estimado}) = 2A$ ;
- Um disjuntor monofásico: 7A;
- Fiação 1,5mm<sup>2</sup> (15A);
- Um contator trifásico: 7A,127V, com um contato auxiliar;
- Uma Botoeira de emergência tipo cogumelo;
- Um botão Liga/Desliga;
- Um Quadro de comando: 60x80 cm;

### 3.4.5 Servidor do experimento

- Computador DELL Optiplex 330, Intel Core 2 Duo 2.4 GHz, 2 GB RAM, 160 GB HD, Ethernet 10/100, caixa de som com conexão USB;
- Sistema operacional: Windows 7;
- Softwares: LabView™- versão 2012, Skype™.

Observações:

- 1) O valor estimado da quantidade de dados a serem salvos por experimento (1h / experimento) foi de aproximadamente 2MB. Considerando que o WebLab esteja disponível 14 horas/dia durante 30 dias/mês teremos que armazenar algo em torno de 840MB por mês.
- 2) Os requisitos mínimos do sistema para executar o NI LabVIEW Development Systems são exibido através da Figura 8.

Windows	Run-time Engine	Ambiente de desenvolvimento
Processador	Pentium III/Celeron 866 MHz ou equivalente	Pentium 4/M ou equivalente
RAM	256 MB	1 GB
Resolução da tela	1024 x 768 pixels	1024 x 768 pixels
Sistema operacional	Windows 8/7/Vista (32 bits e 64 bits) Windows XP SP3 (32 bits) Windows Server 2003 R2 (32 bits) Windows Server 2008 R2 (64 bits)	Windows 8/7/Vista (32 bits e 64 bits) Windows XP SP3 (32 bits) Windows Server 2003 R2 (32 bits) Windows Server 2008 R2 (64 bits)
Espaço em disco	353 MB	3,67 GB (inclui drivers default do DVD de drivers de dispositivo da National Instruments)

Figura 8 – Requisitos do sistema para NI LabVIEW Development Systems.

Fonte: National Instruments.

### 3.4.6 Resolução do CLP

- Resolução da entrada e da saída analógica de corrente (4-20mA):

$$R = \frac{Vref(clp)}{2^n} = \frac{(20 - 4)mA}{2^{15}} = \frac{16mA}{32768} \cong 0,49\mu A$$

Obs.: A resolução especificada como meta foi de 1%, ou seja, 0,16mA.

Para uma faixa de medição de temperatura de 0 a 200°C (correspondendo aos 4-20mA), tem-se capacidade para a seguinte resolução teórica:

$$R = \frac{Vref(clp)}{2^n} = \frac{200^\circ C}{2^{15}} = \frac{200^\circ C}{32768} \cong 0,0061^\circ C$$

- Resolução da entrada e da saída analógica de tensão (0-10V) nos comandos das aberturas da válvula e do *damper*:

$$R = \frac{Vref(clp)}{2^n} = \frac{10V}{2^{15}} = \frac{10V}{32768} \cong 0,3\mu V$$

Obs.: A resolução especificada como meta foi de 1°C.

## 3.5 DETALHAMENTO E DOCUMENTAÇÃO DA SOLUÇÃO

### 3.5.1 Sistema de automação do *damper*

O *damper*, localizado na parte lateral do forno conforme a Figura 9, permite alterar a característica de resposta do sistema controlado através de sua abertura: quanto maior a abertura do *damper*, maior será a perda de calor no interior do forno.

A sua automação envolveu o desenvolvimento de sistema específico de hardware e de software para possibilitar o seu posicionamento sob comando remoto.



Figura 9 – Vista do Damper Antes e Após a Automação.

As características físicas e operacionais envolvidas na automação do sistema atuador do damper são descritas nos tópicos seguintes:

- O sistema físico é composto por partes eletromecânicas, responsáveis pela movimentação e sustentação de todo conjunto posicionador, e emprega peças retiradas de um escâner de mesa;
- O deslocamento linear do damper é realizado pelo acionamento de um motor de passo acoplado ao mesmo por meio de uma correia, onde dois sensores fim de curso são utilizados para delimitar o percurso;
- O controle de posicionamento do damper é efetuado por um circuito eletrônico microcontrolado desenvolvido especificamente para este fim. A Figura 10 exhibe os detalhes do circuito eletrônico criado para esta finalidade. Seu esquemático elétrico encontra-se no Anexo 2.

O sistema de controle de posicionamento recebe um sinal analógico de tensão (0–10V), representando a referência de posição desejada, e então posiciona o damper de modo a resultar na abertura da janela desejada. Um sinal de tensão de 10V corresponde a uma abertura de 100% do damper, e a abertura da janela segue a fórmula de linearização apresentada em 3.3.1.



Figura 10 – Circuito Eletrônico que Executa o Posicionamento do Damper.

A rotina de trabalho executada pelo microcontrolador, responsável por gerenciar o posicionamento do damper, pode ser verificada através do fluxograma exibido na Figura 11. A rotina inicial a ser executada é denominada de “ciclo de *homing*”, onde o microcontrolador verifica o número de pulsos necessários a ser enviado ao motor de passo para que a janela do damper esteja totalmente aberta. Ao final do ciclo de *homing*, o damper se encontra na posição inicial, totalmente fechado. A partir deste ponto, o microcontrolador entra em uma rotina cíclica de verificação e atualização do posicionamento do damper, segundo a referência (*set-point*) lida pelo conversor analógico/digital (A/D).

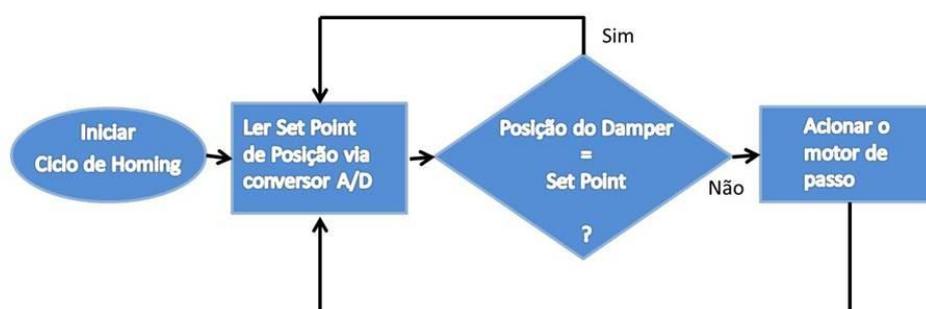


Figura 11 – Fluxograma da Rotina de Posicionamento do Damper.

### 3.5.2 Válvula proporcional para controle da vazão de ar comprimido

A geração de distúrbios e/ou a variação da carga térmica no processo de temperatura pode ser realizada através da injeção de ar comprimido no interior do

forno. Para que esta ação possa ocorrer por acionamento remoto, a solução adotada foi o uso de uma válvula pneumática linear, com atuador eletrônico, que permite controlar a vazão de ar comprimido numa escala de 0-100% de seu valor máximo. Empregou-se uma válvula da marca FESTO, modelo MPP – 3 – 1/8, mostrada na Figura 12. Esta válvula é comandada eletronicamente através de um sinal de tensão de entrada analógico na faixa de 0 a 10V, sendo que a máxima vazão ocorre quando o valor do sinal de entrada é igual a 10V.



Figura 12 – Válvula Pneumática Linear Com Atuador Eletrônico. FESTO, modelo MPP – 3 – 1/8 .

### 3.5.3 Sistema de proteção e intertravamento

O diagrama elétrico exibido na Figura 13 demonstra o esquema de proteção e intertravamento adotado para garantir a segurança de operação do sistema. Este circuito está instalado em um quadro de comando junto à planta de temperatura, e permite bloquear o acesso aos conectores do painel local, impedindo que conexões no painel da planta, feitas inadvertidamente enquanto o sistema estiver operando em modo remoto, possam interferir no experimento em andamento.

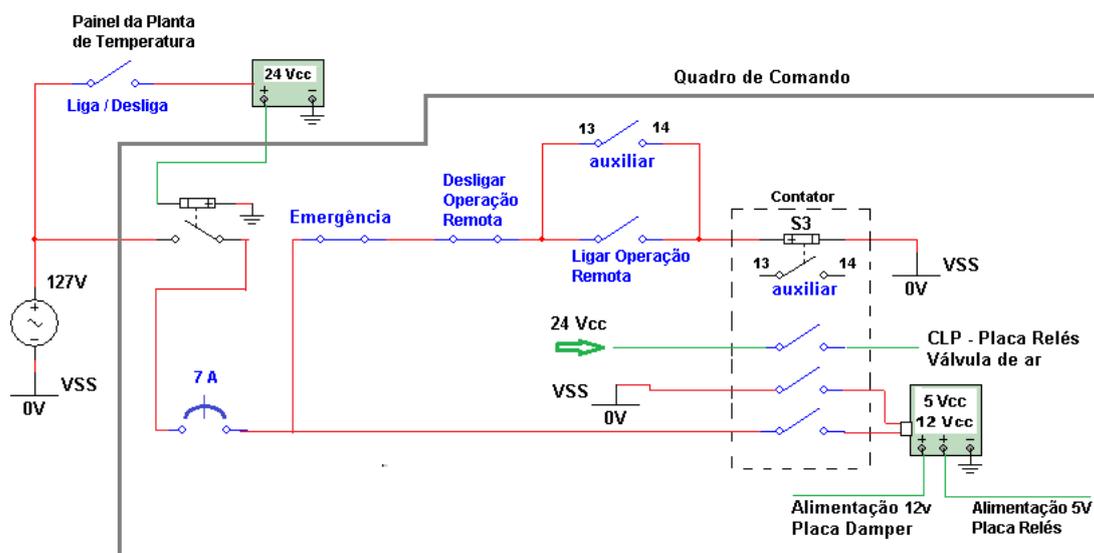


Figura 13 - Diagrama Elétrico do Quadro de Comando.

Um circuito eletrônico auxiliar, contendo relés, é utilizado para realizar o chaveamento propriamente dito. Seu diagrama elétrico pode ser verificado através do Anexo 3. Em caso de pane, uma botoeira de emergência pode ser acionada junto ao quadro de comando local.

### 3.5.4 Sistema de comunicação PC-CLP-Planta

Para realizar a comunicação entre o software gerenciador do experimento (em LabVIEW™) e os componentes da planta de temperatura, adotou-se o uso de um controlador lógico programável (CLP), com entradas e saídas digitais e analógicas. Este controlador terá as funções de:

- Receber os sinais analógicos de corrente (4-20mA) dos transmissores de temperatura;
- Enviar os sinais analógicos de tensão (0-10V) ao damper e a válvula de ar;
- Enviar o sinal analógico de corrente (4-20mA) ao triac;
- Transmitir os dados coletados ao software gerenciador do experimento (LabVIEW™) através do protocolo TCP/IP.

Conforme exibido na Figura 14, o CLP atuará como servidor e o software LabVIEW™ como cliente (localmente). Desta maneira garante-se que, sempre que for requisitada uma conexão pelo cliente, o servidor estará disponível.

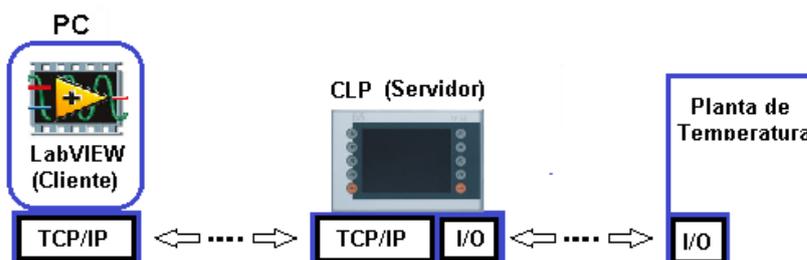


Figura 14 – Sistema de Comunicação Entre o PC, o CLP e os Equipamentos da Planta de Temperatura.

Com o objetivo de cooperar com o desenvolvimento deste projeto, a empresa B&R Automation doou à PUCPR um CLP (modelo PowerPanel PP45) e ministrou um curso para que fosse possível programá-lo de acordo com as necessidades deste trabalho. O equipamento conta com uma interface homem máquina (IHM) integrada, que pode ser programada para permitir a visualização dos parâmetros monitorados, tais como os sinais dos transmissores de temperatura bem como os sinais dos atuadores (damper e válvula). A Figura 15 exibe um detalhe do conjunto CLP / IHM B&R, e o painel de comando onde estão instalados este e outros equipamentos.



Figura 15 – Conjunto CLP e IHM B&R Automation e Painel de Controle

A IHM possui duas portas de comunicação USB, display QVGA de 5.7", uma porta Ethernet e um link de comunicação X2X para conexão com os dois módulos de entrada e saída (analógicas e digitais) utilizados neste projeto. Os modelos dos dois módulos de I/O são: B&R X20CM8281 e B&R X20DM9324. O primeiro contém 4 entradas digitais, 2 saídas digitais, 1 entrada analógica e 1 saída analógica; o segundo contém 8 entradas digitais e 4 saídas digitais.

A comunicação entre o software do CLP e o software LabVIEW™ foi estabelecida através da porta Ethernet, por meio do protocolo de comunicação TCP/IP.

A programação do CLP foi realizada por meio do software B&R Automation Studio, tendo este sido configurado como servidor dentro da estrutura de comunicação adotada. A sequência de execução das tarefas realizadas é descrita através do pseudocódigo exibido na Tabela 9:

Tabela 9 – Pseudocódigo representando a programação do CLP

1	Abrir a interface com a internet;
2	Escutar (esperar) as conexões possíveis;
3	Conexão com a rede efetuada? Se sim, Passo 4; senão Passo 3;
4	Esperar conexão por parte do cliente (Neste caso, o programa em LabVIEW™);
5	Conexão com o cliente efetuada? Se sim, Passo 6; senão Passo 4;
6	Esperar dados (Abertura Damper);
7	Dados recebidos? Se sim, Passo 8; senão Passo 6;
8	Esperar dados (Abertura da Válvula de Ar);
9	Dados recebidos? Se sim, Passo 10; senão Passo 8;
10	Esperar dados (Sinal de controle de % de aquecimento do forno).
11	Dados recebidos? Se sim, Passo 12; senão Passo 10;
12	Enviar dados (Temperatura PT100);
13	Dados enviados? Se sim, Passo 14;
14	Enviar dados (Temperatura Termopar tipo “J”);
15	Dados enviados? Se sim, Passo 6;

O envio / recebimento dos dados é realizado de forma sequencial e contínua até o fim da conexão com o cliente.

Para garantir a segurança do sistema em caso de perda de comunicação, a execução do programa é desviada para uma rotina de tratamento de erro onde o CLP irá enviar sinais para fechar o damper e a válvula de ar.

### 3.5.5 Videoconferência via Skype™ e fixação da webcam

Para permitir que o usuário do WebLab possa acompanhar em tempo real o equipamento que esta operando, empregou-se uma *webcam* para capturar o áudio e o vídeo do sistema em operação. Esta webcam foi fixada em um suporte retrátil,

capaz de realizar o posicionamento necessário, com o fim de se obter uma boa visão da planta e seus principais componentes, durante os experimentos. Pode-se observar a movimentação do damper e escutar os ruídos do sistema de ar comprimido e da válvula proporcional de admissão de ar. A Figura 16, mostra a webcam e seu suporte retrátil.



Figura 16 – Webcam para vídeo conferência e seu suporte.

A visualização do processo pelo usuário do WebLab é feita através de uma chamada Skype™. O Skype é um software de comunicação via internet, que oferece chamadas de áudio e vídeo, em tempo real, gratuitas entre seus usuários. Sua escolha deve-se pela sua versatilidade e simplicidade de uso, e também para não sobrecarregar o software principal do WebLab. Uma conta de uso exclusivo foi criada no Skype para o WebLab. Para realizar a videoconferência, é necessário que o usuário possua uma conta no Skype e tenha o programa instalado em seu computador.

### 3.5.6 Painel de controle

O painel de controle, ou quadro de comando geral, encontra-se instalado junto a planta de temperatura conforme Figura 17, e concentra os seguintes dispositivos e componentes:

- Dispositivos de intertravamento e segurança;
- CLP e IHM;
- Blocos de I/O;
- Placa de relés;

- Placa de controle do damper;
- Fonte de alimentação cc com saídas de 12V e 5V.

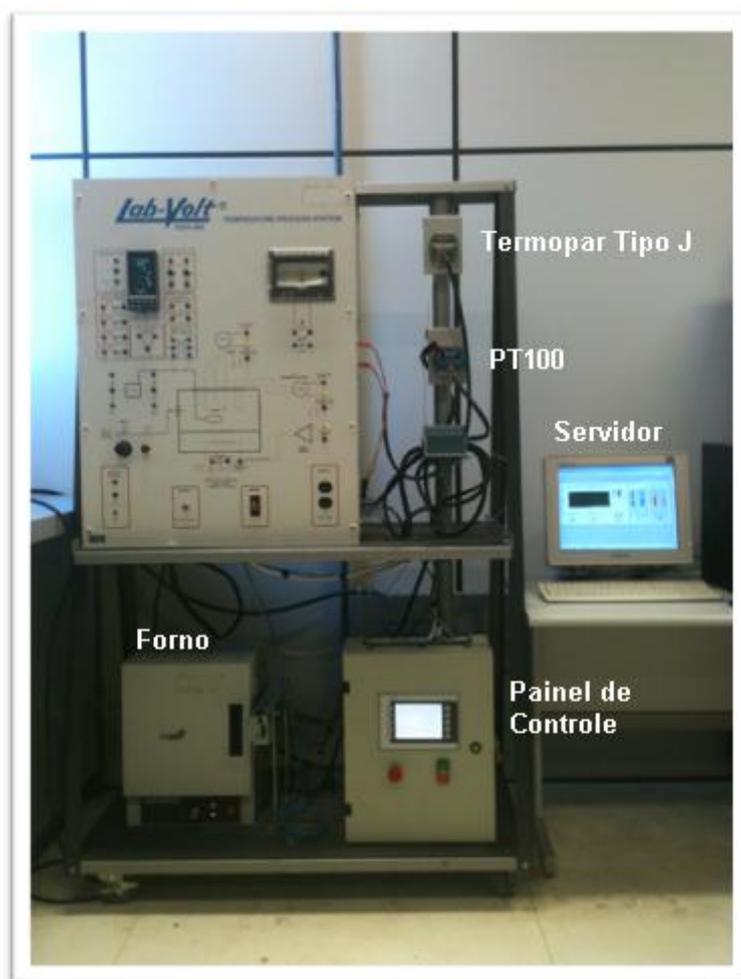


Figura 17 – Vista Geral do WebLab da Estação de Temperatura.

### 3.5.7 Software para cadastrar usuários

Como descrito no capítulo anterior, o LabVIEW™ foi a solução de software adotada para o desenvolvimento da interface *web*, bem como das ferramentas necessárias para realizar o gerenciamento de acesso ao WebLab e o ao controle do experimento.

Dois programas foram desenvolvidos para permitir que o usuário realize o cadastro, faça o login e tenha acesso ao controle do experimento. As rotinas de software utilizadas para realizar o cadastro e o login do usuário foram desenvolvidas com base no trabalho de Delay (2012).

As Figuras 18 e 19 exibem as interfaces do software utilizado para realizar o cadastro dos usuários. Nesta ação, são preenchidos os dados do usuário (nome, sobrenome e usuário Skype (opcional)), e é feita a escolha da data e hora para acessar o WebLab. O campo “Agenda” exibe uma lista dos períodos disponíveis / já reservados.

Após o correto preenchimento das informações e a confirmação do envio dos dados, o software irá criar, no servidor do WebLab, um diretório específico para cada usuário/sessão, onde serão armazenados os dados coletados durante a realização do experimento. Logo em seguida, é enviado um e-mail para o endereço cadastrado, conforme exibido na Figura 20, contendo:

- Dados de acesso: Login e Senha;
- Dados gerais: data de cadastro, hora de cadastro, data agendada, hora agendada e usuário Skype;
- *Link* para acessar o WebLab PUC PR;
- Anexo do manual de instruções contendo o arquivo “*Instruções.pdf*”.

Como a utilização do software de cadastramento não pode ser feita simultaneamente por múltiplos usuários, definiu-se um tempo de utilização de três minutos para a realização da tarefa, caso o sistema detecte a presença de diferentes usuários solicitando conexão.



Figura 18 – Tela do Cadastro de Usuários.

Figura 19 – Tela do Cadastro de Usuários.

