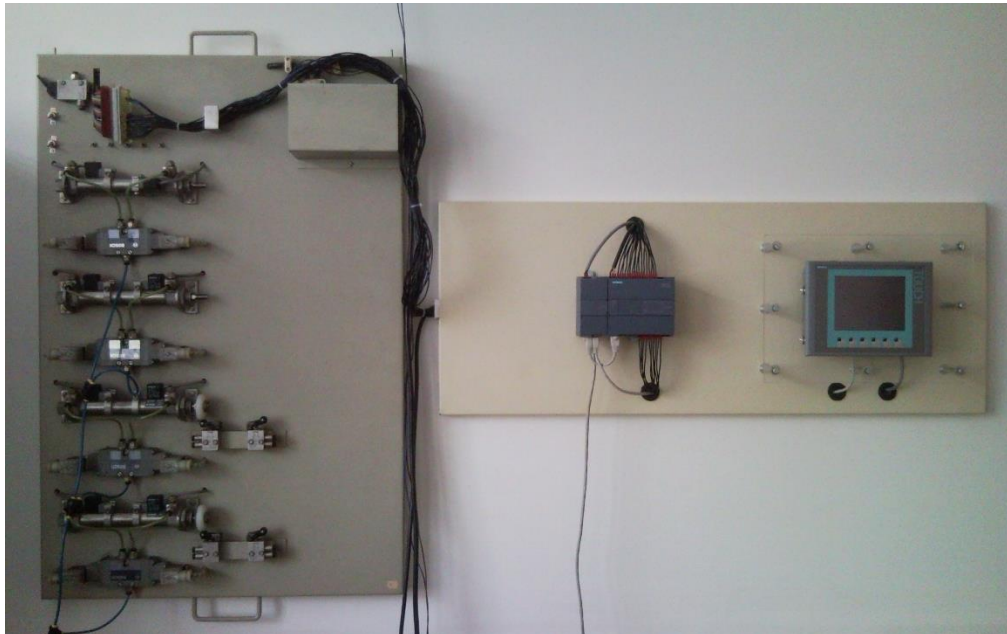




ISEL

**INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA**

**Área Departamental de Engenharia Mecânica**



## **Projecto dum Laboratório Remoto para Automação de Processos Industriais**

**LUIS MIGUEL SILVA MARTINS**  
(Licenciado em Engenharia Mecânica)

Trabalho Final de Mestrado para obtenção do grau de Mestre  
em Engenharia Mecânica

Orientador: Doutor Mário José Gonçalves Cavaco Mendes

Júri:

Presidente: Doutor João Carlos Quaresma Dias

Vogais: Doutor Luís Filipe Figueira Brito Palma

Doutor Mário José Gonçalves Cavaco Mendes

**Dezembro 2013**





**INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA**

**Área Departamental de Engenharia Mecânica**

## **Projecto dum Laboratório Remoto para Automação de Processos Industriais**

**LUIS MIGUEL SILVA MARTINS**  
(Licenciado em Engenharia Mecânica)

Trabalho Final de Mestrado para obtenção do grau de Mestre  
em Engenharia Mecânica

Orientador: Doutor Mário José Gonçalves Cavaco Mendes

Júri:

Presidente: Doutor João Carlos Quaresma Dias

Vogais: Doutor Luís Filipe Figueira Brito Palma

Doutor Mário José Gonçalves Cavaco Mendes

**Dezembro 2013**



## **Agradecimentos**

Gostaria de começar por agradecer ao meu orientador Prof. Doutor Mário J. G. C. Mendes por todo o apoio, motivação, ajuda e disponibilidade demonstradas durante a realização deste trabalho.

Agradecimento ao Prof. Doutor Luis Palma da Universidade Nova de Lisboa e ao Prof. Doutor Caldas Pinto do Instituto Superior Técnico por me terem recebido bem, a mim e ao meu orientador, nas suas respetivas instituições e terem possibilitado o contacto com os seus laboratórios remotos.

Queria também agradecer a colaboração do Prof. Doutor Pedro Silva e da empresa Sandometal pelo material gentilmente cedido.

Agradecimento à Siemens pelo material gentilmente cedido à ADEM/ISEL, o qual me possibilitou fazer o trabalho de Mestrado.

Agradecimento pela motivação, apoio e ajuda demonstrados pelos meus colegas Frederico Caldas e Nuno Ribeiro em todo o trabalho elaborado.

Por fim o mais profundo agradecimento aos meus pais, e à minha namorada pelo incentivo incondicional e pelo apoio financeiro que me proporcionaram, uma vez que sem eles nada disto seria possível.



## Resumo

Nos dias de hoje, com a evolução das tecnologias de informação e consequente modernização da automação industrial, e com uma perspectiva de acompanhar essa tendência por parte do ensino da engenharia, houve a necessidade de implementar um laboratório remoto na Área Departamental de Engenharia Mecânica (ADEM) do Instituto Superior de Engenharia de Lisboa (ISEL), nomeadamente para o ensino da Automação Industrial.

O tema do laboratório remoto surge da necessidade de melhorar e reciclar as atuais condições tecnológicas do Laboratório de Automação Industrial da ADEM/ISEL e com a oferta do autómato programável S7-1200 pela empresa SIEMENS, com capacidade de comunicação PROFINET, foi possível concretizá-lo.

O objetivo deste trabalho foi então desenvolver e implementar um laboratório remoto, que fosse simples e eficaz, para o ensino prático à distância de matérias lecionadas no âmbito da unidade curricular “Automação de Processos Industriais”, nomeadamente os ciclos electropneumáticos. Simples do ponto de vista do utilizador, que permitisse aceder a todas as suas funcionalidades através de *web browser* sem ter que instalar qualquer outro programa. Eficaz enquanto laboratório remoto, ou seja, ter todas as funcionalidades que o aluno dispõe no laboratório físico.

Com este objetivo em mente, foi feito um levantamento do material existente no laboratório em causa, para perceber o que era possível fazer com o mínimo de custos. Implementou-se uma arquitetura simples, com auxílio de vários programas gratuitos, tecnologias de programação como *JavaScript* e *PHP*, bases de dados, tudo de forma a poder ter acesso remoto a todas as funcionalidades do laboratório. E por fim foi implementada uma interface agradável e simples para os alunos poderem aprender, trabalhar e usufruir de todas as potencialidades do laboratório remoto. O laboratório foi implementado e testado pelos alunos de Automação.

## Palavras-chave

Laboratório Remoto, Ensino à distância, Automação Industrial, Controladores Lógicos Programáveis (PLC), Sistema electropneumático



## **Abstract**

Nowadays, with the development of information technologies and the consequent industrial automation modernization, as well as the perspective to follow this tendency in engineering education, there was a need to implement a remote laboratory at Mechanical Engineering Department (ADEM), Instituto Superior de Engenharia de Lisboa (ISEL), particularly in the teaching of industrial automation.

The remote laboratory subject arises from the need to improve and recycle the current conditions of the industrial automation laboratory at ADEM/ISEL and this became possible to materialize with the offer of the S7-1200 PLC by SIEMENS, which has PROFINET communication capabilities.

The main objective of this work was therefore the remote laboratory development and implementation, which was simple and effective for the practical teaching (at distance) of the subjects taught within the course (Process Industrial Automation), namely the electro-pneumatic cycles. Simple from the user point of view, to allow the user to access all its functionalities using a web browser without the need of installing any other software. Effective as a remote laboratory, it means to have all the functionalities that a student can find in the physic laboratory.

With this goal in mind, what was done was an inventory of the existing laboratory equipment to understand what was possible to do with minimal costs. It has been implemented a simple architecture, with the help of freeware software, programing technologies like Javascript and PHP, databases, to be possible to have a remote access to all the laboratory functionalities. To end, it has been implemented a simple user friendly interface to allow the students to use all the capabilities of the remote laboratory. The laboratory has been implemented and tested by the automation students.