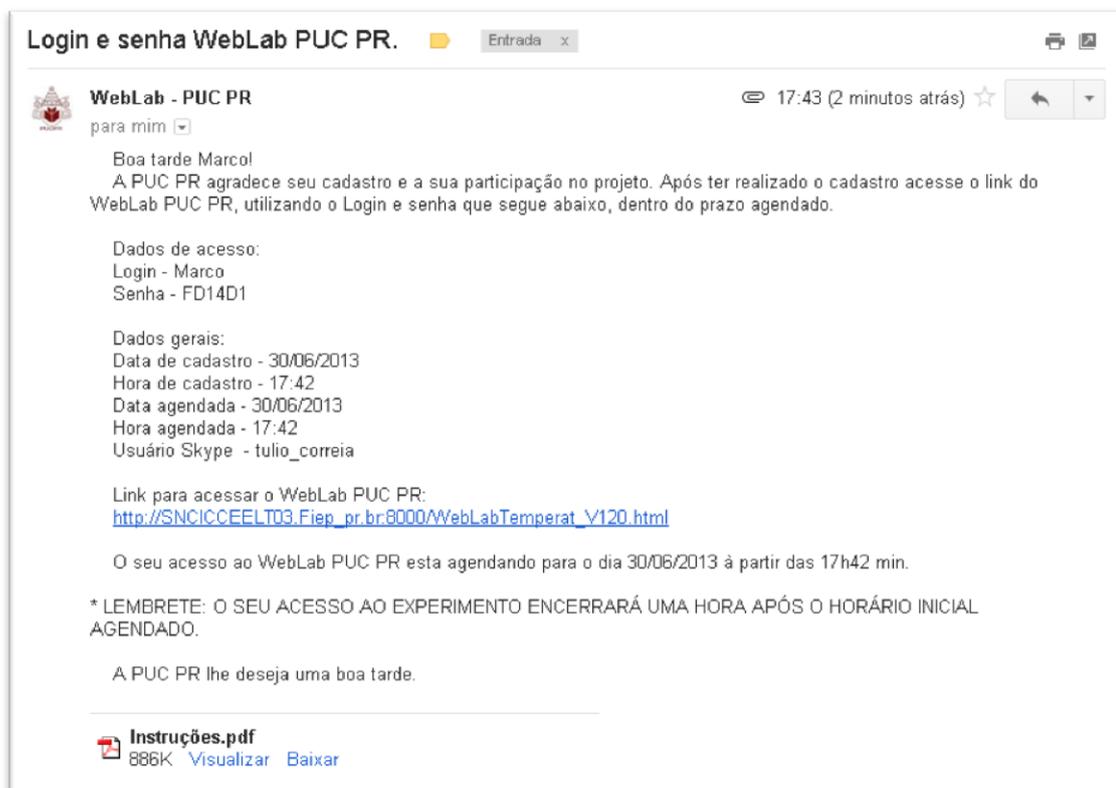


Figura 20 – Exemplo de E-mail Enviado ao Usuário Após a Confirmação do Cadastro.

### 3.5.8 Resumo dos arquivos gerados pelo sistema

- Sistema de Agendamento:  
*Login.txt* (dados do agendamento);
- Sistema de Controle:  
*TemperatureStation\_Data\_Hora.txt* (contém os dados registrados pelo usuário durante a execução de um ou mais experimentos);  
*TemperatureStation\_Data\_Hora.zip* (compactação de todos os arquivos gerados pelo usuário durante o período de acesso ao sistema).

### 3.5.9 Estrutura do software de gerenciamento do WebLab

O desenvolvimento do software utilizado para o gerenciamento do WebLab baseia-se na utilização da estrutura Produtor/Consumidor do LabVIEW™ e no processamento paralelo de algumas tarefas. Sua interface conta com seis telas que permitem ao usuário controlar e monitorar os parâmetros envolvidos no processo de controle de temperatura do forno, conforme descrito a seguir:

- Interface de controle;
- Interface de monitoração;
- Interface de comunicação (E-mail / Skype);
- Interface de atividades didáticas;
- Interface de avaliação do sistema;
- Interface de instruções;
- Interface de Login.

A Figura 21 exibe um detalhamento da arquitetura empregada no desenvolvimento do software de gerenciamento do WebLab. Os blocos numerados de 1 a 13 representam as principais rotinas cíclicas executadas paralelamente e os blocos A, B e C representam as rotinas executadas por eventos.

As rotinas executadas por evento são acionadas em alguma das seguintes ocorrências:

1. Estouro de tempo (nenhuma ação foi tomada pelo usuário durante um intervalo de tempo).

2. Botão Gravar Dados foi acionado;
3. Usuário selecionou alguma das opções do menu;
4. Botão Enviar Dados foi acionado;

No caso da ocorrência de algum evento, este é adicionado a um sistema de filas que coordena a sequência de execução das tarefas. Por exemplo, se o usuário selecionar a opção Inicializar Sistema do menu, teremos a ocorrência do evento “usuário selecionou menu” e o sistema encaminhará a solicitação para a fila 1, que por sua vez ativará a rotina “Inicializar Sistema”. No bloco A da Figura 21 estão elencadas as rotinas endereçadas pela fila 1 e no bloco B, as rotinas endereçadas pela fila 2. A Fila 3 destina-se apenas à rotina de controle e ativação do programa Skype, conforme indicado no bloco C.

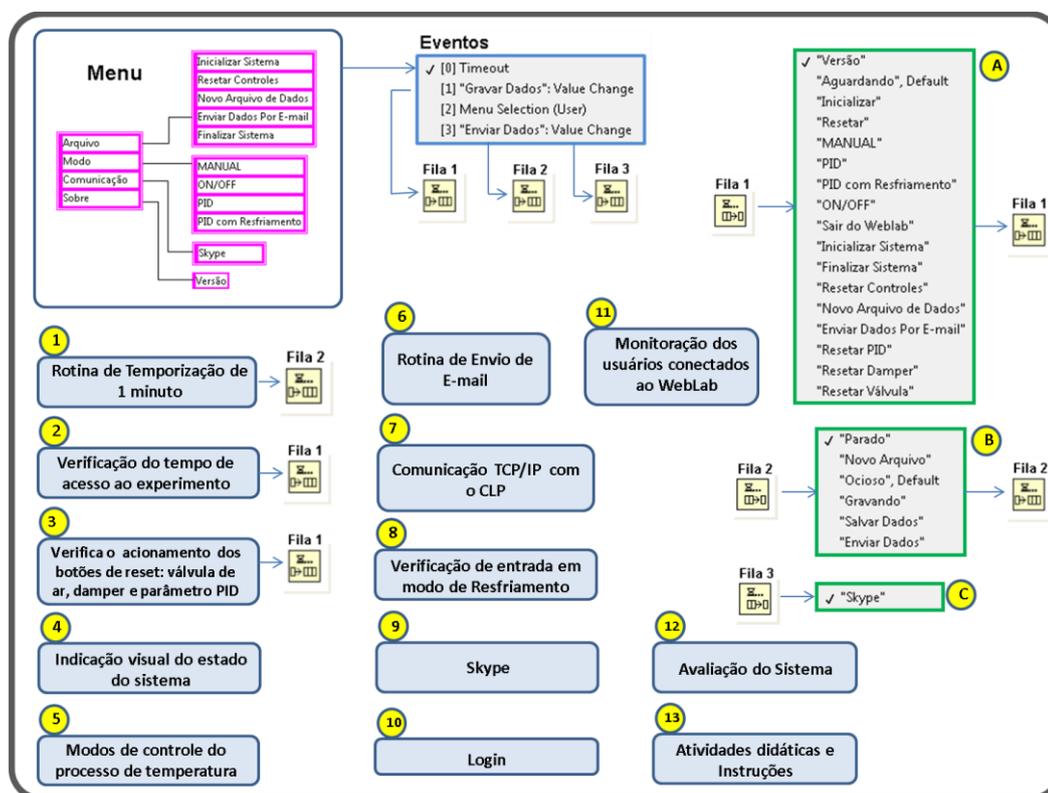


Figura 21 – Arquitetura do Software de Gerenciamento do WebLab.

As opções disponibilizadas no menu superior, conforme exibido na Figura 22, permitem que o usuário ative diferentes rotinas acionadas por evento e tem o intuito de auxiliar o gerenciamento do experimento a ser executado.

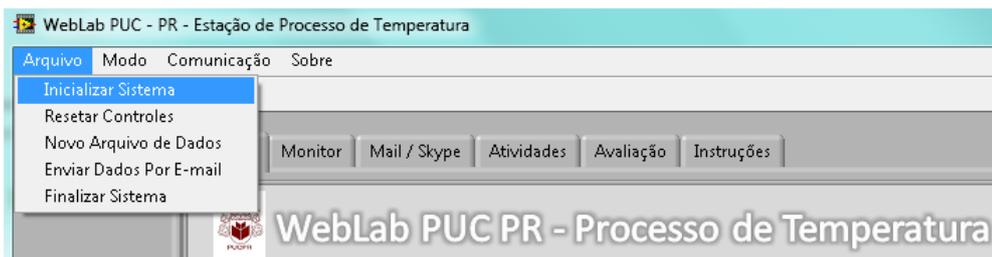


Figura 22 – Arquitetura do Software de Gerenciamento do WebLab.

As opções do menu superior são e explanadas nos tópicos seguintes:

- *Arquivo \ Inicializar Sistema*: a seleção desta opção irá ativar a rotina responsável pelo *reset* de todos os parâmetros ajustáveis do sistema, que por sua vez irá ativar a rotina responsável pela criação de um novo arquivo de dados. O nome do novo arquivo de dados gerado terá o seguinte formato: "TemperatureStation\_data\_hora.txt".
- *Arquivo \ Resetar Controles*: a seleção desta opção irá ativar a rotina responsável pelo *reset* de todos os parâmetros ajustáveis do sistema.
- *Arquivo \ Novo Arquivo de Dados*: A seleção desta opção irá ativar a rotina responsável pela criação de um novo arquivo de dados. O novo arquivo de dados terá o seguinte formato: "TemperatureStation\_data\_hora.txt".
- *Arquivo \ Enviar Dados por E-mail*: a seleção desta opção irá ativar a rotina responsável pelo envio dos dados armazenados durante a realização de um experimento. Todos os arquivos de dados gerados durante o experimento serão agrupados em um novo arquivo com o seguinte nome: "TemperatureStation\_data\_hora.zip". Este arquivo será enviado para o endereço de e-mail do usuário.
- *Arquivo \ Finalizar Sistema*: a seleção desta opção irá ativar, sequencialmente, as rotinas responsáveis pela finalização do sistema. Inicialmente será ativa a rotina responsável por resetar todos os parâmetros ajustáveis do sistema. Em seguida todos os arquivos de dados gerados durante o experimento serão agrupados em um novo arquivo com o seguinte nome: "TemperatureStation\_data\_hora.zip". Este arquivo será enviado para o endereço de e-mail do usuário. Ao término do envio do e-mail o sistema irá

notificar o usuário alertando que o Browser já pode ser fechado. A conexão com o programa Skype™ será encerrada, caso esteja ativa, e a execução do programa será redirecionada para a tela de Login.

- *Modo \ Manual:* esta opção ativa a rotina que configura o controlador para operar no modo de controle "Manual". O controlador irá atuar em malha aberta e o usuário irá definir manualmente o valor percentual da variável manipulada (percentual de aquecimento do forno - MV % Aq).
- *Modo \ ON-OFF:* esta opção ativa a rotina que configura o controlador para operar no modo de controle "ON-OFF". O controlador irá atuar em malha fechada e o usuário irá definir a referência de temperatura desejada (Set Point - °C).
- *Modo \ PID:* esta opção ativa a rotina que configura o controlador para operar no modo de controle "PID". O controlador irá atuar em malha fechada e o usuário irá definir o valor dos parâmetros de ajuste do controlador (ganho proporcional  $K_p$ , tempo integral  $T_i$  e tempo Derivativo  $T_d$ );
- *Modo \ PID com Resfriamento:* esta opção ativa a rotina que configura o controlador para operar no modo de controle "PID com Resfriamento". O controlador irá atuar em malha fechada e os parâmetros de ajuste do controlador (ganho proporcional  $K_p$ , tempo integral  $T_i$  e tempo Derivativo  $T_d$ ) serão definidos pelo usuário. Neste modo de controle a ação inversa do controlador (sinal de saída negativo) é utilizada para modular a abertura da válvula de ar com o intuito de acelerar o processo de resfriamento do forno;
- *Comunicação \ Skype:* esta opção irá redirecionar o programa para a interface de comunicação. Neste ambiente o usuário poderá ativar a rotina responsável pela execução do programa Skype™ que por sua vez irá transmitir o vídeo e o áudio gerado no local em que se encontra a planta de temperatura.
- *Sobre \ Versão:* esta opção irá ativar a visualização de uma mensagem contendo informações sobre a versão do software de controle do WebLab.

Paralelamente ao sistema de eventos temos a execução das rotinas cíclicas (blocos numerados de 1 a 13 na Figura 21 – Ver Anexo 6), responsáveis pelas seguintes tarefas:

Rotina 1 – Temporização de um minuto: quando o botão *Gravar Dados* é acionado esta rotina é habilitada para garantir que os dados coletados sejam gravados a cada intervalo um minuto. Ao término da temporização a rotina encaminha para a fila 2 o pedido de execução da rotina responsável pelo armazenamento dos dados (“Salvar Dados”).

Os dados coletados durante a realização de uma atividade são armazenados no diretório criado para cada usuário durante a etapa de cadastramento, onde o nome do arquivo gerado possui a seguinte formatação: “TemperatureStation\_data\_hora.txt”. A Figura 23 exibe apenas um detalhe do arquivo texto gerado.

Em sua totalidade os dados armazenados pelo sistema são: horário em que o dado foi coletado, tempo em milissegundos, set point, temperatura do sensor PT100, temperatura do sensor termopar tipo “J”, valor percentual da variável manipulada (MV%\_Aq), valor da abertura percentual do damper, valor da abertura percentual da válvula de ar, ganho Kp, tempo integral, tempo diferencial, tipo de controle utilizado, sensor utilizado e o nome fornecido para a atividade.

Hora	Tempo_(ms)	Set Point (°C)	Pt100 (°C)	Termop-J(°C)	MV%_Aq	Damper (%)	válvula(%)
12:36:30	13462875	70,0	55,462646	46,905518	100	99	35
12:36:31	13463875	70,0	55,462646	46,966553	100	99	35
12:36:32	13464875	70,0	55,462646	46,966553	100	99	35
12:36:33	13465875	70,0	55,590820	46,905518	100	99	35
12:36:34	13466875	70,0	55,401611	46,905518	100	99	35
12:36:35	13467875	70,0	55,401611	46,905518	100	99	35
12:36:36	13468875	70,0	55,401611	46,905518	100	99	35
12:36:37	13469875	70,0	55,462646	46,966553	100	99	35
12:36:38	13470875	70,0	55,590820	46,905518	100	99	35

Figura 23 – Detalhe do Arquivo Texto Gerado pelo Sistema.

- Rotina 2 – Verificação do tempo de acesso: controla o tempo de acesso permitido para realizar o experimento, segundo os dados do cadastro. No caso do estouro do tempo de acesso, a rotina encaminha para a fila 1 o pedido de execução da rotina de finalização (“Sair do WebLab”), onde o programa irá salvar os dados coletados, fechar o arquivo aberto, resetar todos

os controles, agrupar todos os arquivos de dados em um novo arquivo com extensão .zip e irá enviar este arquivo para e-mail do usuário.

- Rotina 3 – Botões de *reset* da Válvula de ar, da Abertura do Damper e dos Parâmetros PID: esta rotina verifica se algum dos botões de *reset* foi acionado. Em caso afirmativo a rotina encaminha para o sistema de fila o pedido de execução desta solicitação (*Resetar Damper, Resetar Válvula ou Resetar PID*).
- Rotina 4 – Indicação visual do estado do sistema: esta rotina é responsável por gerar uma indicação visual das ações tomadas pelo sistema, tais como “Gravando, Criando novo arquivo..., etc.”. A visualização destas mensagens ocorrerá no campo *Estado do Sistema*, localizado na tela de Controle.
- Rotina 5 – Modos de controle do processo de temperatura: nesta rotina é executado o tipo de controle selecionado pelo usuário ou definido pelo sistema (Manual, On-Off, PID, Resfriamento ou PID com resfriamento).
  - No modo de operação “Manual”, malha aberta, o usuário terá acesso à variável manipulada “MV% Aq” onde a potência do forno é ajustada numa escala de 0 a 100% de sua potência nominal.
  - No modo de operação “On-Off”, malha fechada, o usuário terá acesso apenas a referência de temperatura desejada. Neste caso o controlador irá ajustar a variável manipulada “MV% Aq” para 100% quando a temperatura estiver abaixo da referência, e para 0 (zero) quando a temperatura for maior ou igual a referência desejada.
  - No modo de operação “PID”, malha fechada, o usuário poderá ajustar a referência de temperatura desejada bem como os parâmetros do controlador PID (ganho proporcional Kp, Tempo integral Ti e Tempo derivativo Td).
  - O modo “Resfriamento” é utilizado apenas pelo sistema e tem a finalidade de resfriar o forno durante os dez minutos finais do tempo alocado para o experimento. Neste modo o sistema irá bloquear o acesso a todos os parâmetros do controlador.
  - O modo “PID com Resfriamento” assemelha-se ao modo “PID” porém neste caso a ação inversa do controlador (saída de controle negativa) é utilizada para modular a abertura da válvula de ar com a finalidade de acelerar o

processo de resfriamento do forno. Neste modo o usuário irá ajustar a referência de temperatura desejada bem como os parâmetros do controlador PID (ganho proporcional  $K_p$ , Tempo integral  $T_i$  e Tempo derivativo  $T_d$ ). A abertura da válvula de ar fica restrita ao controlador não podendo ser ajustada pelo usuário.

- Rotina 6 – Envio de e-mail: esta rotina é executada quando o botão *Enviar E-mail* for ativado. Neste caso o software irá agrupar, em um novo arquivo de dados com extensão .zip, todos os arquivos gerados até o momento e o enviará para o endereço de e-mail do usuário.
- Rotina 7 – Comunicação TCP/IP: esta rotina realiza a troca de dados entre o CLP e o software LabVIEW™ através do protocolo de comunicação TCP/IP. O CLP envia os sinais de temperatura dos transmissores de temperatura TT1 e TT2 ao software LabVIEW™ e recebe deste os sinais responsáveis pela abertura da válvula de ar, pela abertura do damper e o sinal de controle do triac.
- Rotina 8 – Verificação de entrada em modo de resfriamento: esta rotina monitora o tempo de execução do experimento e aciona o modo de operação “Resfriamento” quando faltarem dez minutos para o encerramento do tempo alocado para o usuário.
- Rotina 9 – Skype: esta rotina irá configurar o programa Skype™ para que o mesmo realize uma vídeo chamada para o endereço do usuário Skype, fornecido durante o processo de cadastramento ou após a confirmação do login.
- Rotina 10 – Login: esta rotina é executada sempre que o programa detectar a presença de um novo usuário solicitando conexão. Neste caso o programa irá confrontar os dados de entrada com os dados armazenados no arquivo de login, gerado pelo programa de cadastro. Após a confirmação do login, os dados do usuário serão encaminhados ao programa principal para que este possa fazer uso dos mesmos, bem como gerenciar o tempo de acesso permitido para a execução do experimento.
- Rotina 11 – Monitoração dos usuários conectados ao WebLab: esta rotina verifica se há mais que um usuário tentando se logar ao experimento. Em

caso afirmativo o sistema irá iniciar uma contagem regressiva de trinta segundos e após este tempo, se não ocorrer o login, o sistema irá derrubar (cortar a conexão) de todos os usuários a ele conectados. Se um usuário, que já está logado, perder a conexão com o sistema, não será necessário que o mesmo refaça o processo de login, a menos que exista outro usuário solicitando conexão.

- Rotina 12 – Avaliação do sistema: esta rotina foi incluída para permitir que o usuário faça uma avaliação geral do sistema através de um questionário que inclui um total de nove perguntas. O resultado da avaliação é armazenado em um arquivo texto que se encontra no diretório “\Avaliação” do servidor do experimento. O nome do arquivo texto gerado tem a seguinte formatação: “usuário\_data agendada\_hora agendada.txt”. A Figura 24 exibe, através do bloco de notas do Windows, a estrutura do arquivo texto gerado no processo de avaliação do sistema.

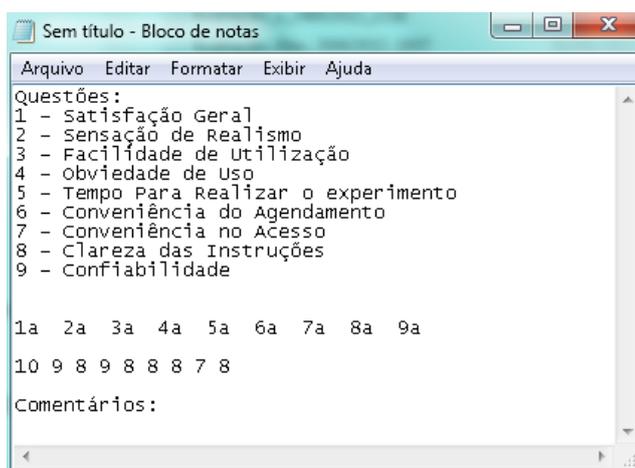


Figura 24 – Estrutura do arquivo texto gerado durante o processo de avaliação do sistema.

- Rotina 13 – Atividades didáticas e instruções: esta rotina inclui a exibição de textos e figuras que orientam o usuário na realização das atividades didáticas sugeridas bem como sobre as instruções de operação de todo o sistema.

### 3.5.10 Aspectos de segurança

O gerenciador interno de usuários remotos do LabVIEW transfere o controle do software aos usuários que se conectam à página do experimento por ordem de

chegada. Desta forma, em caso de falha de conexão durante uma sessão, o controle do sistema poderia ser automaticamente transferido a um usuário não autorizado, caso haja algum que tenha solicitado o controle e esteja aguardando pelo mesmo. Para contornar este problema, sempre que o usuário conectado muda (isto pode ser detectado através de uma rotina específica, que identifica os computadores remotos por IP e User ID no sistema operacional remoto), o sistema solicita novamente os dados de *login* compatíveis com a sessão corrente, devendo esta informação ser corretamente inserida em até 30 segundos (e necessariamente antes do acesso ao WebLab ser concedido); senão, o controle retorna para o servidor. Como é provável que o usuário agendado, que tem a senha correta para o horário, buscará retomar de imediato o controle do experimento, o sistema não interrompe o registro de dados em andamento, dando oportunidade de uma retomada praticamente sem prejuízo. Para minimizar a chance de que este usuário seja prejudicado por uma improvável (mas possível) 'fila de intrusos' tentando tomar o controle do experimento, a não entrada de uma senha correta no prazo de 30 segundos pelo primeiro desta 'fila' acarreta a desconexão de todos os usuários conectados (inclusive os observadores) à página do experimento, o que facilita o retorno do usuário 'oficial'. Somente uma ação coordenada de vários usuários tentando acesso indevido ao sistema poderia comprometer uma sessão. O registro das informações dos usuários remotos que tentam se conectar, em um arquivo '.log' mantido no servidor poderia servir para uma eventual investigação de suas identidades. Entretanto, assim como qualquer sistema que opera sobre a rede da Internet, o WebLab não pode ser considerado imune à ação de conhecedores de técnicas avançadas de *hacking*, nem mesmo com a habilitação de criptografia SSL no LabVIEW WebServer.

### 3.5.11 Tela de Login

Ao conectar-se ao software de gerenciamento do experimento, o usuário será direcionado para a tela de Login. Esta interface possui dois campos para que o usuário forneça o nome e a senha definidos durante a etapa de cadastramento do usuário, conforme exibido na Figura 25. Caso o sistema detecte que dois ou mais usuários estejam conectados a esta interface, será disponibilizado um tempo de 30 segundos para a entrada correta do login e da senha. A temporização poderá ser

acompanhada através do relógio temporizador exibido no canto inferior esquerdo desta interface, conforme exibido na Figura 25.



Figura 25 – Interface de Login do software de gerenciamento do WebLab.

### 3.5.12 Tela de Controle

Após a realização do login, o usuário é redirecionado para a tela principal do software de gerenciamento do WebLab, conforme exibido na Figura 26. Nesta tela, denominada de interface de controle, o usuário poderá monitorar e ajustar as diferentes variáveis envolvidas na operação do sistema.

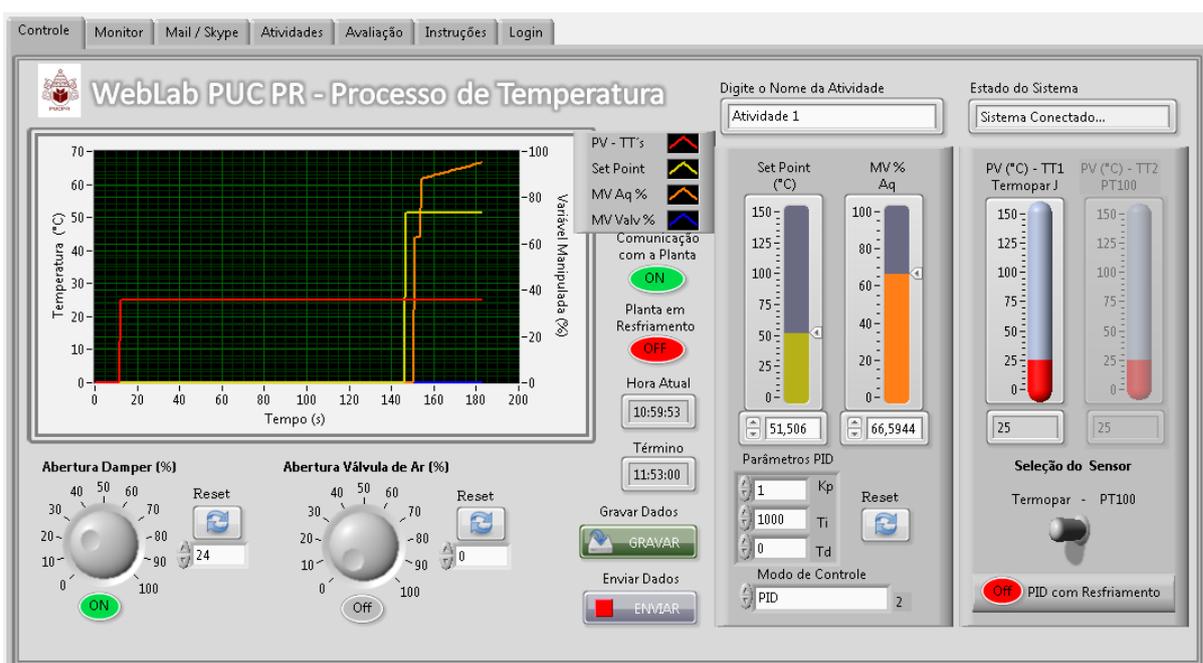


Figura 26 – Interface de Controle do software de gerenciamento do WebLab.

Os itens monitorados através da interface de controle são:

- Temperatura do forno medida pelo sensor termopar tipo J e transmitida pelo transmissor de temperatura TT1;
- Temperatura do forno medida pelo sensor PT100 e transmitida pelo transmissor de temperatura TT2;
- Ações executadas pelo software (campo 'Estado do sistema');
- Referência de temperatura desejada (Set Point °C);
- Sinal de saída do controlador (MV % Aq);
- Hora atual e hora de término do experimento;
- Sinalização de operação em modo de resfriamento do forno;
- Sinalização de operação em modo de controle "PID com resfriamento";
- Sinalização de comunicação com a planta (comunicação entre o CLP e o Servidor);
- Monitor gráfico exibindo: a temperatura de um dos sensores (Termopar tipo J - TT1 ou PT100 - TT2), o sinal de referência de temperatura (set point °C), o sinal de saída do controlador (MV % Aq) e o sinal de saída destinado a abertura da válvula de ar (MV Valv %);

- Status dos botões *Gravar Dados* e *Enviar Dados*.

As ações que o usuário poderá realizar diretamente na interface de controle são:

- Selecionar o sensor de temperatura desejado através de chave seletora;
- Selecionar o modo de operação do controlador (Manual, On-Off, PID ou PID com Resfriamento);
- Ajustar o valor percentual do sinal de saída do controlador quando este operar em modo Manual (MV % Aq);
- Ajustar o valor da referência de temperatura desejada (set point) quando o controlador estiver operando em modo “On-Off”, modo “PID” ou em modo “PID com Resfriamento”;
- Ajustar o valor da abertura percentual do damper (Abertura Damper %);
- Resetar (zerar) o ajuste do valor da abertura percentual do damper;
- Ajustar o valor da abertura percentual da válvula de ar (Abertura Válvula %);
- Resetar (zerar) o ajuste do valor da abertura da válvula de ar;
- Setar os parâmetros do controlador PID (ganho proporcional Kp, tempo integral Ti e tempo derivativo Td);
- Resetar os parâmetros do controlador PID para: Kp(ganho proporcional)=1, Ti(tempo integral)=1000 e Td (tempo derivativo)=1;
- Atribuir um nome para a atividade que esta sendo realizada;
- Ativar a gravação dos dados coletados através do botão *Gravar Dados* ;
- Ativar a rotina de envio de envio dos dados gravados através do botão *Enviar Dados*.

### **3.5.13 Tela de Monitoração**

A segunda opção de tela disponível no software de gerenciamento do WebLab, denominada de interface de monitoração, é apresentada na Figura 27. Como o próprio nome sugere, este ambiente destina-se apenas a monitoração das variáveis envolvidas no sistema não sendo permitido nenhum tipo de ação por parte

do usuário. O diferencial desta interface está no tamanho do monitor gráfico que permite melhor visualização dos sinais monitorados (valor da temperatura dos sensores termopar tipo J e PT100 transmitidos pelos transmissores TT1 e TT2, set point e sinal de controle). O Nome do arquivo criado para armazenar os dados do experimento será exibido no campo “Arquivo de Dados” localizado na parte superior desta tela.

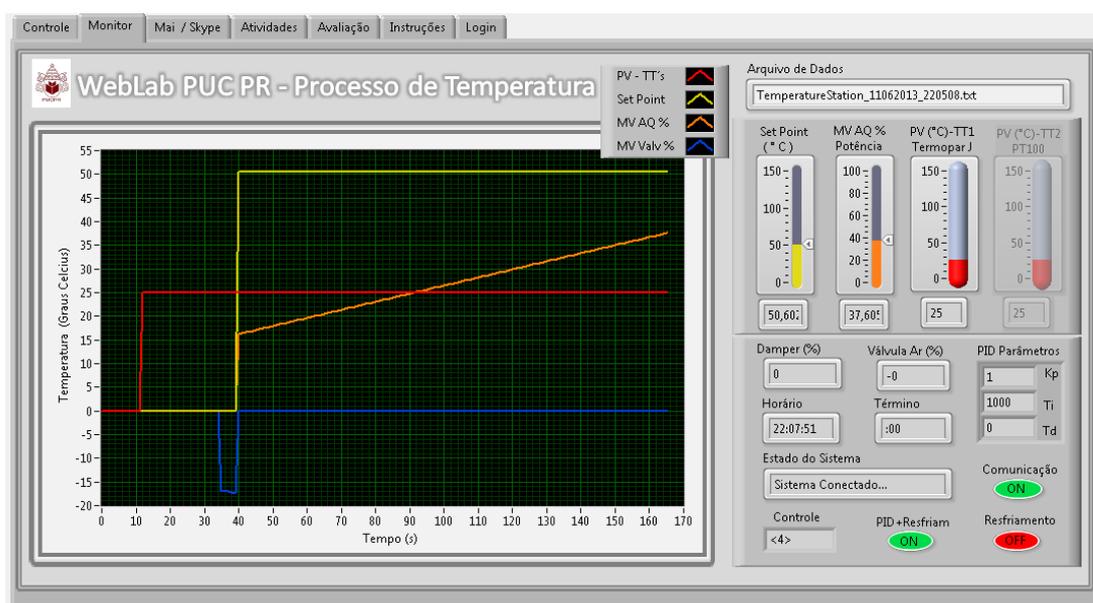


Figura 27 – Interface de Monitoração do Sistema.

### 3.5.14 Tela de Comunicação

A terceira tela, denominada de interface de comunicação e exibida através da Figura 28, disponibiliza ao usuário o serviço de correio eletrônico e o serviço de vídeo chamada, através do programa Skype™.

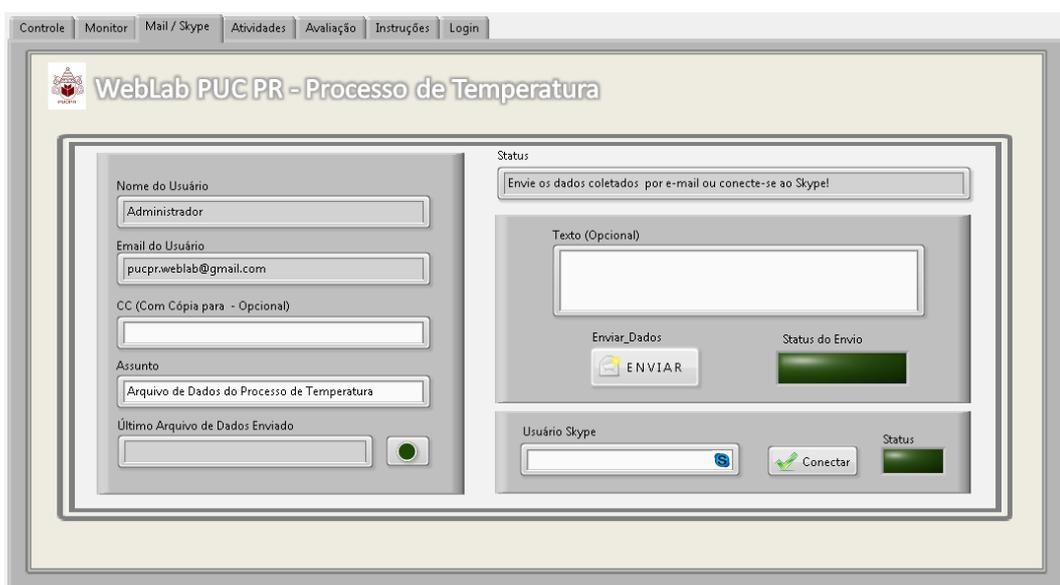


Figura 28 – Interface de Comunicação.

O correio eletrônico permite que os dados coletados durante o experimento sejam encaminhados para o endereço de e-mail fornecido pelo usuário durante a etapa de cadastramento. Observa-se que não é necessário anexar os arquivos de dados gerados durante a execução do experimento, pois o sistema irá agrupá-los automaticamente em um único arquivo com extensão .zip ao término do experimento. O nome do arquivo compactado a ser enviado ao usuário terá a seguinte formatação: *TemperatureStation\_data\_hora.zip*.

Para ativar o serviço de vídeo chamada o usuário deverá:

- Estar logado no programa Skype™;
- Digitar um nome de usuário válido no campo “Usuário Skype” conforme exibido na Figura 28 (este nome é importado dos dados fornecidos durante a etapa de cadastramento do usuário mas poderá ser alterado durante a execução do programa);
- Ativar o botão *Conectar* conforme exibido na Figura 28;
- Aguardar o pedido de confirmação de chamada gerado pelo programa Skype™;
- Aceitar o pedido de confirmação de chamada gerado pelo programa Skype™.

A finalização da vídeo chamada poderá ser feita através da opção de encerramento de chamada do programa Skype™. Caso o usuário não finalize a vídeo chamada, esta será encerrada automaticamente ao final do tempo agendado para a execução do experimento, ou ao final da execução da rotina de finalização do sistema. Após o sistema entrar no modo de resfriamento não será permitido a ativação da vídeo chamada. A Figura 29 exibe uma imagem do WebLab exibida pelo programa Skype, com o detalhe do ângulo de visão da webcam.



Figura 29 – Imagem do WebLab transmitida através do programa Skype™.

### 3.5.15 Tela de Atividades Didáticas

A quarta tela, denominada de interface de orientações sobre as atividades didáticas, destina-se a orientar o usuário durante a realização de uma das três atividades didáticas oferecidas (ver seção 3.2.1). Para cada atividade didática são fornecidos os objetivos propostos, uma breve discussão sobre o tema, uma listagem dos equipamentos envolvidos e a sequência de passos a ser executada para a realização do experimento. A Figura 30 exibe a tela utilizada para a visualização das atividades didáticas.

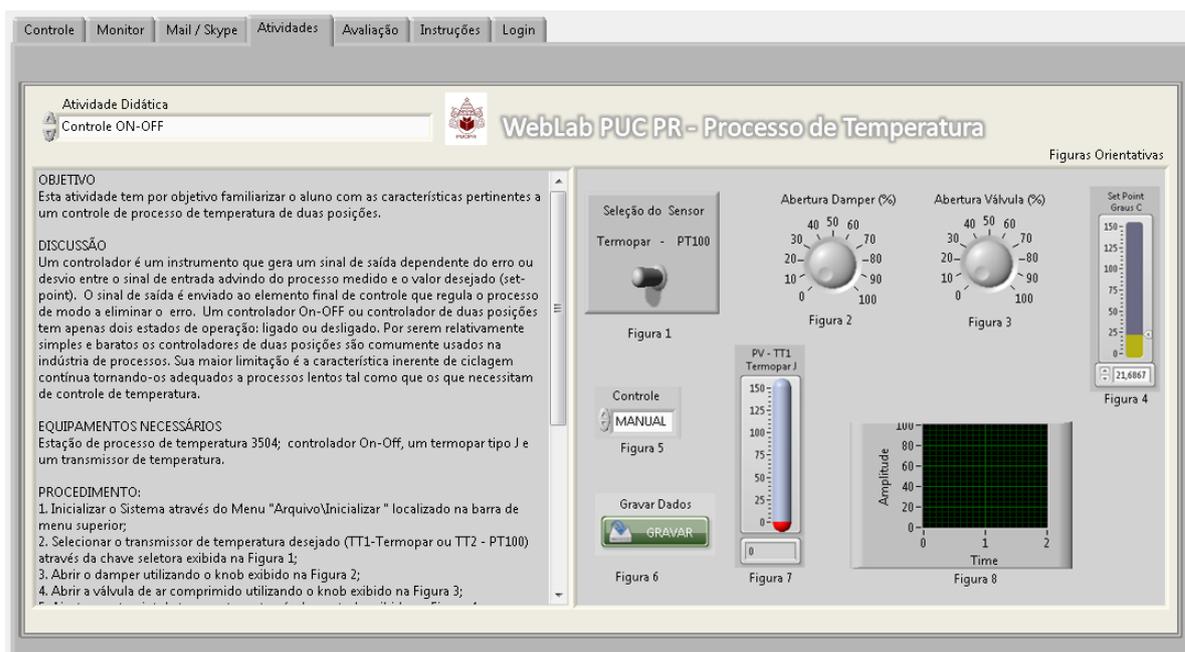


Figura 30 – Interface de Orientações sobre as Atividades Didáticas.

### 3.5.16 Tela de Avaliação (*feedback*) pelo Usuário

A quinta tela, intitulada interface de avaliação, é exibida através da Figura 31. Este ambiente tem a finalidade de permitir que o usuário possa emitir a sua opinião sobre a operacionalidade do sistema de um modo geral. Os resultados das avaliações servirão para realimentar o processo de melhoria do sistema. A avaliação está elaborada em forma de questionário e inclui um total de nove perguntas baseadas no trabalho de Corter et al. (2004). As respostas são feitas através da atribuição de uma pontuação, numa escala crescente de 1 a 10, onde 10 representa a pontuação máxima. As questões que compõem o questionário são listadas a seguir:

- Satisfação geral;
- Sensação de realismo;
- Facilidade de uso;
- Obviedade de uso;
- Tempo para realizar o experimento;
- Conveniência do agendamento;
- Conveniência no acesso;

- Clareza das Instruções;
- Confiabilidade.

Figura 31 – Interface de Avaliação do sistema.

### 3.5.17 Tela de Instruções

A tela denominada ‘interface de instruções’ tem a finalidade de fornecer informações sobre a funcionalidade do sistema e sobre as ferramentas disponibilizadas para a execução das tarefas. Os detalhes desta tela podem ser visualizados através da Figura 32.

Figura 32 – Interface de Instruções.