## **Keywords**

Remote Laboratory, Learning in distance, Industrial Automation, Programmable Logic Controller (PLC), Electro-Pneumatic System



# Índice

AGRADECIMENTOS .....	I
RESUMO .....	III
ABSTRACT.....	V
LISTA DE FIGURAS.....	XI
LISTA DE TABELAS .....	XIII
LISTA DE ABREVIATURAS.....	XV
1 INTRODUÇÃO.....	1
1.1 Motivação.....	1
1.2 Objetivos .....	3
1.3 Contribuições .....	4
1.4 Estrutura do Trabalho.....	5
2 ESTADO DA ARTE.....	7
2.1 Automação Industrial .....	7
2.2 Controladores Lógicos Programáveis .....	8
2.2.1 Linguagens de Programação.....	12
2.3 Redes Industriais .....	14
2.4 Tipos de Laboratórios .....	16
2.4.1 Laboratório Físico.....	18
2.4.2 Laboratório Remoto.....	19
2.4.3 Laboratório Virtual .....	20
2.5 Exemplos de Laboratórios Remotos .....	21
2.5.1 Nacionais .....	22
2.5.2 Internacionais.....	24
3 TECNOLOGIAS IMPLEMENTADAS E MONTAGEM.....	27
3.1 Revisão de Conhecimentos .....	27
3.1.1 Protocolo HTTP.....	27

3.1.2	PROFINET .....	28
3.1.3	VPN .....	31
3.1.4	Controlador lógico programável Siemens S7-1200.....	32
3.1.5	Programas Utilizados.....	34
3.1.6	Linguagens de Programação Utilizadas.....	42
3.2	Tecnologias Utilizadas .....	47
4	LABORATÓRIO REMOTO DO ISEL .....	49
4.1	Tipo de Arquitetura Implementada .....	49
4.1.1	Utilizadores.....	51
4.1.2	Gestão da Base de Dados.....	52
4.1.3	Ambiente de Trabalho Remoto.....	52
4.1.4	Web Server Siemens S7-1200 .....	52
4.2	Instalação do Laboratório Remoto do ISEL.....	53
4.2.1	Reciclagem do Painel de Cilindros Electropneumáticos .....	53
4.2.2	Ficha DIN 41612 .....	54
4.2.3	Montagem do Autómato S7-1200 .....	55
4.2.4	Montagem da Webcam .....	56
4.3	Páginas Web do Laboratório Remoto do ISEL.....	56
4.3.1	Gestão da Base de Dados.....	57
4.3.2	Web Server S7-1200.....	59
4.3.3	ThinVNC .....	65
4.3.4	Yawcam.....	66
5	RESULTADOS.....	67
5.1	Consultar Conteúdos do Laboratório Remoto.....	67
5.1.1	Base da Dados.....	67
5.1.2	Ambiente de Trabalho Remoto.....	69
5.1.3	Web Server .....	71
5.1.4	Webcam .....	74
5.2	Caso Prático .....	75
5.3	Gestor do Laboratório Remoto.....	83
6	CONCLUSÕES .....	87

6.1	Trabalho Final .....	87
6.2	Trabalhos Futuros.....	88
<b>REFERÊNCIAS BIBLIOGRÁFICAS.....</b>		<b>89</b>



## Lista de Figuras

Figura 2.1 – Esquema de Blocos de <i>hardware</i> de um PLC .....	9
Figura 2.2 – Representação simplificada de um PLC.....	10
Figura 2.3 – Ciclo de Scan .....	11
Figura 2.4 – Ciclo de aprendizagem experimental.....	16
Figura 2.5 – Classificação dos vários tipos de Laboratórios.....	17
Figura 2.6 – Kit Experimental RemoteLab1 .....	23
Figura 2.7 – Laboratório Remoto de Automação Industrial – IST .....	24
Figura 2.8 – Painel de Cilindros, e Interfaces em LabView .....	25
Figura 2.9 – Web-LABAI .....	25
Figura 2.10 – Exemplo de uma experiência do WebLab-Deusto .....	26
Figura 3.1 – Esquema de uma VPN.....	32
Figura 3.2 – Siemens S7-1200 .....	33
Figura 3.3 – <i>Portal view</i> .....	35
Figura 3.4 – <i>Project view</i> .....	36
Figura 3.5 – <i>WampServer, localhost</i> .....	38
Figura 3.6 – Arquitetura do ThinVNC.....	40
Figura 3.7 – Yawcam. ....	42
Figura 4.1 – Arquitetura abstrata e simplificada.....	49
Figura 4.2 – Arquitetura conceptual do Laboratório Remoto do ISEL.....	50
Figura 4.3 – Painel de cilindros utilizado. ....	54
Figura 4.4 – Contactos da ficha DIN 41612 para o S7-1200.....	55
Figura 4.5 – Ficha DIN 41612 finalizada. ....	55
Figura 4.6 – Autómato Siemens S7-1200 e HMI Siemens KTP600. ....	56
Figura 4.7 – Localização da Webcam.....	56
Figura 4.8 – Base de Dados do MySQL no PHPMyAdmin. ....	57
Figura 4.9 – Fluxograma da gestão da base de dados.....	58
Figura 4.10 – Fluxograma das <i>Standard Web Pages</i> do S7-1200.....	59
Figura 4.11 – Fluxograma das <i>User Pages</i> .....	60
Figura 4.12 – Painel com os respetivos endereços.....	62

Figura 4.13 – Botões da página Cilindros.html. ....	64
Figura 4.14 – Imagens utilizadas para representar os cilindros .....	65
Figura 4.15 – Fluxograma do ThinVNC. ....	65
Figura 4.16 – Fluxograma do Yawcam.....	66
Figura 5.1 – IPLNet-VPNIntra.....	67
Figura 5.2 – Index.php. ....	68
Figura 5.3 – <i>Login</i> e Inscrição. ....	68
Figura 5.4 – Escolha ou Menu de Navegação.....	69
Figura 5.5 – <i>Login</i> ThinVNC.....	70
Figura 5.6 – Opções de ligação do ThinVNC.....	70
Figura 5.7 – Ambiente de Trabalho Remoto do ThinVNC.....	71
Figura 5.8 – <i>Web Server</i> S7-1200. ....	72
Figura 5.9 – <i>User Pages</i> .....	72
Figura 5.10 – Página dos cilindros.....	73
Figura 5.11 – Página das variáveis.....	74
Figura 5.12 – Webcam. ....	75
Figura 5.13 – Ciclo Electropneumático. ....	76
Figura 5.14 – Esquema em linguagem gráfica SFC do ciclo Electropneumático. 76	
Figura 5.15 – Diagrama Temporal do Ciclo Electropneumático. ....	76
Figura 5.16 – Lista de <i>tags</i> .....	77
Figura 5.17 – Bloco de programação OB1. ....	78
Figura 5.18 – “ <i>Function block</i> ” .....	78
Figura 5.19 – Programação do ciclo A+/B+/C+/D+ A-/D- B-/C-. ....	79
Figura 5.20 – Fluxograma da programação dos ciclos electropneumáticos. ....	79
Figura 5.21 – Como ligar o <i>web server</i> .....	80
Figura 5.22 – Configurar o IP no TIA Portal. ....	81
Figura 5.23 – Configurar o campo “ <i>User defined Web pages</i> ”. ....	82
Figura 5.24 – Login PHPMyAdmin.....	84
Figura 5.25 – Bases de Dados PHPMyAdmin.....	84
Figura 5.26 – Tabelas de utilizadores PHPMyAdmin. ....	85
Figura 5.27 – Utilizadores do Laboratório Remoto PHPMyAdmin. ....	85

## Lista de Tabelas

Tabela 2.1 – Linguagens de Programação segundo IEC 61131-3. ....	13
Tabela 3. 1 – Características da CPU 1214C AC/DC/Rly .....	33
Tabela 3. 2 – Comparativo de tecnologias de vários Laboratórios Remotos.....	47
Tabela 4.1 – Correspondências dos vários endereços do painel com as TAGS. ..	63



## Lista de Abreviaturas

ARPANET	Advanced Research Project Agency Network
ASCII	American Standard Code for Information Interchange
CNC	Controladores Numéricos Computadorizados
CPU	Central Processing Unit
DML	Data Manipulation Language
DDL	Data Definition Language
FBD	Diagrama de Blocos Funcionais
FTP	File Transfer Protocol
GPRS	General packet radio service
GNU	General Public License
GRAF CET	Grphe Fonctionnel de Commande, Étapes Transitions
HMI	Interface Homem Máquina
HTTP	HyperText Transfer Protocol
IL	Instruction List
IEEE	Institute of Electrical and Electronics Engineers
IEC	International Electrotechnical Commission
IP	Internet Protocol
LLDP	Link Layer Discovery Protocol
LD	Ladder diagram
LAN	Rede de Área Local
MAC	Media Access Control address
MIME44	Multipurpose Internet Mail Extension
MIT	Massachussets Institute of Technology
NCSA	National Center for Supercomputing Applications
PHP	Hypertext Preprocessor
PLC	Controladores Lógicos Programáveis
PID	Proportional Integral Derivative
RFC	Request for Comments
ROM	Read-only memory
SDCD	Sistemas Digitais de Controlo Distribuído
ST	Structural Text

SCADA	Sistemas de Supervisão e Aquisição de Dados
SCL	Structured Control Language
SQL	Structured Query Language
SFC	Sequential Function Chart
SNMP	Simple Network Management Protocol
SEQUEL	Structured English Query Language
TI	Tecnologias da Informação
TCP	Transmission Control Protocol
URI	Universal Resource Identifier
VPN	Virtual Private Network
VNC	Virtual Network Computing
WWW	World Wide Web

# 1 Introdução

Os grandes avanços obtidos nas últimas décadas na expansão e utilização das redes de comunicação têm causado mudanças importantes na nossa sociedade. A área da produção já sofreu significativas mudanças com este desenvolvimento, sendo que a área do ensino enfrenta ainda inúmeros desafios. As diretivas emitidas pela União Europeia em relação a estes desafios são claras. O seu principal objetivo é a transformação do ensino superior de modo a poder tornar-se mais aberto, flexível, dinâmico, multidisciplinar e, sobretudo, capaz de se adaptar de acordo com as mudanças experimentadas pela sociedade (Hackl 2001). Hoje em dia, qualquer profissional deve estar informado e atualizado em relação às inovações tecnológicas. Todo o *know-how* adquirido relativo às novas tecnologias da informação e comunicação são competências muito valiosas para qualquer carreira profissional. Para atingir essas mesmas competências, é extremamente importante incentivar à inovação em metodologias de ensino, para que esses métodos possam estimular a aprendizagem cooperativa e individualizada, de forma autónoma e com novas formas de inter-relação entre os diferentes intervenientes no processo educativo. A educação na área do controlo enfrenta desafios específicos e os laboratórios remotos são úteis recursos para os enfrentar (Domínguez et al. 2008).