#### 4.0 INTEGRAÇÃO E TESTES

Os subsistemas foram testados de forma individualizada a fim de comprovar que as características de operação e segurança desejadas foram alcançadas separadamente. Os seguintes testes de integração foram realizados:

- Teste de comunicação entre o CLP e o circuito microcontrolado a fim de ajustar a conversão dos sinais (CLP – microcontrolador);
- Teste de comunicação entre o CLP e os transmissores de temperatura;
- Teste de acionamento da válvula proporcional de ar comprimido;
- Teste de acionamento do motor de passo a fim de ajustar: a direção de deslocamento, a posição dos sensores fim de curso, o posicionamento inicial e a velocidade de deslocamento do damper;
- Teste de comunicação entre Servidor e CLP checando a integridade do envio e recebimento dos dados;
- Teste do sistema de intertravamento para garantir que não ocorra simultaneidade de operação do sistema (local / remota);
- Teste de segurança da rotina de gerenciamento de falha de comunicação entre o servidor e o CLP para verificar se a válvula de ar é fechada;
- Teste de segurança para verificar a funcionalidade da botoeira de emergência e do botão Liga/Desliga;
- Teste individualizado de todas as rotinas implementados nos softwares de cadastramento dos usuários e gerenciamento do WebLab;

Ao término dos testes individuais, os subsistemas foram integrados.

#### 4.1 EXEMPLOS DE REALIZAÇÃO DAS ATIVIDADES DIDÁTICAS

Como parte dos testes de integração, executaram-se sessões-piloto das três atividades didáticas sugeridas (ver item 3.2.1). Registros exemplificando a execução das três atividades didáticas propostas são apresentados a seguir, através de gráficos que expressam o histórico ao longo do tempo dos valores para *set-point* e

valor instantâneo da variável de processo (temperatura), e das variáveis manipulada (percentuais da potência de aquecimento e da abertura da válvula de admissão de ar). Estes ensaios não têm por objetivo obter um desempenho elevado do sistema, senão que comprovar que as funcionalidades do WebLab operam adequadamente e permitem produzir resultados que serão interpretados pelos usuários e usados na produção de relatórios.

### I) ATIVIDADE DIDÁTICA 1: Controle ON-OFF

Condições iniciais:

- Set Point: 50°C (degrau);
- Abertura do Damper: 99%;
- Abertura inicial da Válvula de ar: 35%;
- Sensor utilizado: Pt100.

A Figura 33 exibe o histórico da atividade didática 1. Note-se o aumento da taxa de serviço média da potência de aquecimento após o aumento da abertura da válvula de admissão de ar.

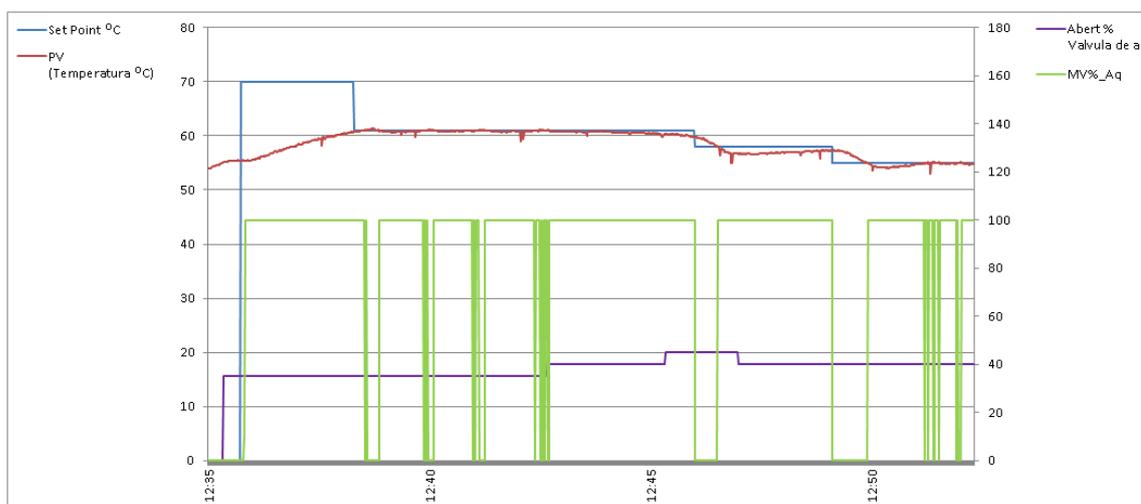


Figura 33 – Gráfico contendo o histórico de uma sessão em que foi realizada a Atividade Didática 1.

### II) ATIVIDADE DIDÁTICA 2: Controle PID

Condições iniciais:

- Set Point: 42°C;
- Abertura do Damper: 100%;
- Abertura inicial da Válvula de ar: 30%;
- Sensor utilizado: Termopar tipo J;

- $K_p=2$  ,  $T_i=50$  ,  $T_d=0$ .

A Figura 34 exibe o histórico da atividade didática 2. Note-se a atuação combinada do aquecimento e da admissão de ar com a finalidade de conduzir a temperatura ao *set point* especificado.

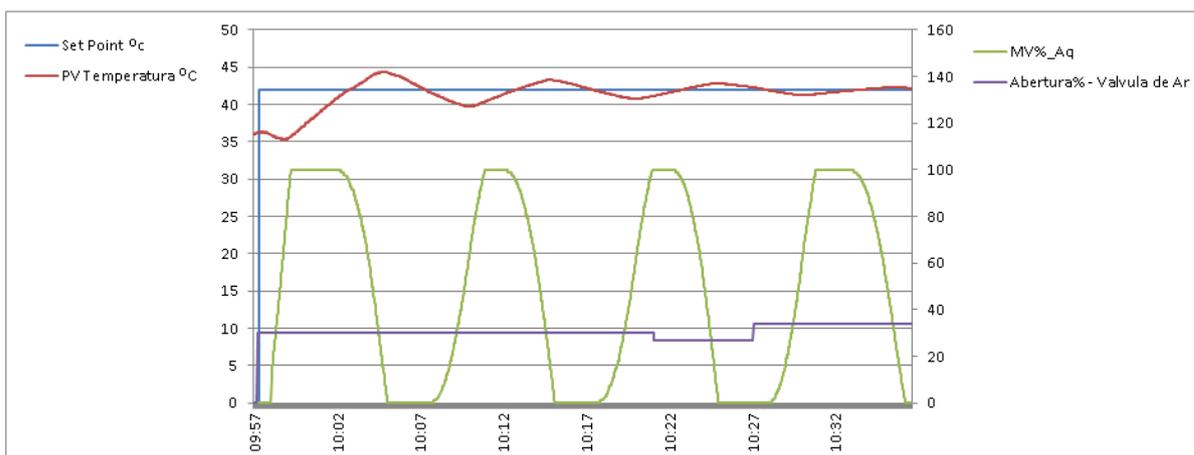


Figura 34 – Gráfico contendo o histórico de uma sessão em que foi realizada a Atividade Didática 2.

### III) AVIDADE DIDÁTICA 3: Aplicando um degrau ao sistema

Condições iniciais:

- Abertura do Damper: 0%;
- Abertura de Válvula de ar: 0%;
- Valor do degrau aplicado à variável manipulada (MV% Aq): 50%;
- Sensor utilizado: Pt 100.

A Figura 35 exibe o histórico da atividade didática 3. Algoritmos de análise podem ser aplicados à curva de resposta do sistema com a finalidade de determinar seu modelo dinâmico nas condições de operação estabelecidas inicialmente.

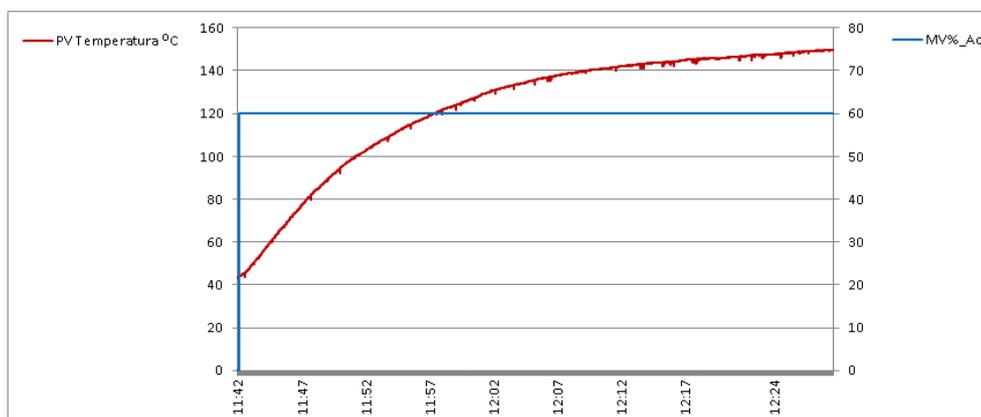


Figura 35 – Gráfico contendo o histórico de uma sessão em que foi realizada a Atividade Didática 3.

A título de exemplificar a produção de um relatório, a formatação dos gráficos e tabelas relacionados a cada experimento foi feita com o auxílio do programa Microsoft Excel 2010. Os dados armazenados foram importados através da opção “De Texto”, localizada na aba “Dados” do Excel, conforme exibido na Figura 36.

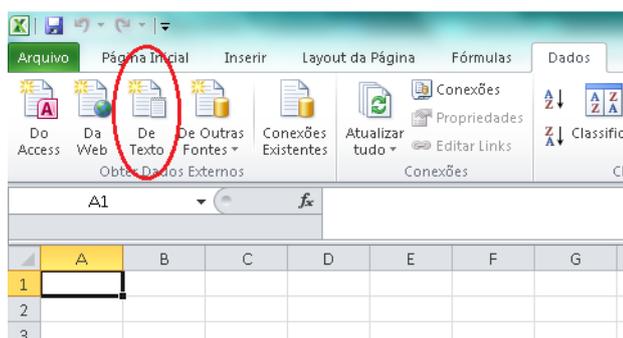


Figura 36 – Aba Dados do Programa Microsoft Excel 2010.

A configuração do Assistente de importação permite que se defina a partir de qual linha os dados do arquivo texto serão importados e que se indique a largura dos campos, ou quebra de colunas entre os dados importados. O resultado da formatação pode ser conferido através da Figura 37.

DETALHE DOS DADOS COLETADOS DURANTE A REALIZAÇÃO DA ATIVIDADE 2: CONTROLE PID													
Hora	Tempo_(ms)	Set Point (°C)	Pt100 (°C)	Tempo-J(°C)	MV%_Aq	Damper(%)	Válvula(%)	Kp	Ti	Td	Controle	Sensor_Utilizado	Atividade
16:40:02	23086282	43	39,74	34,56	100	100	30	3	50	0	Manual	PT100	Atividade_2
16:40:03	23087282	43	39,76	34,57	100	100	30	3	50	0	Manual	PT100	Atividade_2
16:40:04	23088282	43	39,77	34,58	100	100	30	3	50	0	Manual	PT100	Atividade_2
16:40:05	23089282	43	39,79	34,62	100	100	30	3	50	0	Manual	PT100	Atividade_2
16:40:06	23090282	43	39,80	34,63	100	100	30	3	50	0	Manual	PT100	Atividade_2
16:40:07	23091282	43	39,81	35,01	100	100	30	3	50	0	Manual	PT100	Atividade_2
16:40:08	23092282	43	39,82	35,02	100	100	30	3	50	0	Manual	PT100	Atividade_2
16:40:09	23093282	43	39,83	35,02	100	100	30	3	50	0	Manual	PT100	Atividade_2

Figura 37 – Detalhe dos Dados Formatados no Programa Microsoft Excel 2010.

## 5 RESULTADOS E DISCUSSÃO

### 5.1 TESTES DE UTILIZAÇÃO POR UM GRUPO DE USUÁRIOS PILOTO

Após a finalização das etapas de desenvolvimento e testes do WebLab para uma planta de processo de temperatura, iniciaram-se os testes de operação do sistema. Desta forma, um grupo de dezesseis usuários-piloto, dentre eles estudantes e professores, foi convidado a utilizar o sistema para que se pudesse coletar informações sobre as suas impressões acerca da experiência com o WebLab.

A compilação dos resultados é exibida através da Figura 38, onde são apresentados a média e o desvio padrão para os nove itens avaliados. Verifica-se que mesmo os itens com pontuação mais baixa - sensação de realismo (média 8,4), tempo para realizar o experimento (média 8,5) e facilidade de uso (média 8,9) – indicam uma aprovação da experiência de uso pelo grupo.

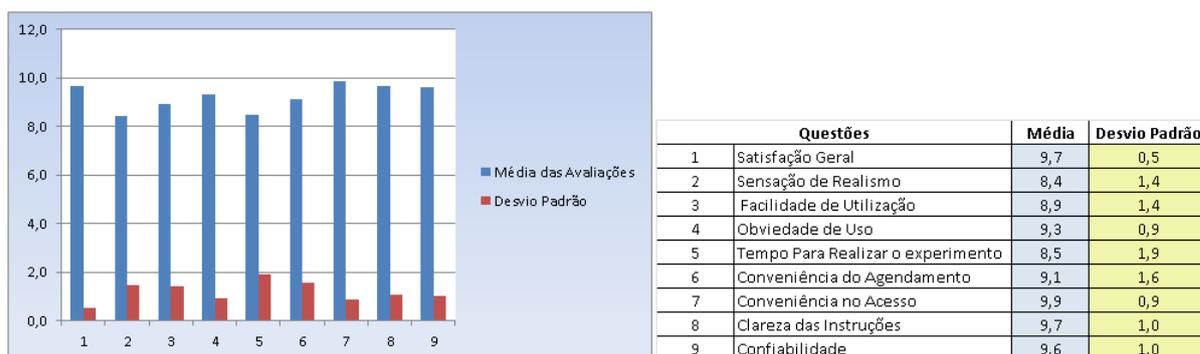


Figura 38 – Avaliação do Sistema pelo Grupo de Usuários Piloto.

### 5.2 ANÁLISE DO ATENDIMENTO DAS ESPECIFICAÇÕES

A Tabela 10 apresenta, de forma sintetizada, a avaliação dos resultados obtidos frente às especificações de projeto estipuladas no item 3.2.3. Conforme se pode verificar nessa tabela, as especificações foram suficientemente atendidas.

Tabela 10 – Resultados Obtidos Versus Especificações Técnicas.

ESPECIFICAÇÕES TÉCNICAS	METAS	DESEMPENHO	DISCUSSÃO
Quantidade de operações do usuário	Máximo 6 etapas	Suficiente 5 etapas	Um experimento pode ser realizado com a execução das 5 etapas: <i>login</i> ; escolha do experimento; leitura de instruções; configuração dos parâmetros e execução do experimento
Quantidade de pontos de visualização por câmeras	1	Suficiente 1 ponto	Utilizou-se apenas uma câmera com microfone com o intuito de monitorar a movimentação do damper e o som provocado pela saída de ar comprimido.
Quantidade de Pontos de interação	Mínimo 3	Suficiente 3 pontos	O sistema contempla: indicadores visuais, gráficos, mensagens de texto.
Grau de detalhamento das instruções	Mínimo 1	Suficiente Grau 2	O grau de informação dada ao usuário para orientar suas ações durante a realização de uma atividade didática foi 2, significando: uso de texto e figuras como meio de orientação.
Instrumentos de coleta de opinião	Mínimo 1	Suficiente 1 Instrumento	O sistema contempla um questionário como meio de coleta de opinião
Número de sistemas operacionais	1	Suficiente 1	Apenas o sistema operacional Windows foi contemplado
Espaço local de armazenamento de dados	Mínimo 10GBytes	Suficiente HD 500GB	Utilizou-se os recursos disponíveis de hardware suficientes
Número de repetições do experimento	Mínimo 1	Suficiente	O usuário poderá repetir um experimento desde que o limite de tempo agendado não seja ultrapassado
Largura de banda de rede	Mínimo 300 kbps	Medição da velocidade da rede	Velocidade média de 70 Mbps
Índice de disponibilidade do sistema	14h a cada 24h (58%)	(não se aplica)	A taxa/índice de disponibilidade efetiva somente poderia ser avaliada após um suficiente tempo de utilização, o que requer que uma demanda compatível tenha se formado.
Número de telas	Máximo 2	Suficiente 1 tela	Ao passar pela tela de <i>login</i> o usuário já terá acesso à tela de controle
Índice de segurança	1	Suficiente 1 Baixo (conforme a escala definida)	Não existe risco de danos físicos causados ao usuário em decorrência de uma operação errônea

Tipos de dispositivos diferentes de acesso	Mínimo 1	Suficiente 1	Adotou-se apenas a plataforma PC
Blocos de tempo alocados para o usuário	Máximo 1	Suficiente 1 Bloco	Implementado conforme a especificação
Tempo de resposta do sistema	Máximo 2s	Suficiente	Com base na experiência de utilização
Resolução da medida de temperatura	1°C	Suficiente para até 0,0061°C	Como demonstrado no item 3.3.6 a resolução especificada para a leitura de temperatura foi de 1°C, bem abaixo da resolução disponível
Resolução da atuação de potência (resfriamento/aquecimento)	1%	Suficiente para até 0,003%	Como demonstrado no item 3.3.6 a resolução da saída de corrente é de 0,49uA por <i>step</i> , muito maior do que a resolução de 1% (0,16mA) especificada

### 5.3 ASPECTOS GERAIS DO DESENVOLVIMENTO DE SOFTWARE

A programação em LabVIEW foi feita com base em uma arquitetura de múltiplos loops paralelos (*threads*), produtores ou consumidores de dados, que se coordenam entre si através de filas de mensagens. Tarefas distintas como disparar ações a partir de entradas (eventos) do usuário, manter a comunicação com os elementos do sistema e executar algoritmos de controle, entre outras tarefas (ver Figura 23), foram assim mantidas independentes, de modo a assegurar a responsividade da aplicação de software como um todo.

O sistema de agendamento implementado em LabVIEW não permite o acesso simultâneo de múltiplos usuários, de forma que uma rotina de limitação de tempo de uso deste sistema é ativada quando há usuários aguardando na fila para utilizá-lo. Este sistema foi desenvolvido com a adaptação e expansão de funcionalidades de um outro pré-existente. Ainda, para recuperar uma sessão em andamento em caso de perda da conexão, foi implementada uma rotina específica de gerenciamento de usuários. Como estas situações atípicas não foram vivenciadas pelo grupo de usuários que fizeram testes de utilização do WebLab, estes aspectos não foram considerados naquelas opiniões retornadas.

A implementação da rotina de videoconferência via Skype exigiu o estudo e a modificação de um conjunto de VI's contendo métodos específicos deste programa

na linguagem LabVIEW, de modo a permitir a abertura automática de uma chamada de áudio e vídeo, para o usuário Skype cadastrado por ora do agendamento, assim que o usuário inicia sua sessão.

A integração do CLP requereu a sua programação através do B&R Automation Studio, de modo a estabelecer a transmissão continuada de valores de parâmetros e variáveis de/para a planta física, via TCP-IP. Esta implementação mostrou-se bastante estável.

Tanto a programação em LabVIEW quanto a feita na linguagem Automation Studio foram objetos de curvas de aprendizado por parte deste autor e de alunos colaboradores, do que ostensivos trabalhos de depuração e aperfeiçoamento de códigos foram continuamente necessários até se alcançar os resultados desejados. Ainda, foi criado um programa para o microcontrolador (firmware) usado no circuito de controle do motor de passo do mecanismo do *damper*, em linguagem C, que foi apropriadamente compilado antes da sua gravação no dispositivo alvo. Por se tratar de uma linguagem clássica e uma tecnologia bem dominada pela equipe, esta foi implementada sem as dificuldades reportadas nos demais casos. De modo geral, a criação, depuração e integração de software requereu tempo e esforços que não poderiam ser desvinculados da experiência prévia dos programadores, com reflexos diretos no tempo de desenvolvimento do projeto, haja vista a dimensão da aplicação necessária ao funcionamento do WebLab.

## **6 CONCLUSÃO E CONSIDERAÇÕES FINAIS**

A adoção do modelo sistemático PDW proposto por Mendes et al. (2010), empregando as ferramentas denominadas ‘análise funcional’, ‘matriz morfológica’ e ‘método de Pugh’, permitiu estruturar e bem organizar as etapas de concepção e desenvolvimento deste trabalho.

A priorização dos requisitos proporcionada pela aplicação da Casa da Qualidade do método QFD forneceu um importante indicativo sobre como devem ser destinados os recursos a serem investidos no desenvolvimento do sistema, de forma a se maximizar a satisfação do usuário.

O atendimento às especificações de projeto alcançado com a solução resultante da aplicação do PDW e a avaliação positiva da utilização do sistema pelo grupo de usuários suportam-se mutuamente.

De um modo geral, o modelo PDW cumpriu um importante papel para orientar as etapas iniciais de estruturação e desenvolvimento do WebLab para uma planta de temperatura - sobretudo na etapa inicial do projeto.

Da experiência do desenvolvimento, foram inferidas contribuições para o aprimoramento do modelo de referência utilizado (PDW):

1. Uma primeira sugestão, ainda que de caráter geral e não limitada somente a projetos de WebLabs, decorre do fato que, em um cenário de recursos limitados, a utilização dos materiais disponíveis para o projeto torna-se imperativa, e apesar de esses representarem frequentemente soluções técnicas superdimensionadas, mas suficientes, e economicamente viáveis (justamente por estarem disponíveis), estas prevalecem sobre as que poderiam ser ditas 'mais racionais'; desta forma, pode-se ter no PDW uma indicação para que tais recursos sejam tomados como requisitos na forma de restrições no início do projeto;
2. Em um WebLab interativo, devem estar presentes soluções para a retomada segura da sessão em caso de interrupção da conexão de dados, para minimizar os riscos de o usuário não concluir seu trabalho durante a sua janela de tempo agendada e ser obrigado a um inconveniente adiamento;
3. Incluir a sugestão de que o sistema WebLab possibilite que ações administrativas sobre o sistema sejam tomadas à distância, de modo que se possa recolocar o sistema em funcionamento em casos de excepcionalidades, sem que seja necessário deslocamento do responsável até o laboratório;
4. É importante a existência de um canal para feedback das opiniões do usuário;

5. Os dispositivos acrescentados com a automação do sistema não devem criar dificuldades de acesso aos componentes pré-existentes no sistema físico, em futuras atividades de manutenção;
6. Possibilitar a eventual desconexão dos sistemas do WebLab para uso do equipamento experimental na sua forma original, se for o caso.

## 6.1 SUGESTÕES PARA TRABALHOS FUTUROS

Como sugestões para trabalhos futuros, em continuação ao desenvolvimento do WebLab da planta de temperatura, tem-se:

- Avaliar a utilização desde outros sistemas operacionais;
- Realização de testes de desempenho de acesso desde computadores ligados à rede com diferentes velocidades de conexão;
- Aprimoramento dos serviços de agendamento / verificação de utilização e do armazenamento de resultados, integrando ao sistema um banco de dados.

#### 4 REFERÊNCIAS

Alliet-Gaubert, M. et al., “Cooperative WebLab in chemical engineering between France and Brazil: Validation of the methodology”, Online journal: Education for Chemical Engineers”, Journal of Science Education and Technology, Volume 7, Issue 1, Pages e7–e13, January 2012.

Andria, G. et al., “Remote Didactic Laboratory ‘G. Savastano,’ Italian Experience for e-Learning,” IEEE Trans. Instrumentation and Measurement, Vol. 56, No. 4, pp. 1135-1147, August 2007.

Asimov, M., “Introduction to design: fundamentals of engineering design”, Prentice Hall, New Jersey, 1962.

ASME RESEARCH, “Design theory and methodology: a new discipline”, Mechanical Engineering, pp.23-27, 1986.

Astrom, K., J., Hagglund, T., “PID Controllers: Theory, Design and Tuning”. Segunda edição, ISA Press, 1995.

Back, N., “Metodologia de projeto de produtos industriais” , Guanabara Dois, Rio de Janeiro, 1983.

Back, N., Ogliari, A., Dias, A., da Silva, J. C., “Projeto Integrado de Produtos – Planejamento, Concepção e Modelagem”, Ed. Manole, Barueri, 2008.

Bi, Q., W. Cai, E. Lee, Q. Wang, C. Hang, Y. Zhang, “Robust identification of first-order plus dead-time model from step response”, Control Engineering Practice 7, 71-77, Ano 1999.

Brito, N., Ribeiro, P., Soares, F., Monteiro, C., Carvalho, V., Vasconcelos, R., “A Remote System for Water Tank Level Monitoring and Control - a Collaborative Case-

study”, IEEE, Industrial Electronics Department, University of Minho, Guimarães, Portugal, 2009.

Carnevali, G., Buttazzo, G., “A virtual laboratory environment for real-time experiments.”, In SISICAS, July 2003.

Castro, R., “Educação a distância na sociedade Informática: desafios na formação continuada de professores do ensino básico – Projeto Ler e Viver”, Dissertação (Mestrado) Curso de Pós-Graduação em Educação. Pontifícia Universidade Católica de Campinas, 2008.

Cheng, L.C.; Melo Filho, L.D.R., “Desdobramento da função qualidade na gestão do desenvolvimento de produtos.”, São Paulo, Editora Blücher, 2007.

Cmuk, D., Mutapcic, T., BILIC, I., “MIRACLE - Model for Integration of Remote Laboratories in Courses that Use Laboratory and e-Learning Systems”, IEEE Transactions on Learning Technologies, Vol. 2, No. 4, october-december, 2009.

Corter, J., Nickerson, J. Esche & C. Chassapis, S., K., "Remote versus hands-on labs: A comparative study", Proc. 34th ASEE/IEEE Frontiers in Education Conference, Savannah, GA, F1G17-F1G21, USA, 2004.

CRUZ, A., A., “Projeto e Implementação de um Framework para WebLabs Baseado em Ajax e Padrões de Projeto”, Dissertação (Mestrado em Engenharia Elétrica) - Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas – UNICAMP, Campinas, 2007.

Cross, N., “Engineering Design Methods”, Ed. Wiley, second edition, Chichester, pp. 65-66, pp.108, 1994.

Deley, R., “WebLab de Um Sistema de Energia Renovável”, Trabalho de Conclusão de Curso – (Graduação em Engenharia Mecatrônica - Controle e Automação) - Pontifícia Universidade Católica do Paraná, Curitiba, 2012.

Farias, A. F., “Desenvolvimento de um Web Lab SOA no Domínio de Redes de Computadores”, Dissertação (Mestrado em Ciência da Computação) – Faculdade de Computação, Universidade Federal de Uberlândia – UFU, Uberlândia, 2008.

Fischer, J., Mitchell, R., Alamo, J. D., “Inquiry-Learning with WebLab: Undergraduate Attitudes and Experiences”, Journal of Science Education and Technology, Vol. 16, No. 4, August 2007.

GIL, A.,C., “Como elaborar projetos de pesquisa.”, São Paulo: Atlas, 1991.

Gomes, L., Patrício, G., Ferreira, R., Costa, A., “Remote experimentation for introductory digital logic course”, IEEE, Universidade Nova de Lisboa - Faculdade de Ciências e Tecnologia & UNINOVA – CTS, 2009.

Gravier, C., Fayolle, J., Bayard, B., Ates, M., Lardon, J., “State of the art about remote laboratories paradigms - foundations of ongoing mutations” International Journal of Online Engineering – Vol. 4, Issue 1, Feb 2008.

Khalil, A., Hasna, M., Benammar, M., Chaabane, M., Amar, C., “Development of a Remote Lab for Electrical Engineering Program”, IEEE, College of Engineering, Qatar University, 2009.

Kaminski, B., Kazmierkowski, M., Sobczuk, D., “Remote Experiment System and Internet Based Teaching of PWM Techniques for Three-Phase Voltage Source Converters”, IEEE, Warsaw University of Technology, Institute of Control and Industrial Electronics, 2009.

Ko, C. C., Chen, B. M., Chen, S. H., Ramakrishnan, V., Chen, R., Hu, S. Y., Zhuang, Y., "A large-scale web-based virtual oscilloscope laboratory experiment", Engineering Science and Educational Journal, pp. 69-76, April 2000.

Maga, D1., Sitar, J., Dudak, J., Hartansky, R., Siroky, P., Halgos, J., Bauer, P., "Electromechanical Actuators WEB-lab", IEEE, Faculty of Mechatronics TnUDA, Slovakia, 2008.

Massachusetts Institute of Technology, "MIT iCampus. iLabs: Internet access to real labs - anywhere, anytime", (<http://ilab.mit.edu/iLabServiceBroker/>). Último acesso: 17 de Julho de 2013.

Mendes, L. A., Santos, M. L. D, Siqueira, M. T. C., "Systematization of the WebLabs Development Process: Towards an Approach Proposal", International Conference on Engineering Education ICEE-2010, Gliwice, Poland, 2010.

Minayo, M., "O desafio do conhecimento", São Paulo, Hucitec, 1993.

Moreira, V. R., Cardoso, F. A. C. M., Arantes, D. S., "Plataforma reconfigurável para ensino e pesquisa em laboratório de sistemas digitais à distância", XIX Simpósio Brasileiro de Informática na Educação (SBIE 2008), Campinas, 2008.

Nedic, Z., Machotka, J., Nafalski, A., "Remote Laboratories Versus Virtual and Real Laboratories", 33rd ASEE/IEEE Frontiers in Education Conference, Boulder, 2003.

Nedic, Z., Machotka, "Remote Laboratory NetLab for Effective Teaching of 1st Year Engineering Students", University of South Australia/School of Electrical and Information Engineering, Adelaide, Australia, 2007.

NetLab (<http://netlab.unisa.edu.au/index.xhtml>), University of South Australia. Último acesso: 17 de Julho de 2013.

Ngolo, M. A., Fernandes, “Arquitetura Orientada a Serviços REST para Laboratórios Remotos”, Dissertação (Mestrado em Engenharia Elétrica e de Computadores) – Faculdade de Ciências de Tecnologia, Universidade Nova de Lisboa, Lisboa, 2009.

Ohfuji, T.; Michiteru, O; Akao, Y, “Método de desdobramento da qualidade (1): elaboração e exercício da matriz da qualidade”, Belo Horizonte: Escola de Engenharia da UFMG. Manual de aplicação do desdobramento da função da qualidade, Vol. 1, 1997.

Pahl, G., Beitz, W., “Engineering Design.”, second edition, London: Springer-Verlag, 1996.

Pahl, G., Beitz, W., “Projeto na Engenharia”, primeira edição, Editora Edgard Blucher, 2005.

Paladini, S., “Experimentação Remota como Suporte a Ambientes de Aprendizagem de Física.”, Dissertação (Mestrado em Engenharia e Gestão do Conhecimento) – Universidade Federal de Santa Catarina – UFSC, Florianópolis, 2008.

Pradarelli, B., Latorre, L., Flottes, M., Bertrand, Y., Nouet, P., “Remote Labs for Industrial IC Testing”, IEEE Transactions on Learning Technologies, Vol. 2, No. 4, October-December 2009.

Pereira, T., Sales, B., Souza, D., Machado, L., Gabriel, J. et al., “Laboratórios Remotos com Sistemas Hápticos para Educação à Distância”, X Symposium on Virtual and Augmented Reality, SVR, João Pessoa, PB, 2008.

Pugh, S., “Total Design: Integrated Methods for Successful Product Design.”, Great-Britain: Addison-Wesley Publishing Company, 1991.

RemoteLab FEUP – Porto University, (<http://remotelabsup.fe.up.pt/index.htm>). Último acesso: 17 de Julho de 2013.

RexLab, (<http://rexlab.ararangua.ufsc.br/?q=pt-br/home>), Universidade Federal de Santa Catarina (UFSC). Último acesso: 17 de Julho de 2013.

Rocha, L., A., “WebLab SOA no Domínio de Redes de Computadores para Experimentos Diff/Serv”, Dissertação (Mestrado em Ciência da Computação) – Universidade Federal de Uberlândia – UFU, 2009.

Santos, M. L. D., Siqueira, M. T. C., Mendes, L. A., “Aplicação do QFD no Processo Sistemático de Desenvolvimento de um WebLab para uma Planta de Processo de Temperatura: Identificação dos Requisitos de Usuário e de Projeto”, XVII Simpósio de Engenharia de Produção – SIMPEP, Bauru, SP, 2010.

Santos, M. L. D., “Proposta de um Modelo de Referência para o Desenvolvimento de Projetos de WebLabs”, Dissertação (Mestrado em Engenharia de Produção e Sistemas) – Pontifícia Universidade Católica do Paraná – PUC PR, Curitiba, 2013.

Siqueira, M. T. C., Santos, M. L. D., Mendes, L. A., “Conceptual design of a thermal process plant Weblab”, Remote Engineering and Virtual Instrumentation, REV’12, University of Deusto, Bilbao, Spain, 2012.

TIDIA-KyaTera (2012). Programa de Internet Avançada, FAPESP, Brasil. (<http://www.kyatera.fapesp.br>). Último acesso: 17 de Julho 2013.

VDI 2221, “Methodik zum entwickeln und konstruieren technischer Systeme und Produkte.”, VDI – Verlag, Düsseldorf, 1985.

Zubia, J., Orduña, P., Irurzun, J., Hernández, U., Sancristobal, E., Martín, S., Castro, M., López-de-Ipiña, D., “Towards an Extensible WebLab Architecture”, IEEE, Faculty of Engineering, University of Deusto, Bilbao, Spain, 2009.

WebLab Deusto; (<https://www.weblab.deusto.es/web/>) . Último acesso: 17 de Julho de 2013.

Wolbank, T., Bauer, P., Macheiner, P., Vogelsberger, M., “Distance Laboratory for Teaching Industrial Electronics – II”, IEEE, Department of Electrical Drives and Machines Vienna University of Technology, Vienna, Austria, 2009.

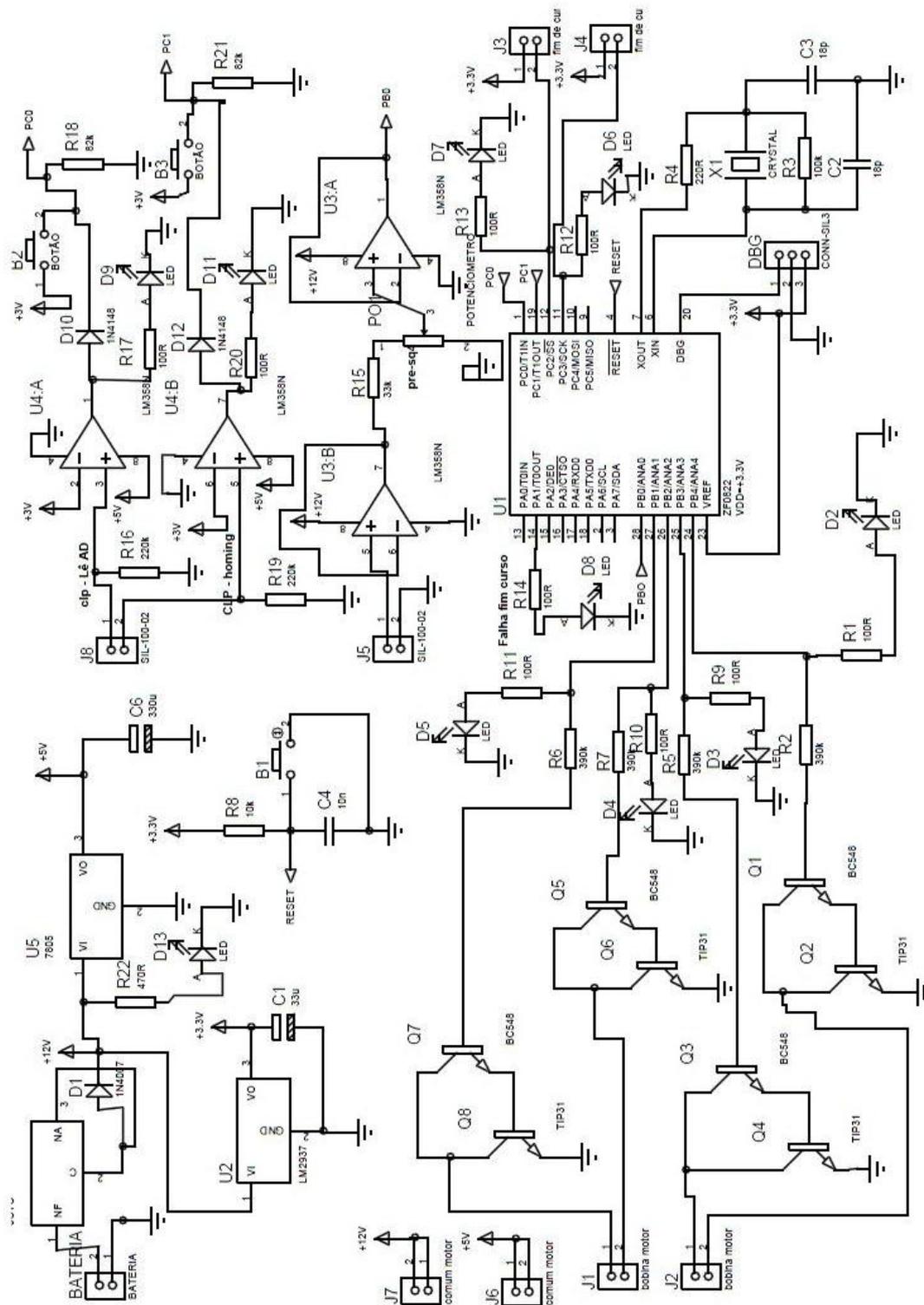
Yin, Robert K., “Estudo de caso.”, Bookman, 2ª edição, 2001.

Yusof, R., Khalid, M., “An Alternative Laboratory Package for Teaching Control Engineering”, IEEE, Center for Artificial Intelligence and Robotics (CAIRO), Universiti Teknologi Malaysia, International Campus, Jalan Semarak, 54100 Kuala Lumpur, Malaysia, 2009.

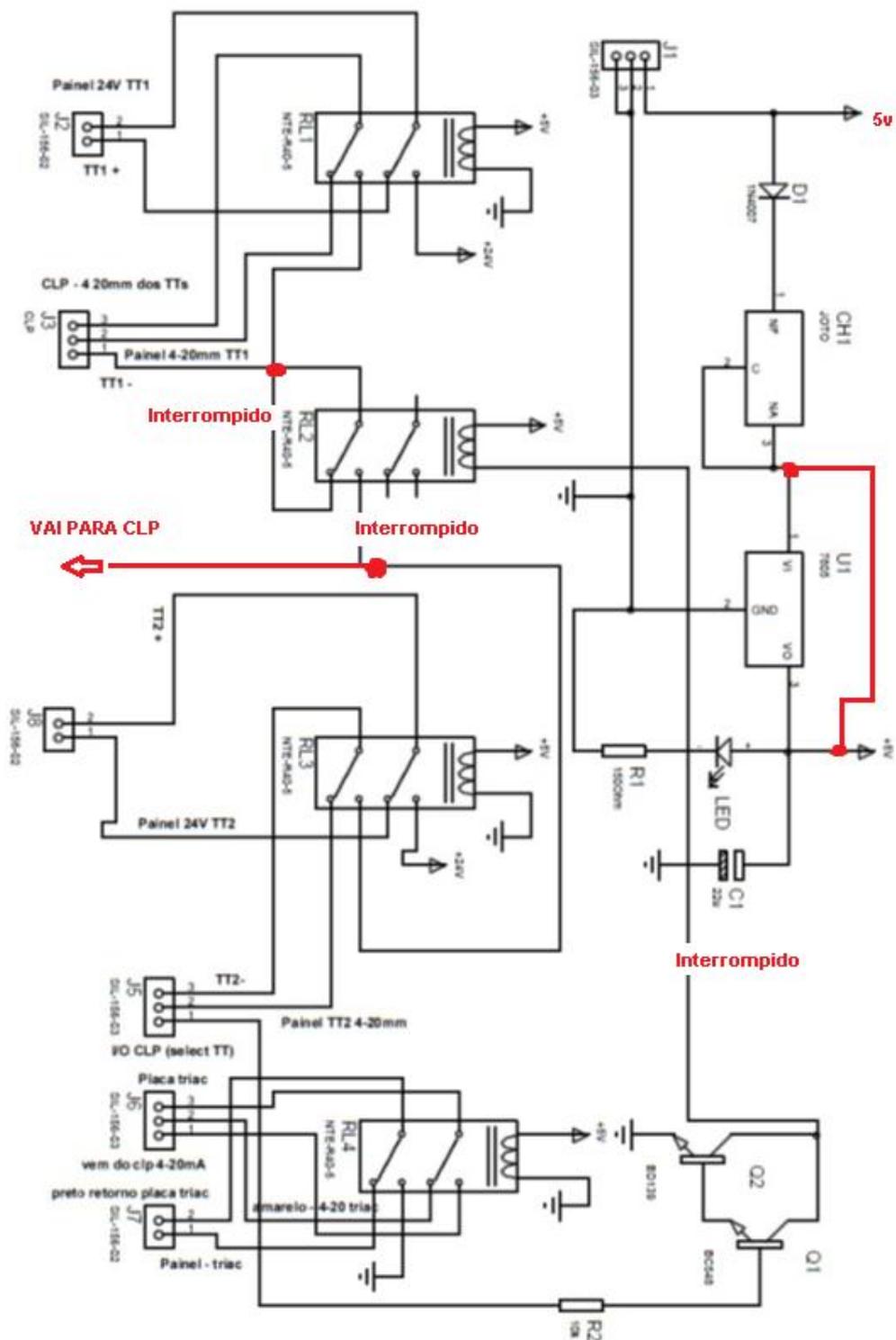
Ziegler, J., G., Nichols, N., B., “Optimum settings for automatic controllers.”, Transactions of the ASME 12, 759. Ano de 1942.