## 1.1 Motivação

### Automação de Processos Industriais

Com o crescente nível de qualidade na produção a partir da segunda guerra mundial houve necessidade de se começar a aprofundar os estudos relativos à automação de processos. O grande desenvolvimento, atrever-me-ia mesmo a dizer que quase exponencial, da informática e de outras tecnologias de troca de informação desde então têm dado uma ajuda importantíssima no campo do desenvolvimento da indústria em termos mundiais. Também a crescente competitividade dos mercados no que respeita aos preços reduzidos como na variabilidade de produtos que cada indústria hoje em dia tem de apresentar para ser cada vez mais competitiva, levou a que a automação de processos industriais evoluísse para uma disciplina interdisciplinar onde terá que atuar em várias frentes.

Com o objetivo de colmatar as carências identificadas nesta área, houve a necessidade de inserir a formação para a automação de processos industriais nas várias instituições de ensino de engenharia do país (Pires 2002).

### Ensino à Distância

Neste tempo intensamente tecnológico e globalizado, uma questão de fundo é a de como viver com a tecnologia, e em particular com as tecnologias da informação e da comunicação, tirando delas o melhor partido. Com a evolução da globalização torna-se indispensável a comunicação e troca de informação pela internet. Para tal o desafio dos laboratórios remotos bem como de todo o tipo de ensino à distância é não confinar a troca de informação entre quatro paredes, mas sim o de poder disponibilizar conhecimentos e equipamentos para que cada um possa usufruir de todas as condições para aprender e aplicar conhecimentos independentemente de onde se possa encontrar (Barbosa and Ramos 1997).

A origem do ensino à distância remonta ao século XIX com as universidades a oferecer alguns cursos por correspondência. Com a evolução das tecnologias de informação foi possível na segunda metade do século XX implementar emissões televisivas e vídeos gravados, o que levou a uma melhoria da qualidade deste tipo de ensino. O grande salto é então já nos anos 90 com o início da utilização em massa da internet. Aí sim foi possível um novo conceito de interação entre aluno e professor chamado de *e-learning* (Sousa, Gericota, and Alves 2009).

Com a evolução do ensino à distância nos últimos 150 anos, o ensino à distância experimentou várias experiências e formatos, sendo possível fazer uma retrospectiva das vantagens que o aluno tem com este tipo de ensino (Carmo 1996):

- O que quer aprender (conteúdos de aprendizagem);
- Onde quer aprender (local de aprendizagem);
- Como quer aprender (métodos e media);
- Quando quer aprender (ocasião do dia ou da semana);
- O ritmo a que quer aprender;
- A quem quer recorrer para aprofundar conhecimentos ou colher orientações metodológicas (equipa central ou centros de apoio locais);
- A que sistema de creditação se quer submeter (grau académico, certificado profissional ou certificado de frequência).

Quanto aos formatos organizacionais de ensino observam-se três tipos (Carmo 1996):

- Um **modelo departamental**, que se apresenta sob a forma de unidades orgânicas especializadas, fazendo parte de organizações maiores, de ensino presencial, sob diversas designações (departamento, centro, instituto, etc).
- Um **modelo autónomo**, com o formato de uma organização concebida de raiz, com a finalidade dominante de prestar serviços de ensino e formação à distância.
- Um **modelo de rede**, integrando uma grande diversidade de organizações públicas e privadas, do sector educativo e empresarial que, de comum, têm o objetivo de promover o ensino e a formação à distância.

Percebendo que existia a carência de um laboratório que permitisse aos alunos poderem estar pacatamente nos seus lares podendo aplicar e treinar os seus conhecimentos, procurou-se aplicar todas as potencialidades que a internet nos oferece hoje em dia num laboratório que fosse fácil e prático de utilizar na unidade curricular de Automação de Processos Industriais.

## 1.2 Objetivos

O ensino da automação industrial requer um número elevado de aplicações práticas para que os alunos possam interiorizar todos os conhecimentos teóricos. Como tal, é compreensível que apenas as horas disponibilizadas nos horários para a disciplina, não sejam suficientes para que cada um possa praticar tudo o que necessita. Tendo isso em causa, um laboratório remoto colmataria essa falha, e poderá ainda criar um novo estímulo aos alunos pela sua novidade e comodidade. Com tudo isto em mente foram elaborados os objetivos a que o Laboratório Remoto de Automação de Processos Industriais da ADEM/ISEL se propõe:

- Projetar um laboratório remoto reutilizando e reciclando o material existente no laboratório físico.
- Construir um *website* que seja acessível por um navegador de internet, e permita o acesso remoto do laboratório aos alunos mediante um nome de utilizador e palavra passe.
- Permitir que o aluno possa aceder ao autómato e processos de forma remota e fácil.

- Possibilidade de o utilizador poder fazer *upload* de vários programas para o autómato e poder testá-los nos equipamentos físicos do laboratório.
- Possibilidade de visualização do painel de cilindros em funcionamento em tempo real.
- Permitir o ensino da Automação Industrial, nomeadamente o ensino da electropneumática, com outras abordagens/vertentes, de forma a permitir ao aluno mais horas de contacto, à distância, com o material laboratorial.

Todos os objetivos acima propostos são exequíveis com o material existente atualmente no laboratório físico, não necessitando de grande investimento por parte da Área Departamental de Engenharia Mecânica (ADEM/ISEL) na aquisição de novos equipamentos.

### 1.3 Contribuições

Com a evolução da automação industrial e dos sistemas de informação e comunicação, tornou-se importante o desenvolvimento de uma ferramenta que permitisse aos alunos da ADEM/ISEL trabalhar autonomamente a partir de casa. Tendo em conta que o ensino de engenharia é bastante prático as contribuições desta ferramenta para um futuro próximo são:

- A arquitetura implementada disponibiliza ao aluno a possibilidade de programar remotamente o PLC no seu *software* original, sem ser necessário uma licença de utilização individual;
- Esta mesma arquitetura e o autómato S7-1200 de Siemens permite trabalhar em desenvolvimento de sinópticos.
- A possibilidade de integração de novas experiências já existentes no laboratório físico, como por exemplo o pórtico de três eixos.
- A possibilidade de utilização prática de uma rede industrial de alto nível, como o TCP/IP, no ensino de automação industrial;
- A criação de um *web server* para fazer toda a gestão dos vários utilizadores do laboratórios remoto e a possibilidade de editar e controlar estes mesmos utilizadores.

## **1.4 Estrutura do Trabalho**

O trabalho divide-se em seis capítulos.

No Capítulo 1 Introdução, fala-se da motivação, dos objetivos propostos e da estrutura para a elaboração deste trabalho.

Capítulo 2 Estado da Arte, onde se encontram alguns conceitos mais gerais mas essenciais para a compreensão do trabalho desenvolvido, bem como dos tipos de laboratórios existentes.

Capítulo 3 Tecnologias Implementadas e Montagem, incide nos conceitos específicos utilizados para a elaboração do trabalho e das tecnologias utilizadas para a implementação do laboratório remoto.

Capítulo 4 Laboratório Remoto do ISEL, explica o tipo de arquitetura implementada, como foi feita a instalação do laboratório remoto e como e porquê foram elaboradas as páginas para o navegador de internet.

Capítulo 5 Resultados, são apresentados todos os passos para a utilização de uma forma correta do laboratório remoto por parte do utilizador bem como o gestor pode gerir todos os utilizadores.

Capítulo 6 Conclusões, faz uma reflexão dos objetivos propostos no capítulo 1, retirando as respetivas conclusões do trabalho elaborado.



## 2 Estado da Arte

As experiências laboratoriais e as suas avaliações são essenciais no estudo de qualquer aluno de engenharia. Elas fazem com que o aluno consolide toda a matéria teórica envolvida em cada experiência.

Existem vários tipos de laboratórios experimentais, os laboratórios presenciais, laboratórios remotos, laboratórios virtuais e laboratórios híbridos. Cada um deles tem as suas especificidades, vantagens e desvantagens. Este capítulo tenta assim fazer um estado da arte em termos dos laboratórios existentes.

Para uma melhor compreensão do que é um laboratório remoto será efetuado um enquadramento do que já existe em termos nacionais e internacionais.

### 2.1 Automação Industrial

Resumidamente a automação de processos industriais ou simplesmente automação industrial é a aplicação de tecnologias, sejam estas em termos de sensores e atuadores ou de *softwares*, numa máquina que é utilizada num determinado processo, ou no controlo desse mesmo processo industrial de forma a aumentar a sua eficiência, baixar os custos, aumentar a produção, reduzir os gastos de matéria-prima e energia associada, aumentar a segurança, entre outros.

Para que haja automação temos de ter instrumentos e dispositivos para a poder pôr em prática. Esses equipamentos têm de ter capacidade de processamento de operações lógicas, como são exemplo os Controladores Lógicos Programáveis (PLC), Microcontroladores, Sistemas Digitais de Controlo Distribuído (SDCD) ou os Controladores Numéricos Computadorizados (CNC). Estes dispositivos têm de ser flexíveis, capazes de responder a alterações de produto, daí poderem ser programáveis com auxílio a *softwares* apropriados para cada dispositivo.

Várias indústrias utilizam hoje em dia estas tecnologias de uma forma corrente e em alguns casos mesmo dependente, como são o caso da:

- Indústria automóvel, estampagem, soldadura, pintura;
- Indústria de moldes, plásticos, vidros e cerâmicas;
- Indústria química, papel, madeira;

Em todas as indústrias referidas existem ainda áreas comuns que utilizam processos de automação industrial como as operações de paletização, empacotamento, transporte de matérias, controlo de qualidade e armazenamento.

Quando se pretende automatizar algum processo existem várias etapas que não podem ser descoradas, nomeadamente:

- Identificar processos e racionalizar, ou seja, conhecer o processo a fundo, identificar as suas diferentes fases e sequência das mesmas e por fim simplificar e normaliza-los;
- Automatizar os processos individuais, ou seja, criação de células onde os processos são executados separadamente;
- Integrar, ou seja, interligar todo o sistema produtivo ou sistemas produtivos, de forma a conseguir maior integração de todos os processos.

A automação envolve vários tipos de dispositivos, cada um com a sua função específica:

- Computadores e periféricos, ou seja, computadores pessoais, servidores, PLC's, HMI's (Interface Homem Máquina), entre outros;
- Equipamento de produção, ou seja, robôs, máquinas CNC, transportadores, sensores e atuadores de vários tipos e funções, entre outros;
- Redes locais, ou seja, protocolos de suporte físico que permitam ligar vários computadores, PLC's e outros dispositivos em rede;
- *Software*, ou seja, bases de dados, sistemas operativos, programas de configuração dos PLC's, entre outros. (Pires 2002)

## **2.2 Controladores Lógicos Programáveis**

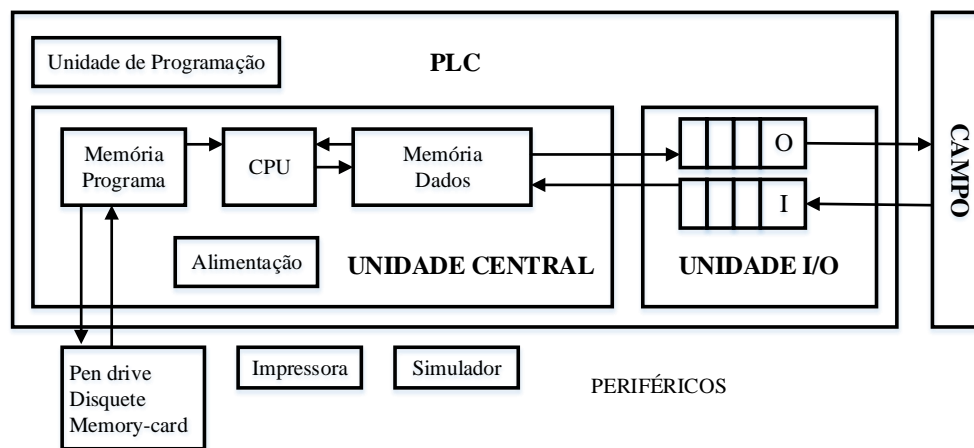
O dispositivo PLC tem na sua constituição várias partes e cada uma delas com a sua função específica:

- Unidade central, que organiza todas as ações de controlo;
- Unidade de entrada e saída I/O, do inglês *Input* (entradas) e *Output* (saídas), é a interface que liga a unidade central do PLC e os vários sensores e atuadores;

- Unidade de programação, é a interface do utilizador para comunicar com o PLC, ou seja, que permite fazer toda a programação lógica desejável, pode ser feito por uma consola, um computador pessoal ou teclado apropriado.

Existem ainda alguns periféricos que nos ajudam a complementar todo o PLC como são exemplo:

- Os simuladores I/O, serve para simular entradas e saídas no PLC quando se pretende experimentar algum programa;
- Impressoras, permite imprimir o programa;
- Dispositivos de grande memorização de dados, são dispositivos que permite aumentar a memória do PLC como disquetes, *pen drive* ou *memory-card*.



**Figura 2.1 – Esquema de Blocos de hardware de um PLC** (Prudente 2007).

A Figura 2.1 tenta representar um PLC e todos os seus periféricos, desde que este recebe informações recolhidas dos vários sensores instalados no “campo”, até ao tratamento dessa mesma informação pelo PLC e posterior atuação através dos atuadores instalados no “campo”.

Dentro do PLC existem várias características que merecem alguma atenção, nomeadamente:

- Microprocessador e CPU;
- Ciclo de Scan;
- Sistema Operacional ou *Firmware*;
- Memória.

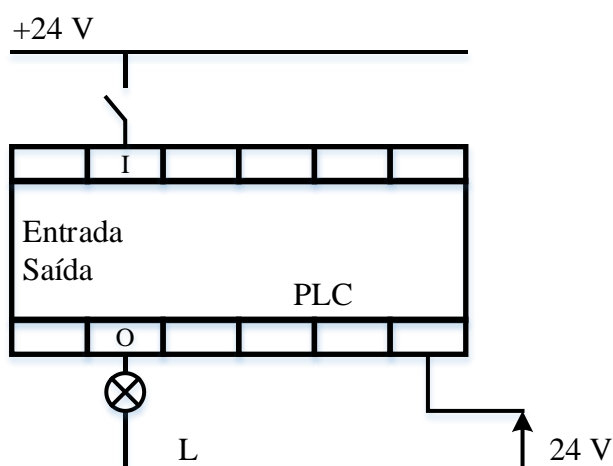
## Microprocessador e CPU

O microprocessador é a parte responsável pelas operações lógicas e matemáticas de forma rápida, e como tal torna-se no elemento mais complexo do PLC. As operações são executadas de forma sequencial, ou seja, pela mesma ordem que foram elaboradas no programa.

O microprocessador é então a parte mais importante da CPU (*Central Processing Unit*). A CPU é responsável pelo desempenho do PLC, ou seja, pela sua velocidade de processamento, potência, entre outros.

A CPU é então a responsável por fazer a aquisição dos sinais elétricos adquiridos dos vários sensores instalados, processar os mesmos e ver quais as alterações a fazer consoante as operações lógicas descarregadas para a CPU e agir em conformidade, ou seja, enviando todos os sinais elétricos para poder acionar os vários atuadores instalados.

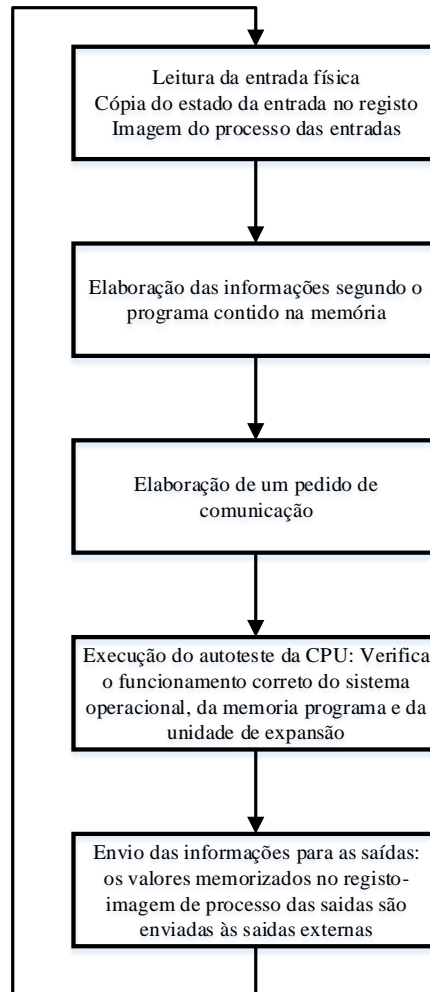
A Figura 2.2 representa de forma simplificada o modo como o PLC interpreta as entradas e saídas.