## 2. ESQUEMÁTICO DA PLACA DE POSICIONAMENTO DO DAMPER



## 3. ESQUEMÁTICO DA PLACA DE RELÉS



OBS.: EM VERMELHO ESTÃO AS ALTERAÇÕES REALIZADAS

#### 4. PROGRAMAÇÃO DO CLP

```

(*****
* COPYRIGHT -- Bernecker + Rainer
*****
* Program: Server
* File: Server.st
* Author: Bernecker + Rainer
* Created: February 21, 2008
*****
* Implementation of program Server
*****)
PROGRAM_CYCLIC
DigitalOutput2 := FALSE;

Server.Valor_real4 := AnalogInput1 / 327.67;

float_len := ftoa(Server.Valor_real4 ,ADR(Server.data_buffer_envia));

Server.Valor_real5 := AnalogInput2 / 327.67;

float_len := ftoa(Server.Valor_real5 ,ADR(Server.data_buffer_envia2));

//float_len := ftoa(Server.Valor_real4 ,ADR(Server.data_buffer_envia));

CASE Server.sStep OF

  0: (* Open Ethernet Interface *)
    Server.TcpOpen_0.enable := 1;
    Server.TcpOpen_0.pIfAddr := 0; (* Listen on all TCP/IP Interfaces*)
    Server.TcpOpen_0.port := 1234; (* Port to listen*)
    Server.TcpOpen_0.options := 0;
    Server.TcpOpen_0; (* Call the Function*)

    IF Server.TcpOpen_0.status = 0 THEN (* TcpOpen successfull*)
      Server.sStep := 5;
    ELSIF Server.TcpOpen_0.status = ERR_FUB_BUSY THEN (* TcpOpen not finished -> redo *)

      (* Busy *)
    ELSE (* Goto Error Step *)
      Server.sStep := 100;
    END_IF

  5:

    Server.linger_opt.lLinger := 0; (* linger Time = 0 *)
    Server.linger_opt.lOnOff := 1; (* linger Option ON *)

    Server.TcpIoctl_0.enable := 1;
    Server.TcpIoctl_0.ident := Server.TcpOpen_0.ident; (* Connection Ident from AsTCP.TCP_Open
*)

    Server.TcpIoctl_0.ioctl := tcpSO_LINGER_SET; (* Set Linger Options *)
    Server.TcpIoctl_0.pData := ADR(Server.linger_opt);
    Server.TcpIoctl_0.dataLen := SIZEOF(Server.linger_opt);
    Server.TcpIoctl_0;

    IF Server.TcpIoctl_0.status = 0 THEN (* TcpIoctl successfull *)
      Server.sStep := 10;

```

```

ELSIF Server.TcpIoctl_0.status = ERR_FUB_BUSY THEN (* TcpIoctl not finished -> redo *)

    (* Busy *)
    ELSE (* Goto Error Step *)
        Server.sStep := 100;
    END_IF

10: (* Wait for Client Connection *)
    Server.TcpServer_0.enable := 1;
    Server.TcpServer_0.ident := Server.TcpOpen_0.ident; (* Connection Ident from
AsTCP.TCP_Open *)
    Server.TcpServer_0.backlog := 1; (* Number of clients waiting simultaneously for a connection*)
    Server.TcpServer_0.pIpAddr := ADR(Server.client_address); (* Where to write the client IP-
Address*)
    Server.TcpServer_0; (* Call the Function*)
    Server.Tcp_Estado := 'Esperando por conexão cliente!';
    IF Server.TcpServer_0.status = 0 THEN (* Status = 0 if an client connects to server *)
        Server.sStep := 15;
    ELSIF Server.TcpServer_0.status = ERR_FUB_BUSY THEN (* TcpServer not finished -> redo
*)
        (* Busy *)
        ELSE (* Goto Error Step *)
            Server.sStep := 100;
        END_IF

15:
    Server.TcpIoctl_0.enable := 1;
    Server.TcpIoctl_0.ident := Server.TcpServer_0.identclnt; (* Connection Ident from
AsTCP.TCP_Server *)
    Server.TcpIoctl_0.ioctl := tcpSO_LINGER_SET; (* Set Linger Options *)
    Server.TcpIoctl_0.pData := ADR(Server.linger_opt);
    Server.TcpIoctl_0.datalen := SIZEOF(Server.linger_opt);
    Server.TcpIoctl_0;

    IF Server.TcpIoctl_0.status = 0 THEN (* TcpIoctl successfull *)
        Server.sStep := 20;
    ELSIF Server.TcpIoctl_0.status = ERR_FUB_BUSY THEN (* TcpIoctl not finished -> redo *)
        (* Busy *)
        ELSE (* Goto Error Step *)
            Server.sStep := 100;
        END_IF

20: (* Wait for Data *)
    Server.TcpRecv_0.enable := 1;
    Server.TcpRecv_0.ident := Server.TcpServer_0.identclnt; (* Client Ident from
AsTCP.TCP_Server *)
    Server.TcpRecv_0.pData := ADR(Server.data_buffer1); (* Where to store the incoming data *)
    Server.TcpRecv_0.datamax := SIZEOF(Server.data_buffer1); (* Length of data buffer *)
    Server.TcpRecv_0.flags := 0;
    Server.TcpRecv_0; (* Call the Function*)

    Server.Tcp_Estado := 'Conectado! Esperando receber valor da abertura Damper!';

    IF Server.TcpRecv_0.status = 0 THEN (* Data received *)

```

```

        Server.sStep := 21;
        Server.recv_timeout := 0;
    ELSIF Server.TcpRecv_0.status = tcpERR_NO_DATA THEN (* No data received - wait *)
        Server.recv_timeout := Server.recv_timeout + 1;
    IF Server.recv_timeout > 50 THEN
        Server.sStep := 40;
        Server.recv_timeout := 0;
    END_IF
    ELSIF Server.TcpRecv_0.status = ERR_FUB_BUSY THEN (* TcpRecv not finished -> redo *)

        (* Busy *)
    ELSIF Server.TcpRecv_0.status = tcpERR_NOT_CONNECTED THEN
        Server.sStep := 40;
    ELSE (* Goto Error Step *)
        Server.sStep := 100;
    END_IF

21: (* Wait for Data *)
    Server.TcpRecv_0.enable := 1;
    Server.TcpRecv_0.ident := Server.TcpServer_0.identclnt; (* Client Ident from
AsTCP.TCP_Server *)
    Server.TcpRecv_0.pData := ADR(Server.data_buffer2); (* Where to store the incoming data *)
    Server.TcpRecv_0.datamax := SIZEOF(Server.data_buffer2); (* Lenght of data buffer *)
    Server.TcpRecv_0.flags := 0;
    Server.TcpRecv_0; (* Call the Function*)

    Server.Tcp_Estado := 'Conectado! Esperando receber valor de controle TRIAC!';

    IF Server.TcpRecv_0.status = 0 THEN (* Data received *)
        Server.sStep := 22;
        Server.recv_timeout := 0;
    ELSIF Server.TcpRecv_0.status = tcpERR_NO_DATA THEN (* No data received - wait *)
        Server.recv_timeout := Server.recv_timeout + 1;
    IF Server.recv_timeout > 50 THEN
        Server.sStep := 40;
        Server.recv_timeout := 0;
    END_IF
    ELSIF Server.TcpRecv_0.status = ERR_FUB_BUSY THEN (* TcpRecv not finished -> redo *)

        (* Busy *)
    ELSIF Server.TcpRecv_0.status = tcpERR_NOT_CONNECTED THEN
        Server.sStep := 40;
    ELSE (* Goto Error Step *)
        Server.sStep := 100;
    END_IF

22: (* Wait for Data *)
    Server.TcpRecv_0.enable := 1;
    Server.TcpRecv_0.ident := Server.TcpServer_0.identclnt; (* Client Ident from
AsTCP.TCP_Server *)
    Server.TcpRecv_0.pData := ADR(Server.data_buffer3); (* Where to store the incoming data *)
    Server.TcpRecv_0.datamax := SIZEOF(Server.data_buffer3); (* Lenght of data buffer *)
    Server.TcpRecv_0.flags := 0;
    Server.TcpRecv_0; (* Call the Function*)

    Server.Tcp_Estado := 'Conectado! Esperando receber valor da abertura Válvula!';

```

```

IF Server.TcpRecv_0.status = 0 THEN (* Data received *)
    Server.sStep := 30;
    Server.recv_timeout := 0;
ELSIF Server.TcpRecv_0.status = tcpERR_NO_DATA THEN (* No data received - wait *)
    Server.recv_timeout := Server.recv_timeout + 1;
    IF Server.recv_timeout > 50 THEN
        Server.sStep := 40;
        Server.recv_timeout := 0;
    END_IF
ELSIF Server.TcpRecv_0.status = ERR_FUB_BUSY THEN (* TcpRecv not finished -> redo *)

    (* Busy *)
ELSIF Server.TcpRecv_0.status = tcpERR_NOT_CONNECTED THEN
    Server.sStep := 40;
ELSE (* Goto Error Step *)
    Server.sStep := 100;
END_IF

30: (* Send Data back to Client *)
Server.TcpSend_0.enable := 1;
Server.TcpSend_0.ident := Server.TcpServer_0.identclnt; (* Client Ident from AsTCP.TCP_Server
*)

Server.TcpSend_0.pData := ADR(Server.data_buffer_envia); (* Which data to send *)
Server.TcpSend_0.datalen := SIZEOF(Server.data_buffer_envia); (* Length of data to send *)

//
// Server.TcpSend_0.pData := ADR(Server.data_new); (* Which data to send *)
// Server.TcpSend_0.datalen := SIZEOF(Server.data_new); (* Length of data to send *)
Server.TcpSend_0.flags := 0;
Server.TcpSend_0; (* Call the Function*)

Server.Tcp_Estado := 'Conectado! Enviando valor sensor TT1!';

IF Server.TcpSend_0.status = 0 THEN (* Data sent *)
    Server.sStep := 35;
ELSIF Server.TcpSend_0.status = ERR_FUB_BUSY THEN (* TcpSend not finished -> redo *)

    (* Busy *)
ELSIF (Server.TcpSend_0.status = tcpERR_NOT_CONNECTED) THEN
    Server.sStep := 40;
ELSE (* Goto Error Step *)
    Server.sStep := 100;
END_IF

35: (* Send Data back to Client *)
Server.TcpSend_0.enable := 1;
Server.TcpSend_0.ident := Server.TcpServer_0.identclnt; (* Client Ident from AsTCP.TCP_Server
*)

Server.TcpSend_0.pData := ADR(Server.data_buffer_envia2); (* Which data to send *)
Server.TcpSend_0.datalen := SIZEOF(Server.data_buffer_envia2); (* Length of data to send *)

//
// Server.TcpSend_0.pData := ADR(Server.data_new); (* Which data to send *)
// Server.TcpSend_0.datalen := SIZEOF(Server.data_new); (* Length of data to send *)
Server.TcpSend_0.flags := 0;
Server.TcpSend_0; (* Call the Function*)

```

```
Server.Tcp_Estado := 'Conectado! Enviando valor sensor TT2!';
```

```
IF Server.TcpSend_0.status = 0 THEN (* Data sent *)
```

```
    Server.sStep := 20;
```

```
ELSIF Server.TcpSend_0.status = ERR_FUB_BUSY THEN (* TcpSend not finished -> redo *)
```

```
    (* Busy *)
```

```
ELSIF (Server.TcpSend_0.status = tcpERR_NOT_CONNECTED) THEN
```

```
    Server.sStep := 40;
```

```
ELSE (* Goto Error Step *)
```

```
    Server.sStep := 100;
```

```
END_IF
```

```
40:
```

```
Server.TcpClose_0.enable := 1;
```

```
Server.TcpClose_0.ident := Server.TcpServer_0.identCnt;
```

```
Server.TcpClose_0.how := 0; // tcpSHUT_RD OR tcpSHUT_WR;
```

```
Server.TcpClose_0;
```

```
IF Server.TcpClose_0.status = 0 THEN
```

```
    Server.sStep := 50;
```

```
ELSIF Server.TcpClose_0.status = ERR_FUB_BUSY THEN (* TcpClose not finished -> redo *)
```

```
    (* Busy *)
```

```
ELSE (* Goto Error Step *)
```

```
    Server.sStep := 100;
```

```
END_IF
```

```
50:
```

```
Server.TcpClose_0.enable := 1;
```

```
Server.TcpClose_0.ident := Server.TcpOpen_0.ident;
```

```
Server.TcpClose_0.how := 0; //tcpSHUT_RD OR tcpSHUT_WR;
```

```
Server.TcpClose_0;
```

```
IF Server.TcpClose_0.status = 0 THEN
```

```
    Server.sStep := 0;
```

```
ELSIF Server.TcpClose_0.status = ERR_FUB_BUSY THEN (* TcpClose not finished -> redo *)
```

```
    (* Busy *)
```

```
ELSE (* Goto Error Step *)
```

```
    Server.sStep := 100;
```

```
END_IF
```

```
100: (* Here some error Handling has to be implemented *)
```

```
    Server.Tcp_Estado := 'Erro na conexão! Reiniciar sistema';
```

```
END_CASE
```

```
Server.Valor_real1 := atof(ADR(Server.data_buffer1));
```

```
Server.Valor_real2 := atof(ADR(Server.data_buffer2));
```

```
Server.Valor_real3 := atof(ADR(Server.data_buffer3));
```

```
Server.Valor_real1_1 := Server.Valor_real1 * 327.67;
```

```
Server.Valor_real2_2 := Server.Valor_real2 * 327.67;  
Server.Valor_real3_3 := Server.Valor_real3 * 327.67;  
Server.Valor_int1 := LREAL_TO_INT(Server.Valor_real1);  
Server.Valor_int2 := LREAL_TO_INT(Server.Valor_real2);  
Server.Valor_int3 := LREAL_TO_INT(Server.Valor_real3);  
Server.Valor_int1 := Server.Valor_int1*327 ;  
Server.Valor_int2 := Server.Valor_int2*327;  
Server.Valor_int3 := Server.Valor_int3*327;  
AnalogOutput1 := Server.Valor_int1;  
IF Server.Valor_int1 <> Server.Valor_int_aux THEN  
    DigitalOutput2 := TRUE;  
END_IF  
Server.Valor_int_aux := Server.Valor_int1;  
AnalogOutput2 := Server.Valor_int2;  
AnalogOutput3 := Server.Valor_int3;  
END_PROGRAM
```

## 5. FIRMWARE DO MICROCONTROLADOR ZILOG

```

/*****
*****
* WebLab PUC PR: Acionamento do Damper do forno da planta de temperatura - By Marco Túlio C. Siqueira
* All Rights Reserved - versão 03
* Observações: Nesta versão foi incluído a leitura de setpoint dentro da rotina principal Main (laço While)
* para que a posição do dumper esteja sendo sempre atualizada sem ter que receber a solicitação do
CLP (pulso)
* Foi mantida a rotina de interrupção que verifica o pulso do CLP uma vez que não há nenhum
problema em mantê-la.
*****
*****/

#include <stdio.h>
#include <eZ8.h>
#include <string.h>
#include <MATH.H>

/*****
*****
unsigned char AD2,ajuda,saida,a,b;
unsigned int AD[50],AD1,setpoint,posicao_atual;
float VREF,V,PP;
double posicao_minima,posicao_maxima,MEDIO;
double x,y,z,conta;
double pow(double x, double y);

void temporizador(void)
{
    TOCTL|=0x80; //ATIVA TIMER0
    while(ajuda==1)
    {

    }
    //ajuda=0;
}

/*rotina conversor AD */
void conversor_AD(void)
{
    ADCCTL|=0x80;//Dispara o conversor AD
    while(ADCCTL==0x88)
    {
    }
    AD1=ADCD_H; //CARREGA OS 8 BITS + SIGNIFICATIVOS
    AD1*=0x04;//DESLOCA OS 8 BITS 2 CASAS PARA A ESQUERDA
    AD2=ADCD_L; //CARREGA OS BITS - SIGNIFICATIVOS
    AD2=AD2 >> 6;//DESLOCA OS BITS - SIGNIFICATIVOS 6 CASAS PARA A ESQUERDA
    AD[0]=AD1+AD2; //SOMA DOS BYTES PARA FORMAR 10 BITS
    //V=(AD*VREF); // converte leitura do sensor em Graus C.
    //GRAU=(TEMP*VREF)/0.010; // converte leitura do sensor em Graus C.
    //PP=(V*100)/2;
}

```

```

AD[1]=AD[0];
AD[2]=AD[1];
AD[3]=AD[2];
AD[4]=AD[3];
AD[5]=AD[4];
AD[6]=AD[5];
AD[7]=AD[6];
AD[8]=AD[7];
AD[9]=AD[8];
AD[10]=AD[9];
AD[11]=AD[10];
AD[12]=AD[11];
AD[13]=AD[12];
AD[14]=AD[13];
AD[15]=AD[14];
AD[16]=AD[15];
AD[17]=AD[16];
AD[18]=AD[17];
AD[19]=AD[18];
AD[20]=AD[19];

MEDIO=(AD[1]+AD[2]+AD[3]+AD[4]+AD[5]+AD[6]+AD[7]+AD[8]+AD[9]+AD[10])/10;
setpoint=(MEDIO * posicao_maxima)/1023;
}

/** rotina de verificação de falha nos sensores **
void falha_sensores(void)
{
    unsigned char a;
    a=0;
    a=PCIN;    //lê o portC
    a&=0x0C;   //preserva apenas os ports C2 e C3 (00001100) que são sinais dos 2 fins de curso
    if (a==0x0C)//verifica se os dois sensores estão ativados
    {
        a=0x02;
        PAOUT=a; //seta o portA1 indicando falha em um dos sensores
        //DI;    //desabilita interrupções;
        IRQCTL&=0x7F; // DESABILITA todas as interrupções (A Instrução DI desabilita todas as
interrupções)
        while(1) //trava operação até o reset do microcontrolador
        {
            PBOUT=0;
            break;
        }

        //PBOUT=0;
        //break;
    }
}

} //fim rotina falha_sensores

// Ciclo de inicialização da posição
void homing_cycle(void)
{

```

```

IRQ2ENH=0x00;//Configura Prioridade de interrupção Alta Port C2 e C3 - MSB
IRQ2ENL=0x00;//Configura Prioridade de interrupção Alta Port C2 e C3 - LSB
//unsigned char b;
IRQCTL|=0x80; //HABILITA todas as interrupções
b=0;
conta=5;
falha_sensores(); //chama rotina de verificação de falha sensores
b=PCIN; //lê porC
b&=0x0C; //preserva apenas os ports C2 e C3 (00001100)

if(b==4)//verifica se o PortC2 (00000100) está ativo (sensor direito)
{
    posicao_minima=0; //a configuração inicial é feita no laço Main e o valor inicial setado é
posicao_minima=1;
    posicao_atual=0; //a configuração inicial é feita no laço Main e o valor inicial setado é posicao_atual=1;

//*****
//Avanço da posição zero para ajuste da folga do damper em função da posição do sensor
while(posicao_atual<100)
    {
        conta=conta+1;
        if(conta>4)
            {
                conta=1;
            }
        //saida=2^conta;
        x=2;
        y=conta;
        z=pow(x,y);
        saida=z;
        PBOUT=saida;
        ajuda=1;
        temporizador();
        posicao_atual=posicao_atual+1;
    }

//*****

    posicao_minima=0; //a configuração inicial é feita no laço Main e o valor inicial setado é
posicao_minima=1;
    posicao_atual=0; //a configuração inicial é feita no laço Main e o valor inicial setado é posicao_atual=1;

}

while(posicao_minima!=0) //rotina para busca da posição zero do damper
{
    falha_sensores(); //verifica se os dois sensores estão ativos ao mesmo tempo
    conta=conta-1;
    //conta=conta+1; // opcional para inverter o sentido de rotação do motor
    if(conta<1)
    //if (conta>4)// opcional para inverter o sentido de rotação do motor
    {
        conta=4;
        //conta=1; // opcional para inverter o sentido de rotação do motor
    }
    x=2;