**Figura 2.2 – Representação simplificada de um PLC (Prudente 2007).**

### Ciclo de Scan

A ordem com que a CPU executa todas as operações lógicas chama-se ciclo de scan. O tempo que leva até executar todas estas operações chama-se tempo de scan. O tempo de scan é uma das características mais importantes aquando da aquisição de um novo PLC.



**Figura 2.3 – Ciclo de Scan** (Prudente 2007).

A Figura 2.3 tenta reproduzir todos os passos que o PLC faz durante o ciclo de scan.

Um tempo também importante de referir é o tempo de resposta, ou seja, o tempo que vai desde a variação das entradas até á variação das saídas (Prudente 2007).

### Sistema Operativo ou *Firmware*

O sistema operacional ou *firmware* é o programa que a CPU utiliza para poder executar todas as operações lógicas requeridas. Apenas os construtores têm acesso a este *software*, sendo que os utilizadores podem fazer *upload* de versões mais atuais para a CPU. Este *software* está gravado numa memória com o nome de ROM. Entre outras este sistema operativo é responsável por:

- Autodiagnóstico;
- Proteção dos dados;
- Função de Interrupção (*interrupt*).

O auto diagnóstico é uma verificação que o PLC faz sobre o funcionamento do seu circuito interno, de forma a saber se existe algum problema com o mesmo. Caso se confirme algum problema o PLC deverá emitir um alarme de aviso ao utilizador.

A proteção de dados é nada mais nada menos que uma bateria que todos os PLC's têm de forma a assegurar que toda a informação contida não se perde em caso de falta de energia. Esta bateria terá que ser substituída quando o PLC deteta que a mesma não está em condições pelo auto diagnóstico.

Função de interrupção acontece quando um determinado sinal vindo dos sensores obriga a CPU a interromper o ciclo normal de funcionamento para executar outro com uma prioridade mais elevada de forma a corrigir alguma coisa que não estivesse de forma adequada ao esperado. Terminada esta operação o PLC retorna ao seu ciclo de funcionamento normal (Prudente 2007).

## Memória

Para que toda a dinâmica envolvida entre as entradas e as saídas da CPU seja possível de concretizar, o autómato tem de “ler” e “escrever” as informações, e isto apenas é possível porque existem memórias. Algumas das características associadas às memórias são:

- Possibilidade de ler e escrever;
- Velocidade de escrita/leitura;
- Modalidade de cancelar;
- Comportamento em caso de falta de energia;
- Quantidade de informação memorizável (capacidade).

Uma das características a ter em conta aquando da compra de um PLC é a sua capacidade de memória. Essa depende da dimensão dos processos industriais que se pretende automatizar. Tendo isso em conta terá que se ter uma previsão de forma a poder prevenir a expansão futura do PLC de forma a não ter que se comprar um novo autómato. (Prudente 2007)

### 2.2.1 Linguagens de Programação

De forma a poder haver cada vez mais uniformização nas linguagens de programação, a *International Electrotechnical Commission* (IEC), que é uma instituição de normalização para tecnologias elétricas, eletrônicas e relacionadas, teve de intervir e

fazer mesmo uma norma dedicada às linguagens de programação para que cada construtor de PLC's não venha a ter a sua própria linguagem de programação.

A norma IEC 61131-3 resume todas as linguagens de programação aceites por esta instituição. Esta faz logo uma distinção à partida como mostra a figura abaixo.

**Tabela 2.1 – Linguagens de Programação segundo IEC 61131-3.**

Linguagens de Programação	
Linguagem Gráfica	Linguagem Textual
<ul style="list-style-type: none"> <li>• Diagrama Ladder</li> <li>• Diagrama de Blocos Funcionais</li> </ul>	<ul style="list-style-type: none"> <li>• Lista de Instruções</li> <li>• Texto Estruturado</li> </ul>
<ul style="list-style-type: none"> <li>• Sequential Function Chart</li> </ul>	

### Linguagem de Programação Textual

- Lista de Instruções – IL (*Instruction List*):

Este modo de programação utiliza uma sequência de instruções tipo texto de baixo nível. É constituída por uma série de códigos “máquina”, ou seja, utiliza as instruções do processador diretamente.

- Texto Estruturado – ST (*Structural Text*):

Este modo de programação utiliza como o seu próprio nome o indica texto estruturado de alto nível, ou seja, linguagens de programação ao nível de Pascal, C++, Basic, entre outras. É o tipo de linguagem mais potente para PLC, aquela que mais partido pode tirar, mas contra tem o facto de ser a mais complicada de entender e programar. Esta requer alguns conhecimentos de programação avançada.

### Linguagem de Programação Gráfica

- Diagrama Ladder – LD:

Como o nome assim o indica, do inglês *ladder* (escada), o tipo de representação desta linguagem faz-se tipo escada, com uma barra vertical do lado esquerdo chamada barra de alimentação, com uma barra vertical do lado direito chamada de retorno comum ou massa e com uma linha horizontal dividida em duas partes, sendo que a do lado esquerdo é a zona de *input* destinada às entradas e a do lado direito é a zona de *output* destinada às saídas. Esta é talvez a linguagem de programação mais utilizada para a programação de PLC's a nível internacional e nacional.

- Diagrama de Blocos Funcionais – FBD:

É uma linguagem constituída por uma série de blocos que utilizam simbologia lógica combinatória (AND, OR, XOR, etc). É mais utilizada por quem tem prática na eletrónica digital e lógica, onde a simbologia é a mesma.

- Sequential Function Chart – SFC:

Primeiramente chamada Grafcet (acrónimo do francês Graphe Fonctionnel de Commande, Étapes Transitions), este tipo de linguagem é diferente das anteriores, pelo modo como são utilizados os diagramas funcionais, tendo até uma norma apenas para descrever estas mesmas regras a Norma IEC 60848. Esta baseia-se no sequenciamento das tarefas a executar pelo PLC de forma automática, ou seja, as instruções contidas nas transições apenas são cumpridas quando as condições inseridas no passo anterior estiverem satisfeitas. Desta forma cria-se uma sequência de eventos que são executados passo a passo. É escrita com linguagem gráfica e utiliza comandos alfanuméricos (Prudente 2007).

## 2.3 Redes Industriais

Em automação de processos industriais entende-se por rede industrial todos os protocolos de comunicação *standard* utilizados para monitorização, deteção e controlo dos processos industriais, com troca de informação entre os vários dispositivos. Estes protocolos permitem termos computadores ligados a PLC's para por exemplo guardar bases de dados, ou ter PLC's ligados entre si para trocarem informações relativas ao processo a controlar.

Com a implementação da automação de processos e evolução da informática, as indústrias sentiram a necessidade de implementarem também estes sistemas de comunicação para poderem controlar toda a informação relativa aos processos, seja para sistemas de controlo de produção ou manutenção, seja apenas para supervisionar o processo em tempo real (A. Lopes, Afonso, and Antunes 2005).

Existem várias configurações para a comunicação entre dispositivos, uma comunicação implica perguntas e respostas e existem protocolos que permitem perguntas e respostas (*multi-master*), e outros em que cada dispositivo tem o seu papel já definido (*master/slave*) (Pinto 2004).

## Redes de Campo (FIELDBUS)

Entende-se redes de campo como aquelas que têm robustez para transmitir dados enquanto expostas a ambientes adversos, interferências eletromagnéticas, entre outras, como são o caso dos ambientes industriais

Como referido estas terão que ter algumas características próprias para lidar com este tipo de ambientes adversos:

- Robustez física
- Tipo de mecanismo de transporte
- Desempenho

A robustez física engloba todas as características como a topologia das redes, meio físico que as suporta, número máximo de nós permitidos e a distância máxima que pode cobrir.

O tipo de mecanismo de transporte tem em conta os fatores que condicionam a forma como a informação é transmitida entre os vários dispositivos, ou seja, protocolos de comunicação, velocidades de transmissão, dimensão dos dados a transferir, controlo e diagnóstico de erros.

O desempenho relaciona-se com o tempo de transmissão dos dados.

Algumas das redes de campo existentes são:

- ASI
- World FIP
- CANOpen
- DeviceNet
- Interbus-S
- LonWorks
- PROFIBUS-DP
- PROFIBUS-PA

## Ethernet TCP/IP

Este tipo de rede utiliza o protocolo TCP/IP como base para comunicar com os vários dispositivos. O suporte físico para este transporte chama-se *ethernet*. A grande vantagem deste tipo de comunicação é a sua conjugação de meios físicos em que se pode processar, com velocidades diferentes de envio de informação, bem como a possibilidade de utilização de redes *wireless* para locais onde a acessibilidade via cabo não é possível.

Outra grande vantagem é a possibilidade de utilização de equipamentos informáticos ditos normais, como o nosso computador pessoal.

Este tipo de tecnologia tem-se tornado bastante utilizado pelo seu elevado desenvolvimento e grande utilização a nível mundial (Pinto 2004).

## 2.4 Tipos de Laboratórios

Aprender com recurso a uma experiência envolve componentes que vão desde o fazer até ao pensar. De acordo com (Centre 2012), aprender através de uma experiência tem quatro fases colocadas num ciclo, como ilustrada na Figura 2.4. O aluno pode entrar em qualquer ponto do ciclo, mas as fases devem ser seguidas em sequência. Existe uma diferença entre o fornecimento de informação para os alunos e o testar essa mesma informação numa experiência. Intuição, imaginação, interatividade são consideradas como peças chave na compreensão dos processos. Desta forma, os alunos adquirem e desenvolvem o seu próprio conhecimento através de um desafio, a experiência.

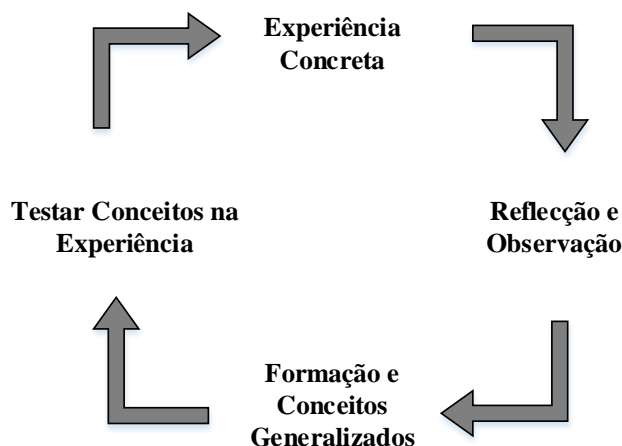


Figura 2.4 – Ciclo de aprendizagem experimental (Centre 2012).

Um estudo recente sobre a utilidade dos laboratórios remotos nas escolas secundárias (Lowe, Newcombe, and Stumpers 2013) fala do contexto que as experiências podem ter para influenciar os resultados da aprendizagem de um aluno. Os laboratórios remotos permitem que alunos e professores utilizem redes de alta velocidade, juntamente com câmaras, sensores e controladores, para realizarem experiências laboratoriais no laboratório físico real, que está localizado remotamente. A pesquisa mostrou que, quando usado apropriadamente pode trazer uma série de benefícios, incluindo a partilha de recursos entre várias instituições e apoiar o acesso às instalações que seriam inacessíveis por razões de custo ou técnicas.

Ensino experimental é e sempre foi uma das preocupações constantes no ensino da engenharia em particular. Permite observar e explorar as aplicações do mundo real, das teorias fundamentais e desenvolver uma melhor compreensão das aulas teóricas (Jara et al. 2011).

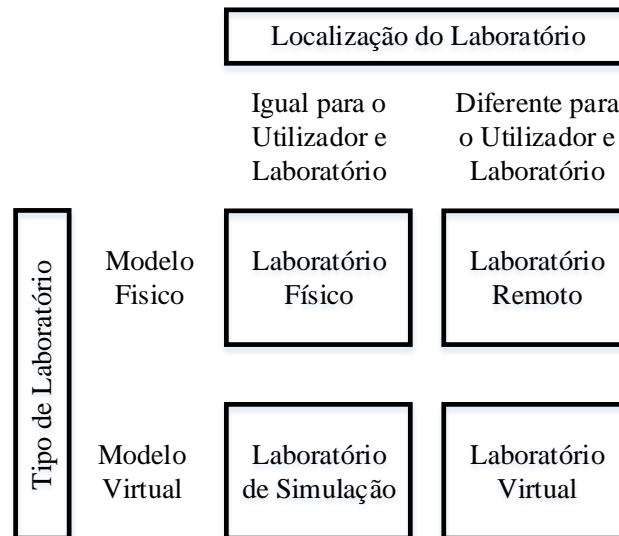
Os três principais critérios de classificação para o laboratório e utilizador são:

1. O tipo de interação do utilizador com o laboratório;
2. A natureza do laboratório;
3. O tipo de local do utilizador e do laboratório.

Em relação ao primeiro critério, dois tipos de interação utilizador/laboratório são possíveis:

1. Controlar diretamente os equipamentos, o que corresponde ao laboratório físico tradicional;
2. Controlar os equipamentos através de uma interface, usando instrumentação virtual ou *softwares* de realidade virtual.

Os outros dois critérios são frequentemente combinados, resultando na definição de quatro tipos de laboratórios, conforme ilustrado na Figura 2.5.



**Figura 2.5 – Classificação dos vários tipos de Laboratórios.**

Existe ainda o laboratório híbrido que consiste numa subclasse de laboratórios onde a localização do utilizador é diferente do laboratório. Representa um laboratório *on-line* que fornece simulações por meio de *software* e experiências reais remotamente num único ambiente (Gomes and Bogosyan 2009).

### 2.4.1 Laboratório Físico

O laboratório físico é o mais comum, sendo o mais antigo. Os laboratórios presenciais fazem parte do percurso escolar/educativo de qualquer aluno da área científica. O aluno presencia as experiências ou simulações sob condições ambientais controladas e normalizadas, de modo a assegurar que não ocorrem influências estranhas que possam alterar os resultados. Pode-se fazer medições, correções da experiência, retirar valores para análise, entre muitas outras tarefas, ficando com uma percepção mais real de todo o ambiente envolvente à mesma.

#### Áreas de Aplicação

Este tipo de laboratório é praticamente transversal a todo tipo de ciências, desde as engenharias, à biologia, à física, entre outras.

#### Vantagens e Desvantagens

As vantagens deste tipo de laboratório são:

- O aluno pode presenciar a experiência com todas as condicionantes físicas que ela possa envolver;
- Noção prática das técnicas e protocolos necessários para o sucesso da experiência;
- Interagir e familiarizar com os equipamentos e ferramentas de laboratório comumente utilizadas na prática;
- Poderá repetir as experiências as vezes que forem necessárias;
- As experiências podem ser executadas simulando as condições reais, o que permite uma enorme versatilidade nas experiências (Corter et al. 2011).

As desvantagens são:

- O alto investimento para a montagem dos laboratórios;
- Os custos relativos com a manutenção do laboratório e com a sua segurança;
- Os horários a que se pode fazer as experiências serem por vezes pouco flexíveis;
- A presença quase sempre obrigatória de alguém para supervisionar as experiências;
- Ser deslocado geograficamente muitas vezes dos locais de trabalho;

- Algumas experiências precisam de consumíveis que podem ser bastante caros;
- Por vezes torna-se complicado simular os ambientes adversos em que verdadeiramente ocorrem algumas das simulações pretendidas (Ma and Nickerson 2006).

### 2.4.2 Laboratório Remoto

Laboratório remoto significa a utilização de telecomunicações para realizar remotamente uma experiência ou simulação com um ambiente real, a localização física dos equipamentos a ensaiar estão deslocados do local onde o aluno se encontra, ou seja, o local geográfico onde a experiência ocorre é diferente do local de quem realiza a mesma.

Arquitetura de um laboratório remoto conta com o paradigma cliente-servidor, onde as tecnologias de comunicação por internet estão no centro. Mesmo usando diferentes tecnologias, a maioria das soluções de laboratórios remotos contam com arquiteturas de *softwares* simples, que permitem o controlo da experiência e a ligação com a rede, atuando como um servidor do laboratório. Na outra extremidade, o utilizador vai usar um programa tipo de cliente (para a maioria dos casos páginas escritas em HTML, lidas pelo navegador *web*), que oferece o controlo remoto e os recursos de monitorização da experiência (Gomes and Bogosyan 2009).

#### Áreas de Aplicação

Os laboratórios remotos têm vindo a ganhar algum terreno na área de instrumentação por exemplo (Restivo et al. 2009), onde é possível remotamente a recolha de valores. Outra das áreas muito utilizadas pelos laboratórios remotos é a robótica e a automação industrial ao nível do ensino da engenharia, onde é possível fazer programações de equipamentos e posteriormente pode ver o resultado (Jara et al. 2011). A eletrónica é outra das áreas que tem vindo a apostar neste tipo de laboratório (Sousa, Gericota, and Alves 2009).

#### Vantagens e Desvantagens

As vantagens são:

- A flexibilidade que estes oferecem aos alunos em termos de tempo;
- O acesso ao laboratório pode ser de 24 horas, 7 dias por semana;
- Permite uma melhor programação das atividades durante o semestre;

- Permite experimentar sob diversas condições, sem medo de danificar o equipamento ou o utilizador sair ferido;
- Permite diversas repetições das experiências para aprender com os erros ou para ter mais dados para análise;
- Retorno do investimento devido à partilhada de recursos;
- Permite colaborações de ensino e investigação entre instituições em todo o mundo;
- Estimula a aprendizagem e procura de conhecimento autónomo;
- Impede que haja danos nos equipamentos pelo mau manuseamento por parte dos utilizadores.