```

```

y=conta;
z=pow(x,y);
saida=z;
PBOUT=saida;
ajuda=1;//flag de auxilio para o temporizador (enquanto 1 temporiza)
temporizador();
b=PCIN; //lê porC
b&=0x0C; //preserva apenas os ports C2 e C3 (00001100)
if(b==0x04) //verifica se o PortC2 (00000100) está ativo (sensor direito)
{
    posicao_minima=0;
    posicao_atual=0;
    //conta=0;

    /*******
    //Avanço da posição zero para ajuste da folga do damper em função da posição do sensor
    while(posicao_atual<100)
    {
        conta=conta+1;
        if(conta>4)
        {
            conta=1;
        }
        //saida=2^conta;
        x=2;
        y=conta;
        z=pow(x,y);
        saida=z;
        PBOUT=saida;
        ajuda=1;
        temporizador();
        posicao_atual=posicao_atual+1;
    }

    /*******
**

    posicao_minima=0;
    posicao_atual=0;

    }
}

b=PCIN;
b&=0x08;
while(b!=0x08) //rotina para encontrar a posição maxima de abertura do damper através detectado pelo
PortC3 (fim de curso)
{
    falha_sensores();
    conta=conta+1;
    if(conta>4)
    {
        conta=1;
    }
    //saida=2^conta;
    x=2;

```

```

        y=conta;
        z=pow(x,y);
        saida=z;
        PBOUT=saida;
        ajuda=1;
        temporizador();
        posicao_atual=posicao_atual+1;
        b=PCIN;    //lê porC (fim de curso)
        b&=0x08;   //preserva apenas o portC3 (00001000)
    }
    posicao_maxima=posicao_atual-2;
    //posicao_maxima=posicao_atual-5;
    //conversor_AD();    //faz leitura da posição de referência desejada.
    setpoint=posicao_minima; //seta posição de inicio do damper (fechado)
    while(IRQ2!=0)
    {
        IRQ2=0;
    }
    IRQ2ENH=0x00; //Configura Prioridade de interrupção Alta Port C2 e C3 - MSB
    IRQ2ENL=0x03; //Configura Prioridade de interrupção Alta Port C2 e C3 - LSB
} //fim rotina homing_cycle

/*Rotina de Interrupção do PortC0 - INTERRUPÇÃO CLP SOLICITANDO POSICIONAMENTO DUMPER */
void interrupt int0_handler(void)
{
    SET_VECTOR(C0,int0_handler);

    conversor_AD();

} //fim rotina de interrupção do PortC0

/*Rotina de Interrupção do PortC1 - INICIAR CICLO DE HOMING */
void interrupt int1_handler(void)
{
    SET_VECTOR(C1,int1_handler);

    //IRQ2ENH=0x01;    //Configura Prioridade de interrupção Alta Port C2 e C3 - MSB
    //IRQ2ENL=0x01;    //Configura Prioridade de interrupção Alta Port C2 e C3 - LSB
    //ajuda=1;
    //temporizador();

    //IRQ2ENH=0x03;    //Configura Prioridade de interrupção Alta Port C2 e C3 - MSB
    //IRQ2ENL=0x03;    //Configura Prioridade de interrupção Alta Port C2 e C3 - LSB
    IRQCTL&=0x7F;    // DESABILITA todas as interrupções (A Instrução DI desabilita todas as
interrupções)
    posicao_minima=1; //seta posição mínima com valor diferente de zero para reiniciar ciclo
    posicao_atual=1;
    ajuda=1;
    while(IRQ2!=0x00)
    {
        IRQ2=0;
    }
    homing_cycle();
}

```

```

//IRQ2ENH=0x03; //Configura Prioridade de interrupção Alta Port C2 e C3 - MSB
//IRQ2ENL=0x03; //Configura Prioridade de interrupção Alta Port C2 e C3 - LSB

} //fim rotina de interrupção do PortC1

/*Rotina de Interrupção do PortC2 - CHAVE FIM DE CURSO DIREITA
void interrupt int2_handler(void)
{
    SET_VECTOR(C2,int2_handler);
    posicao_minima=0;

} //fim rotina de interrupção do PortC2 */

/*Rotina de Interrupção do PortC3 - CHAVE FIM DE CURSO ESQUERDA
void interrupt int3_handler(void)
{
    SET_VECTOR(C3,int3_handler);

    posicao_maxima=posicao_atual;
//  IRQCTL&=0x7F; // DESABILITA todas as interrupções
//  IRQCTL&=0x80; // HABILITA todas as interrupções

} //fim rotina de interrupção do PortC3 */

//Rotina de Interrupção do Timer0
void interrupt int4_handler(void)
{
    SET_VECTOR(TIMER0,int4_handler);

    ajuda=0; //sinaliza fim temporização

} //fim rotina de interrupção do Timer0

//*****
//*****
void main (void)
{
    char bb;
    //Configurando o PortB como função alternativa
    PBADDR=0x01; //Configura PortB como função alternativa (entrada analógica)
    PBCTL=0x01; //Seleciona o PortB0 como função alternativa (conversor AD)
    PBADDR=0x02; //Configura PortB como função alternativa (entrada analógica)
    PBCTL=0x01; //Seleciona o PortB0 como função alternativa (conversor AD)

    //Configurando o PortA como saída
    PAADDR=0x01; //Configurando PortA DATA DIRECTION
    PACTL=0x00; //Configurando PortA1 como saída (Pinos de dados LCD)

    //configurando o ADC
    ADCCTL=0x00; //Configura o AD para trabalhar com o PortB0, referência interna e single shot

```

Principal

```

//ADCCTL|=0x20; //Configura o AD para trabalhar com Ref. externa.

// Configurando o PortC
PCADDR=0x01; //Configurando PORTC (DATA Direction)
PCCTL=0x0F; //Configurando Port's como entrada: PC0,PC1,PC2,PC3

//Configurando as interrupções dos PORT's
IRQCTL&=0x7F; // DESABILITA todas as interrupções (A Instrução DI desabilita todas as
interrupções)

//Habilitando E Desabilitando todas as interrupções que foram configuradas
//IRQCTL|=0x80; // Habilita todas as interrupções (A Instrução EI habilita todas as interrupções)

//Configurando o Timer0 e sua Interrupção (Rotina de amostragem do sensor)
T0CTL=0x00; //DESABILITA TIMER0 e seta Timer0 para operar no modo ONE SHOT
//T0CTL|=0x01; //MODO DE OPERAÇÃO: CONTINUO
//T0CTL|=0x38; //PRESCALE em 128
//T0CTL|=0x39; //PRESCALE em 128
//T0CTL|=0x10; //PRESCALE em 128
//T0CTL|=0x09; //PRESCALE em 128
T0CTL|=0x08; //PRESCALE em 128
T0H=0x00; //VALOR INICIAL DE CONTAGEM MSB
T0L=0x01; //VALOR INICIAL DE CONTAGEM LSB
T0RH=0xFF; //VALOR FINAL DE CONTAGEM (VALOR DE RECARGA MSB)
T0RL=0xFF; //VALOR FINAL DE CONTAGEM (VALOR DE RECARGA LSB)
IRQ0ENH=0x00; //CONFIG.INTERRUPÇÃO DO "TIMER0" EM PRIORIDADE baixa- MSB
IRQ0ENL=0x20; //CONFIG.INTERRUPÇÃO DO "TIMER0" EM PRIORIDADE baixa- LSB
//IRQCTL&=0x7F; // DESABILITA todas as interrupções (A Instrução DI desabilita todas as
interrupções)
//IRQCTL|=0x80; //HABILITA todas as interrupções
//T0CTL|=0x80; //ATIVA TIMER0

posicao_atual=1; //seta configurações iniciais para iniciar o ciclo de homing pela primeira vez
posicao_minima=1; //
//configurações gerais
//VREF=2.1933/1023; //Constante de conversão da temperatura em GRAUS C
//utilizando referência externa
VREF=2.0/1023; //Constante de conversão da temperatura em GRAUS C

IRQ2ENH=0x00; //Configura Prioridade de interrupção Alta Port C2 e C3 - MSB
IRQ2ENL=0x03; //Configura Prioridade de interrupção Alta Port C2 e C3 - LSB
//IRQCTL|=0x80; //HABILITA todas as interrupções
homing_cycle(); //chama rotina de homing (verificar o curso do damper)
//IRQCTL|=0x80; // Habilita todas as interrupções (A Instrução EI habilita todas as interrupções)

//conversor_AD();
//setpoint=posicao_minima;
//posicao_atual=0;
//setpoint=posicao_maxima;
//setpoint=0;
while(1) // Infinite while loop

```

```

{
    conversor_AD(); //Chama rotina de leitura do Set Point de posição

    falha_sensores();
    if ((posicao_atual==setpoint)|(posicao_atual==(setpoint-1))|(posicao_atual==(setpoint+1)))
    //if (posicao_atual==setpoint)
    {
        PBOUT=0;
        falha_sensores();
        //conversor_AD(); //Chama rotina de leitura do Set Point de posição
    }

    if(posicao_atual<(setpoint-2))
        //if(posicao_atual<setpoint)
    {
        bb=PCIN; //lê estado do sensor fim de curso
        bb&=0x08;
        if(bb==0x08)
        {
            conta=conta-1;
        }

        conta=conta+1;
        if(conta>4)
        {
            conta=1;
        }

        //saida=2^conta;
        x=2;
        y=conta;
        z=pow(x,y);
        saida=z;

        PBOUT=saida;
        posicao_atual=posicao_atual+1;
        temporizador();
        ajuda=1;

        /*
        if(posicao_atual>posicao_maxima)
        {

            PBOUT=0;
            a=0x02;
            PAOUT=a; //seta o portA1 indicando falha em um dos sensores
            //DI; //desabilita interrupções;
            IRQCTL&=0x7F; // DESABILITA todas as interrupções (A Instrução DI desabilita todas
as interrupções)
            break; //trava operação até o reset do microcontrolador
            //while(1) //trava operação até o reset do microcontrolador
            //{
            //}

        }
        */

```

```

}
if(posicao_atual>(setpoint+1))
{
    bb=PCIN; //lê porC
    bb&=0x0C; //preserva apenas os ports C2 e C3 (00001100)

    if(bb==4) //verifica se o PortC2 (00000100) está ativo (sensor direito)
    {
        conta=conta+1;
    }

    conta=conta-1;
    if(conta<1)
    {
        conta=4;
    }

    //saida=2^conta;
    x=2;
    y=conta;
    z=pow(x,y);
    saida=z;

    PBOUT=saida;
    posicao_atual=posicao_atual-1;
    ajuda=1;
    temporizador();

    /*
    if(posicao_atual<posicao_minima)
    {
        PBOUT=0;
        a=0x02;
        PAOUT=a; //seta o portA1 indicando falha em um dos sensores
        //DI; //desabilita interrupções;
        IRQCTL&=0x7F; // DESABILITA todas as interrupções (A Instrução DI desabilita todas
as interrupções)
        break; //trava operação até o reset do microcontrolador
        //while(1) //trava operação até o reset do microcontrolador
        //{
        //}
    }
    */
}
}

```

## 6. FLUXOGRAMAS

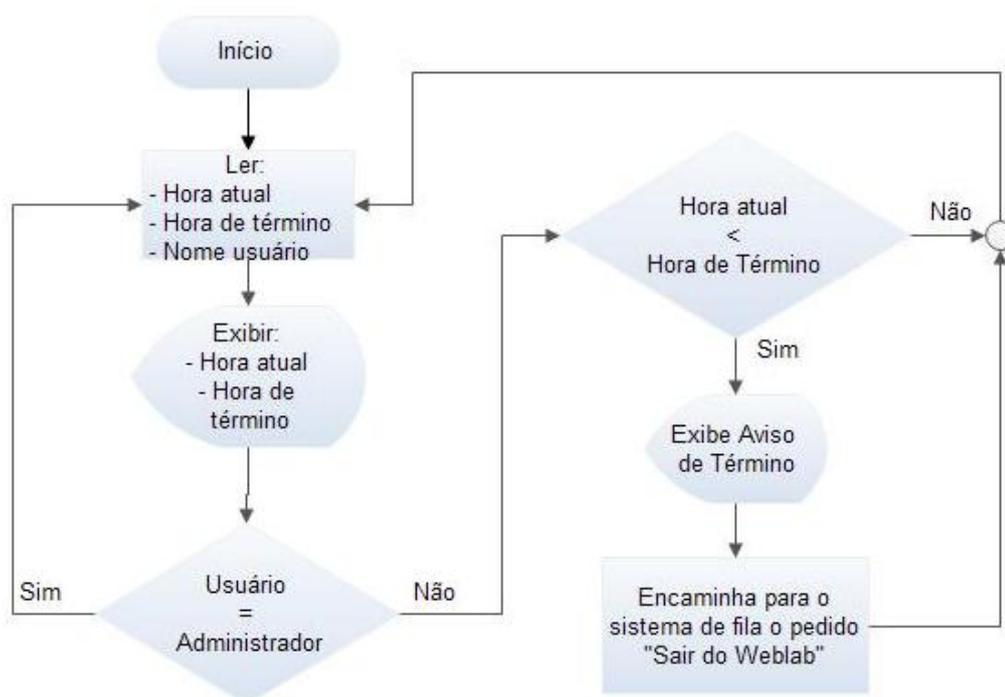
- *Temporização de um minuto*: quando o usuário iniciar a gravação dos dados esta rotina é habilitada para garantir que as informações coletadas sejam gravados a cada intervalo um minuto.

### ROTINA DE TEMPORIZAÇÃO DE UM MINUTO



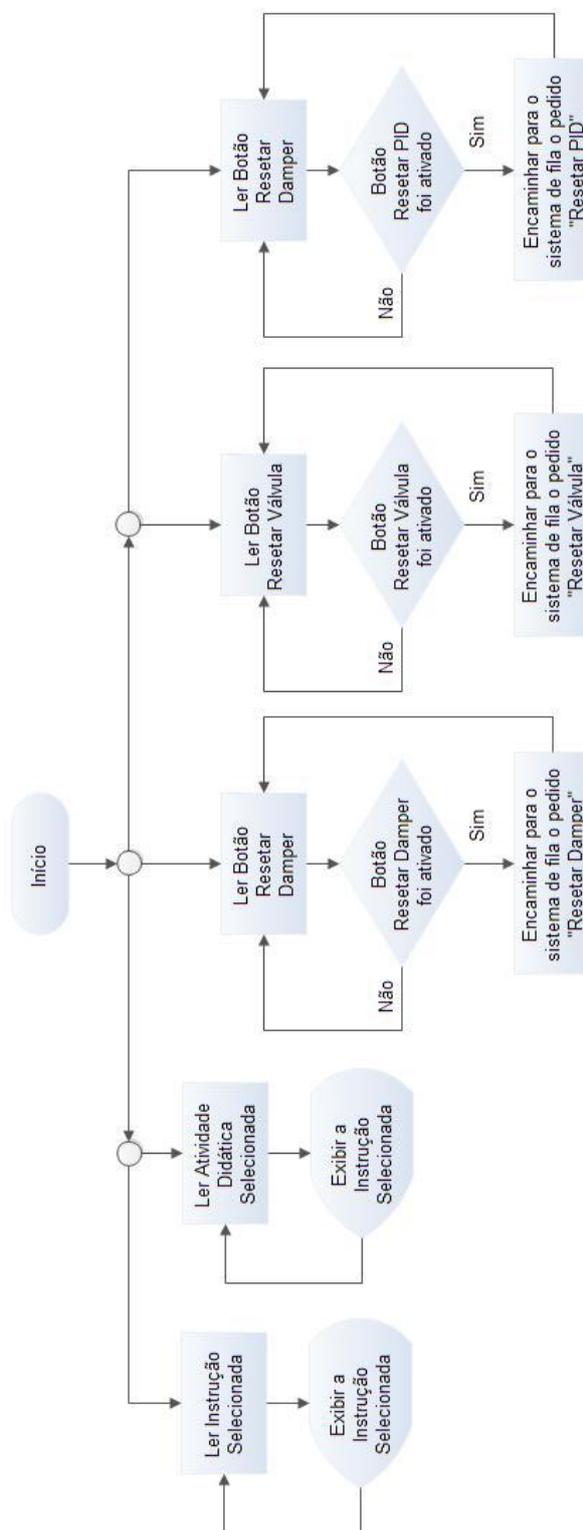
- *Verificação do tempo de acesso ao experimento*: controla o tempo de acesso permitido para realizar o experimento, segundo os dados do cadastro.

### ROTINA DE VERIFICAÇÃO DO TEMPO DE ACESSO AO EXPERIMENTO



- Botões de reset da Válvula de ar, da Abertura do Damper e dos Parâmetros PID: esta rotina verifica se algum dos botões de reset foi acionado.

### ACIONAMENTO DOS BOTÕES DE RESET

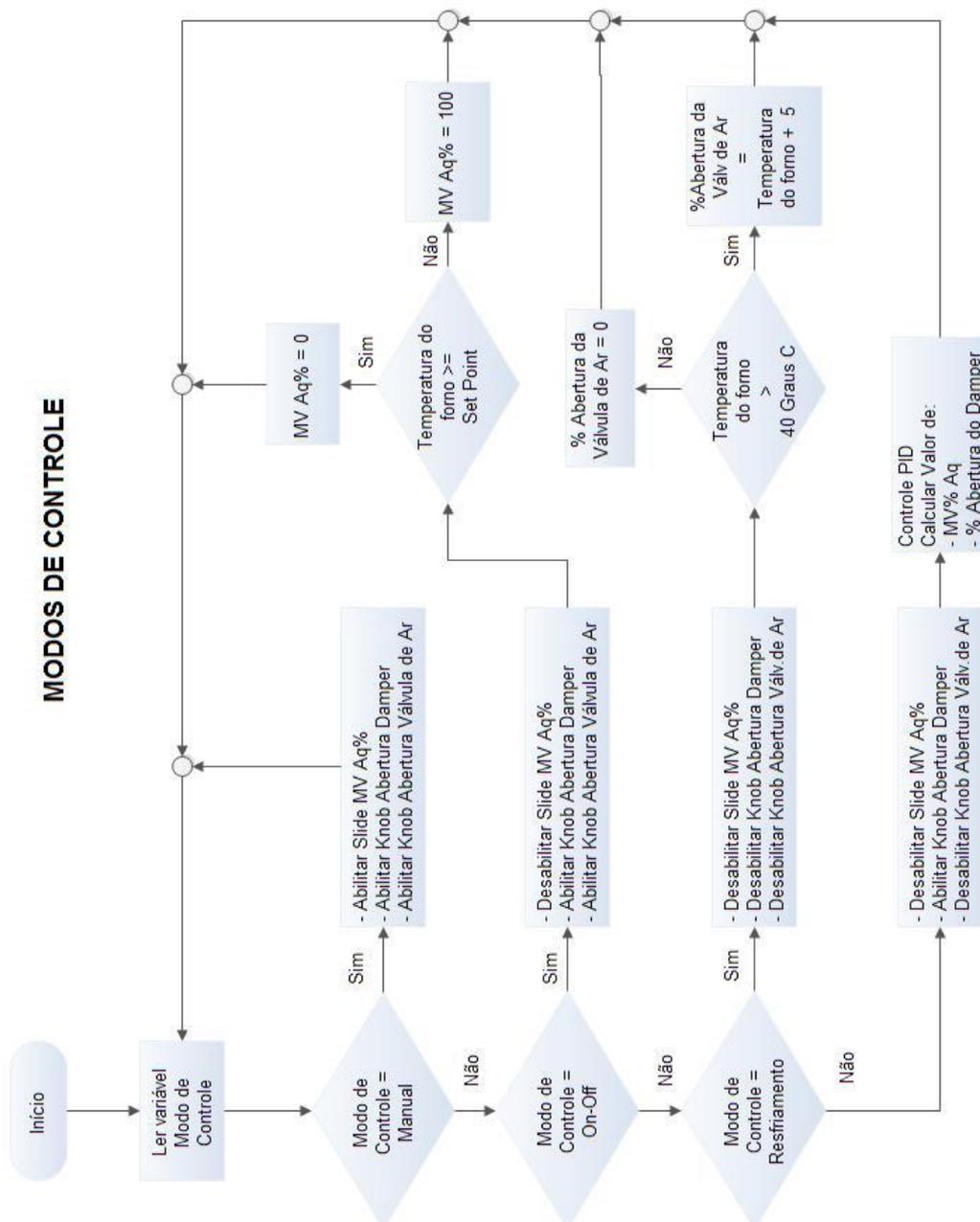


- *Indicação visual do estado do sistema*: esta rotina é responsável por gerar uma indicação visual das ações tomadas pelo sistema, tais como “Gravando, Criando novo arquivo..., etc.”