As desvantagens são:

- A falta de noção de segurança no manuseamento dos equipamentos;
- A interação com os equipamentos é reduzida;
- Não existe uma troca de experiências com outros colegas de grupo ou de docentes para alunos;
- Não existe uma noção de preparação da experiência, os laboratórios remotos normalmente já se encontram preparadas para uso imediato;
- A não possibilidade de vários alunos fazerem a mesma experiência por impossibilidades físicas (Gomes and Bogosyan 2009).

### 2.4.3 Laboratório Virtual

Os laboratórios virtuais consistem em simulações de dispositivos físicos por meio de *software*. São as imitações de experiências reais. Todas as infraestruturas necessárias para o laboratório não são reais, são simuladas em computador. Estes começam a ser vistos como uma alternativa de ensino com as despesas elevadas dos laboratórios físicos (Gomes and Bogosyan 2009).

Este tipo de laboratório é utilizado por vezes antes de testar o sistema real, para minimizar os estragos que os alunos possam causar sem qualquer tipo de experiência (Benmohamed, Leleve, and Prevot 2006).

## Áreas de Aplicação

Este tipo de laboratório é amplamente utilizado nas áreas das ciências, nomeadamente nas físicas. Também nas engenharias estes laboratórios têm alguma aplicação como nas áreas das termodinâmicas.

Outra das áreas onde as simulações virtuais são utilizadas é na aviação e no automobilismo. Os pilotos de aviação treinam em simuladores totalmente virtuais para se habituarem aos comandos do avião. Na fórmula 1, por exemplo, existem vários construtores com simuladores virtuais para que os pilotos possam aprender o circuito com maior facilidade, entre outras utilidades (Benmohamed, Leleve, and Prevot 2006).

## Vantagens e Desvantagens

As vantagens dos laboratórios virtuais são:

- As simulações virtuais reduzem supostamente o tempo que demora a aprender uma determinada matéria ou tarefa;
- Proporcionam um ambiente de laboratório sem as preocupações de segurança relacionadas com os equipamentos reais.

As desvantagens são:

- A exposição excessiva a simulações pode resultar numa desconexão entre a realidade física e virtual;
- Alguns laboratórios virtuais podem não simular a realidade de forma clara, o que leva por vezes, os alunos a não aprenderem por tentativa e erro;
- O custo de simulação não é necessariamente menor do que o custo de ter um laboratório físico, pois nalguns destes as despesas e tempo de desenvolvimento do *software* podem-se refletir no seu custo final.
- Quanto maior a complexidade da experiência a simular mais difícil será de simular a proximidade com a realidade (Ma and Nickerson 2006).

## 2.5 Exemplos de Laboratórios Remotos

De seguida serão apresentados exemplos de laboratórios remotos nacionais e internacionais tomados como relevantes do ponto de vista das tecnologias utilizadas e das funcionalidades disponíveis para os seus utilizadores.

Em termos de laboratórios nacionais:

- Laboratório Remoto de Automação Industrial (Lab-RAI) do Departamento de Engenharia Eletrotécnica da Faculdade de Ciências e Tecnologias da Universidade Nova de Lisboa;
- Laboratório Remoto de Automação Industrial do Instituto Superior Técnico da Universidade Técnica de Lisboa;

Em termos de laboratórios internacionais:

- Laboratório Remoto de Automação Industrial da Faculdade de Engenharia Mecânica e Eletrotécnica da Universidade Veracruz no México.
- Laboratório Remoto Web-LABAI da Universidade Politécnica " Antonio José de Sucre " UNEXPO, de Barquisimeto na Venezuela.
- Laboratórios Remotos WebLab-Deusto da Universidade de Deusto.

Houve ainda mais alguns laboratórios que foram considerados mas não em pormenor como estes, nomeadamente os laboratórios do *Massachusetts Institute of Technology (MIT)* (Hardison et al. 2005) e (Harward et al. 2004) na área da eletrotecnia, os laboratórios da Faculdade de Engenharia da Universidade do Porto na área da metrologia (Restivo et al. 2009) e de redes de computadores (Oliveira 2008) e (Coelho 2001) e ainda um laboratório remoto em física da Faculdade de Ciências e Tecnologia da Universidade de Coimbra (S. P. de M. L. Lopes 2007).

### 2.5.1 Nacionais

#### Laboratório Remoto de Automação Industrial (Lab-RAI)

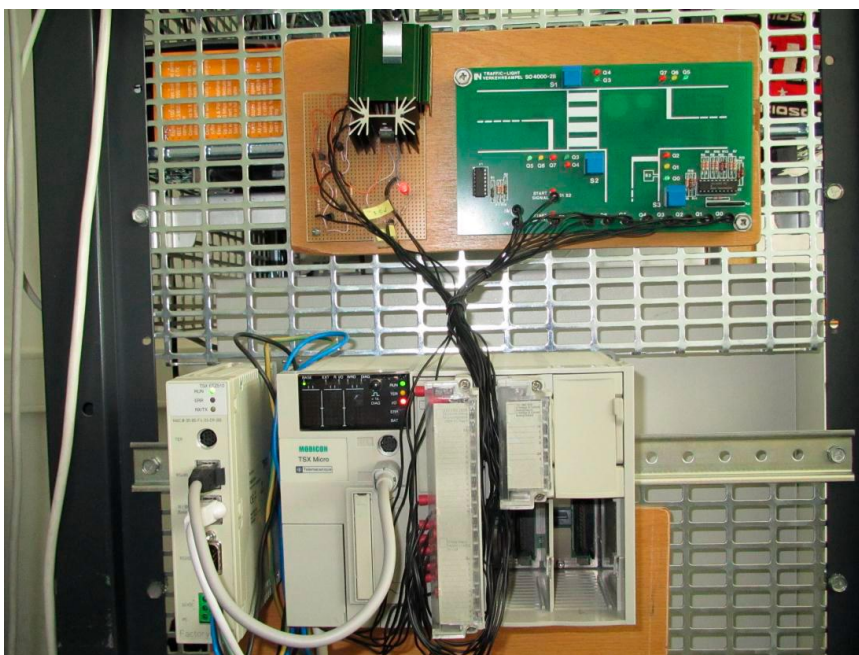
O Laboratório Remoto de Automação Industrial (Lab-RAI) da Faculdade de Ciências e Tecnologias da Universidade Nova de Lisboa apresenta, foi implementado tendo como um dos seus objetivos gerais o ensino da Automação Industrial. Daí houve a necessidade de prestar um apoio mais teórico com o disponibilizar e gerir conteúdos teóricos (sebentas, apresentações, enunciados de exercícios, manuais, etc...).

Em termos práticos o laboratório permite ao utilizador fazer até três experiências diferentes. Com o elevado número de alunos teve que haver uma gestão dos utilizadores e das experiências disponíveis. Essa mesma gestão foi efetuada da seguinte forma:

- Um servidor que fará a gestão de utilizadores e dos recursos disponíveis.
- Vários recursos laboratoriais, que serão geridos e disponibilizados separadamente, ou seja, outro computador por cada experiência remota.

As tecnologias utilizadas da parte do laboratório foram por exemplo para a base de dados o MySQL, para as páginas *web* o HTML e PHP, com pequenos recursos a *JavaScript*.

Para o utilizador as tecnologias a utilizar são o *browser* Internet Explorer ou equivalente com suporte para controlo ActiveX e uma instalação do JVM (*Java virtual machine*), é necessário ainda uma ligação á internet (Borracha 2012).



**Figura 2.6 – Kit Experimental RemoteLab1** (Borracha 2012).

A Figura 2.6 mostra uma das três experiências que são possíveis de fazer dentro do laboratório remoto.

### Laboratório Remoto de Automação Industrial – IST

O Laboratório Remoto de Automação Industrial do Instituto Superior Técnico da Universidade Técnica de Lisboa está mais dedicado ao ensino da electropneumática.

Os alunos com base no *software* dos autómatos SAIA utilizados têm acesso a uma interface que lhes permita interagir com o painel demonstrado na figura 2.7.

A ligação dos alunos ao laboratório é feita via VPN por uma questão de facilidade e protecção.

As tecnologias utilizadas do lado do utilizador são os *softwares* da SAIA juntamente com o navegador de internet. As tecnologias do laboratório são todas com recurso uma vez mais aos *softwares* SAIA.

Por navegador de internet os alunos têm acesso a uma interface com vários botões que devidamente programados permitem ter uma interação com o autómato. Permite

ainda ter uma percepção dos movimentos das hastes através de pequenas animações com o desenho dos cilindros (Ribeiro 2012).