### INDICAÇÃO DO ESTADO DO SISTEMA

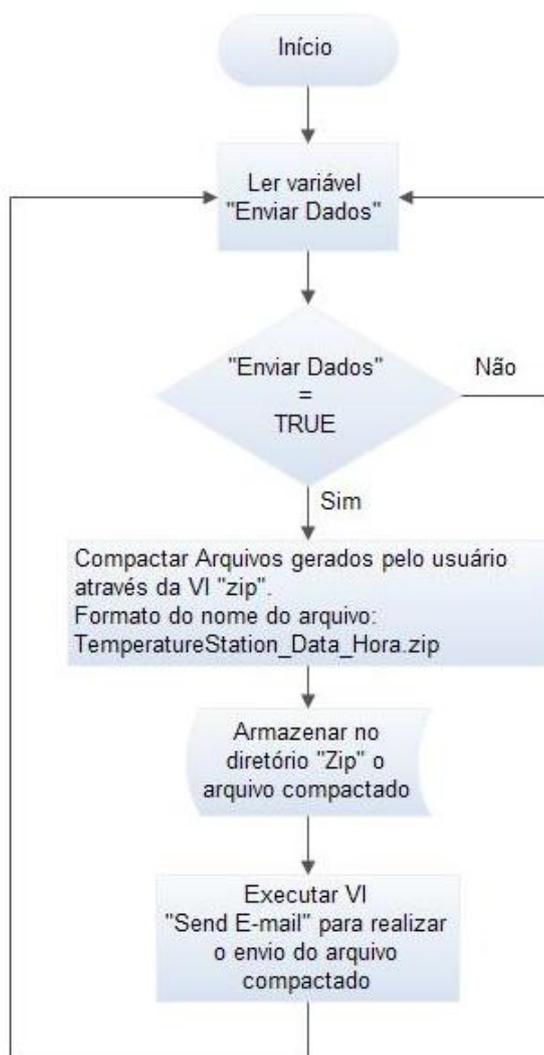


- *Modos de controle do processo de temperatura:* nesta rotina é executado o tipo de controle selecionado pelo usuário ou definido pelo sistema (Manual, On-Off, PID, Resfriamento ou PID com resfriamento).



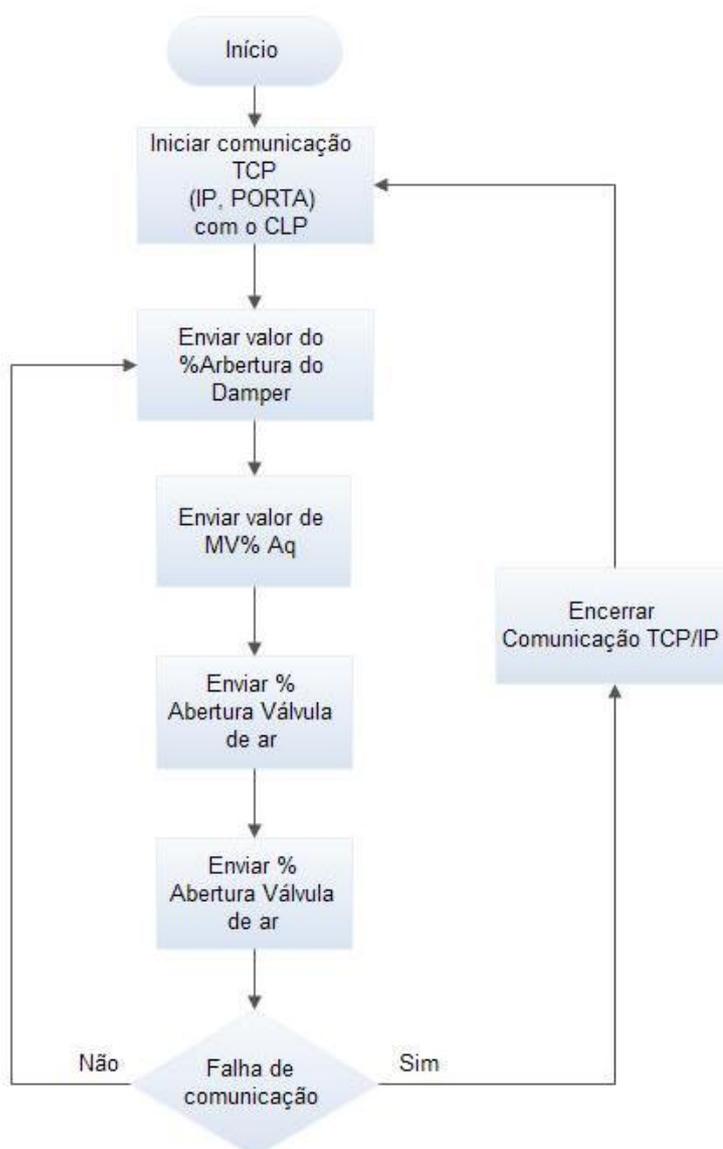
- *Envio de e-mail*: esta rotina permite que o usuário envie todos os dados coletados para seu endereço de e-mail.

## ENVIO DE E-MAIL

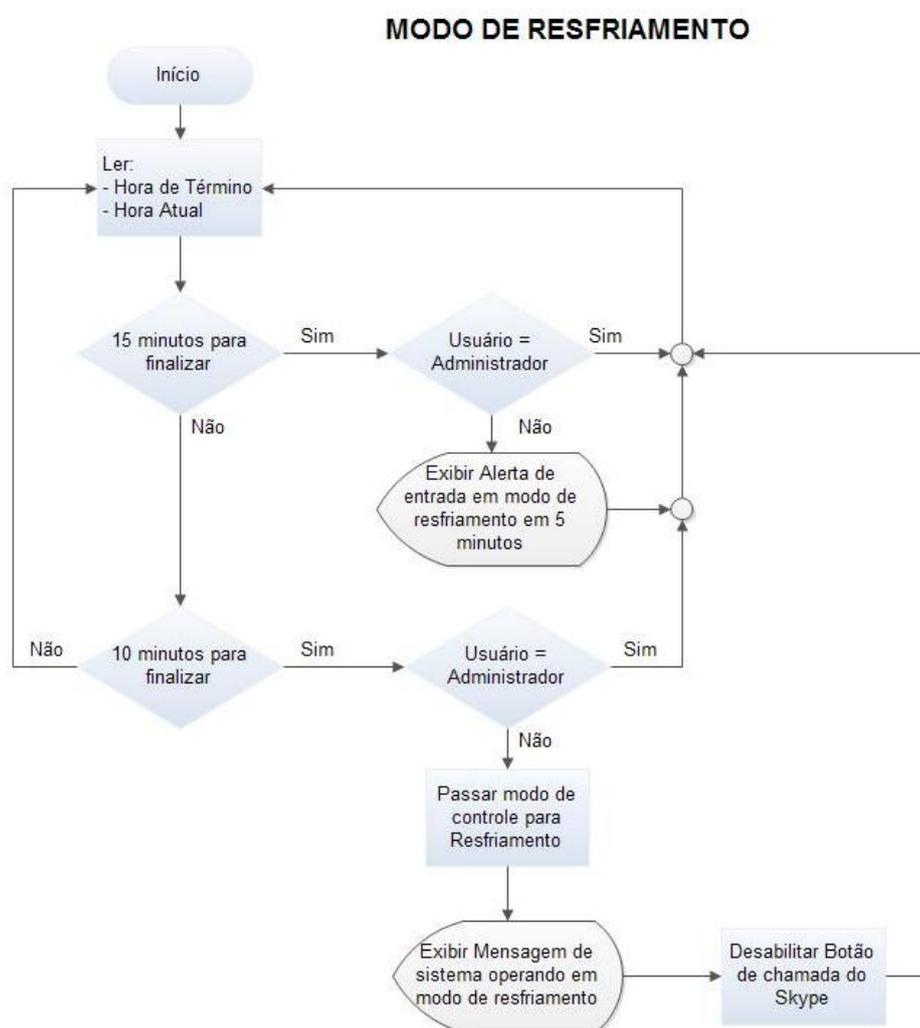


- *Comunicação TCP/IP*: esta rotina realiza a troca de dados entre o CLP e o software LabVIEW™ através do protocolo de comunicação TCP/IP.

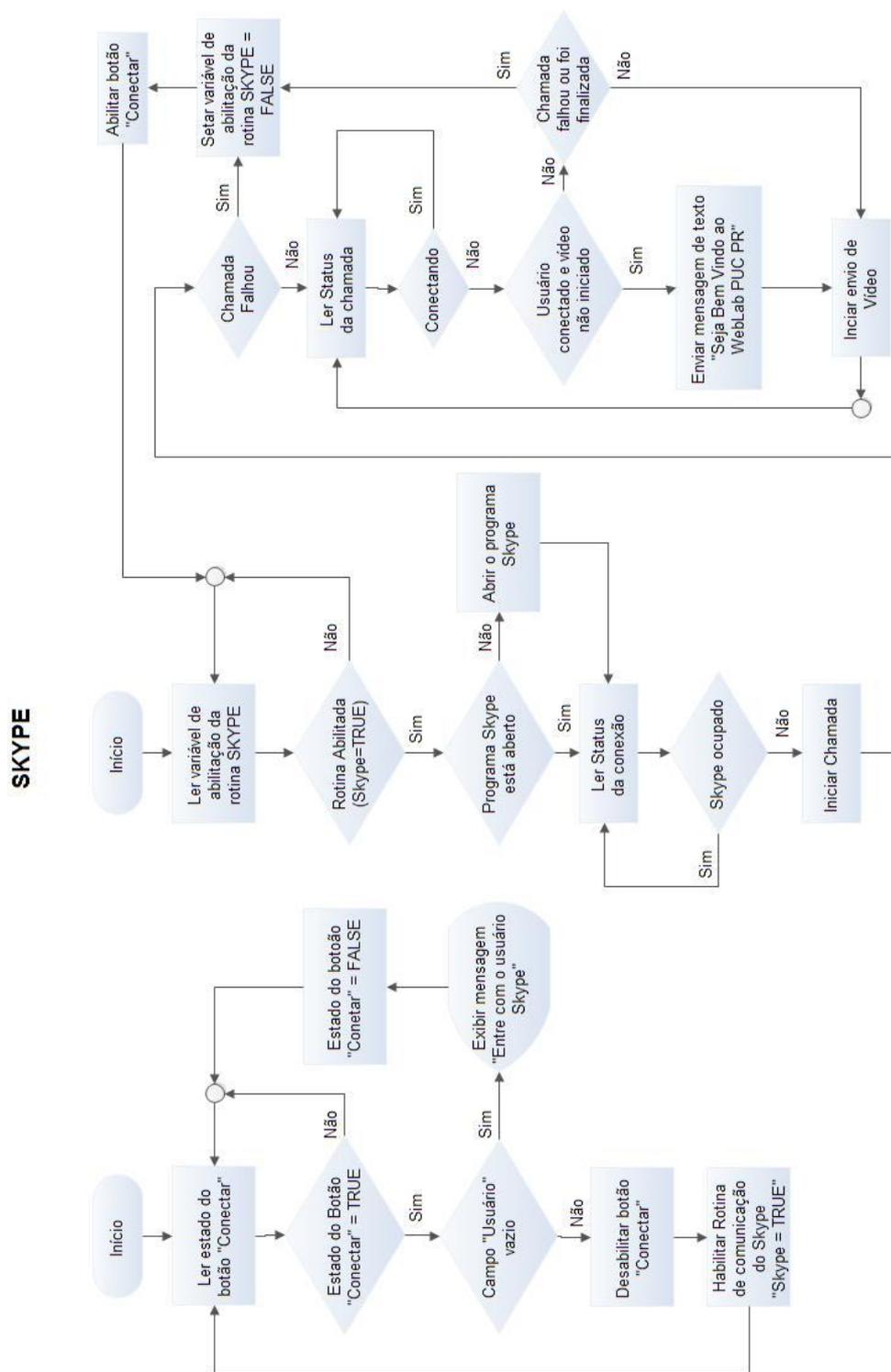
### COMUNICAÇÃO TCP/IP



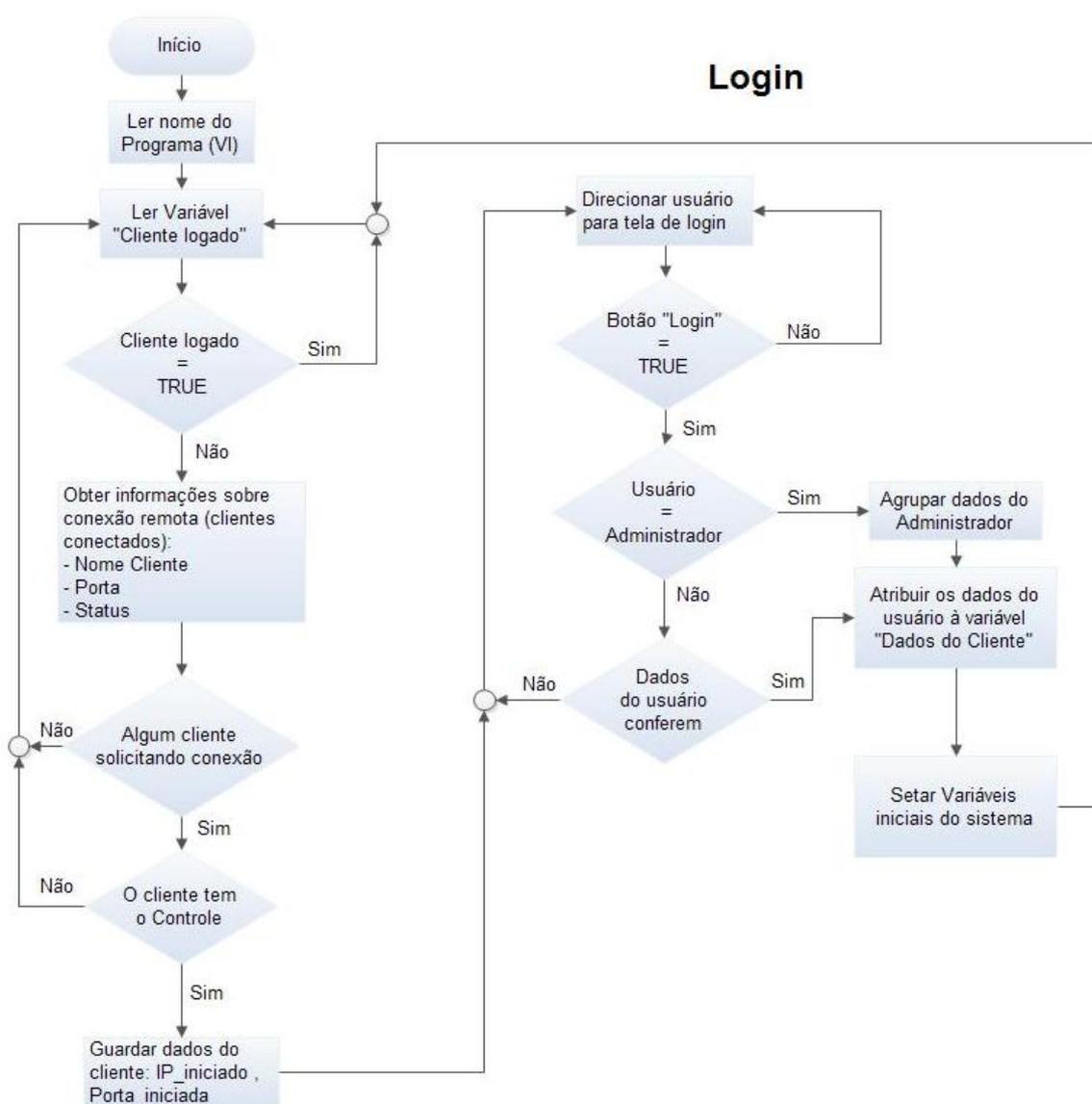
- *Verificação de entrada em modo de resfriamento*: esta rotina monitora o tempo de execução do experimento e aciona o modo de operação “Resfriamento” quando faltarem dez minutos para o encerramento do tempo alocado para o usuário. Tem como objetivo restabelecer as condições iniciais de operação para o próximo usuário.



- *Skype*: esta rotina irá configurar o programa Skype™ para que o mesmo realize uma vídeo chamada para o endereço do usuário Skype, fornecido durante o processo de cadastramento ou após a confirmação do login.

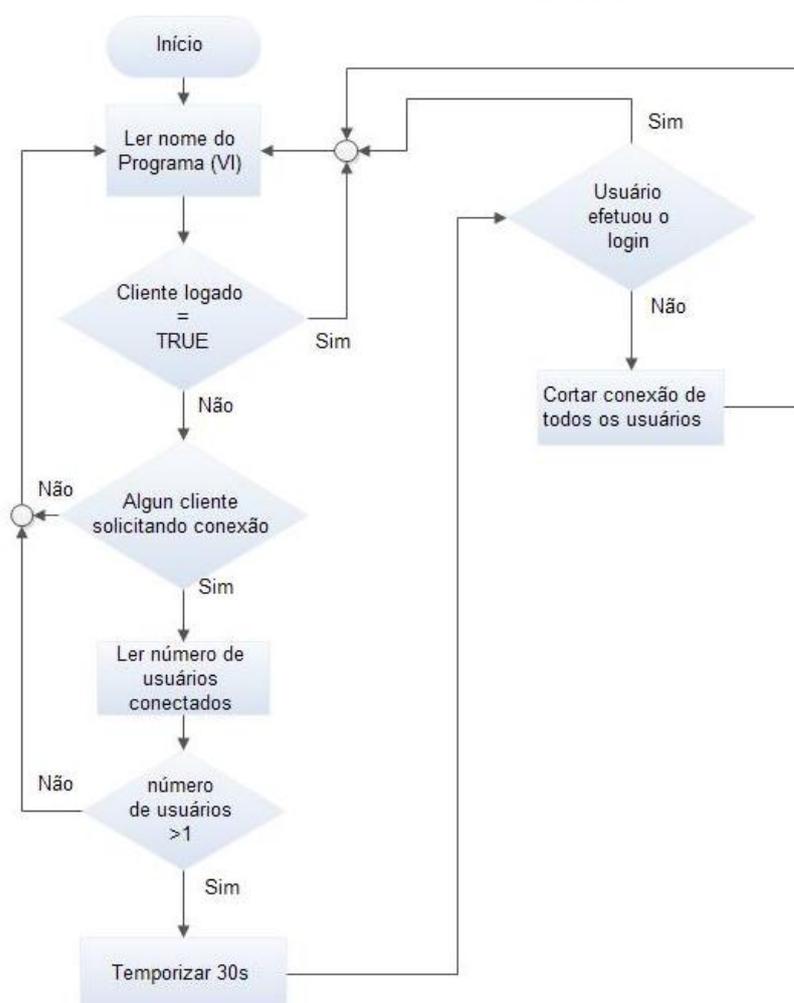


- *Login*: esta rotina é executada sempre que o programa detectar a presença de um novo usuário solicitando conexão. Neste caso o programa irá confrontar os dados de entrada com os dados armazenados no arquivo de login, gerado pelo programa de cadastro. Após a confirmação do login, os dados do usuário serão encaminhados ao programa principal para que este possa fazer uso dos mesmos, bem como gerenciar o tempo de acesso permitido para a execução do experimento.



- *Monitoração dos usuários conectados ao WebLab*: esta rotina verifica se há mais que um usuário tentando se logar ao experimento. Em caso afirmativo o sistema irá iniciar uma contagem regressiva de trinta segundos e após este tempo, se não ocorrer o login, o sistema irá derrubar (cortar a conexão) de todos os usuários a ele conectados. Se um usuário, que já está logado, perder a conexão com o sistema, não será necessário que o mesmo refaça o processo de login, a menos que exista outro usuário solicitando conexão.

### Monitoração de Usuários



- *Avaliação do sistema*: esta rotina foi incluída para permitir que o usuário faça uma avaliação geral do sistema através de um questionário que inclui um total de nove perguntas.