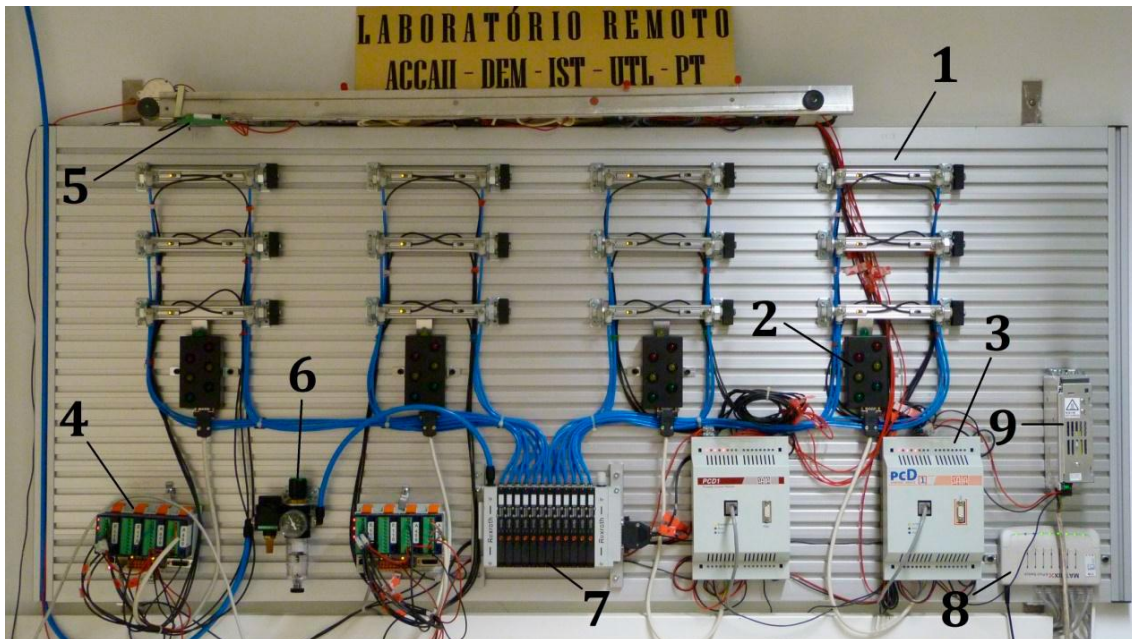


Figura 2.7 – Laboratório Remoto de Automação Industrial – IST(Ribeiro 2012).

## 2.5.2 Internacionais

### Laboratório Remoto de Automação Industrial da Universidade Veracruz no México

O Laboratório Remoto de Automação Industrial da Faculdade de Engenharia Mecânica e Eletrotécnica da Universidade Veracruz no México apresenta uma solução interessante para o ensino da electropneumática. Este faz uma abordagem física com um painel de cilindros electropneumáticos normal, mas para além disso faz uma mistura com algumas aplicações virtuais ligadas ao autómato por meio de uma placa de aquisição de sinais feitas com auxílio do *software* LabView.

Acesso remoto ao laboratório faz-se via *web*, para o ambiente virtual foi utilizado o *web server* do LabVIEW. Este servidor permite que vários utilizadores acedam ao painel frontal simultaneamente e em tempo real, mas apenas um deles pode controlá-lo e realizar a experiência (Villa-López et al. 2013).

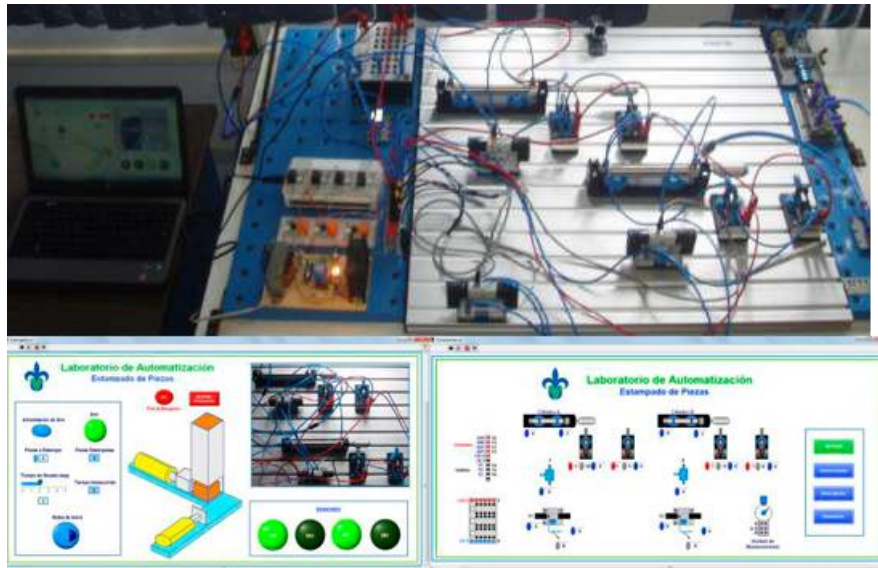


Figura 2.8 – Painel de Cilindros, e Interfaces em LabView (Villa-López et al. 2013).

### Laboratório Remoto (Web-LABAI)

O Laboratório Remoto da Universidade Politécnica " Antonio José de Sucre " UNEXPO, permite aos alunos realizar, a partir da Internet, práticas controlo e automação de processos através de um sistema de *hardware/software* que permite a monitorização e controlo de variáveis, remotamente a partir de uma página *web*, com um modelo de um processo industrial que simula um processo de fabricação de alimentos.

O *hardware* é composto por um servidor, um autómato e uma câmara IP. O *software* é constituído por um servidor Modbus TCP/IP incorporado no autómato e de um servidor *web* que contem as páginas *web* através da qual os utilizadores acedem ao laboratório remoto. A aplicação do utilizador consiste num conjunto de *applets* Java que podem controlar e monitorar o sistema, além da possibilidade de poder ver o processo em tempo real, através da câmara IP (De la Cruz et al. 2010).

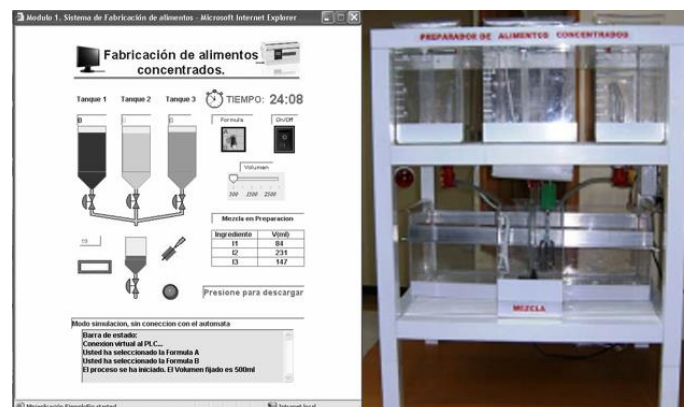


Figura 2.9 – Web-LABAI (De la Cruz et al. 2010).

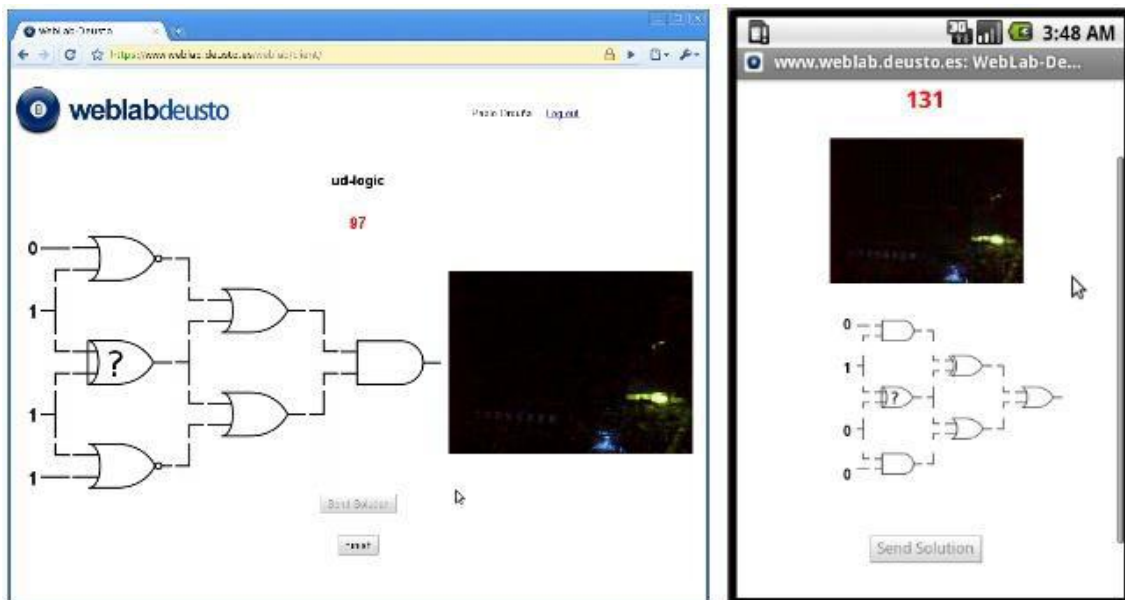
## Laboratórios Remotos (WebLab-Deusto)

Os laboratórios remotos da universidade de Duesto são baseados em *software* aberto, e é usado pelos alunos desta instituição desde 2005. É considerado pela universidade uma ferramenta de ensino bastante importante, tendo esta uma longa tradição neste tipo metodologia de ensino prático, disponibilizando variadas experiências diferentes relacionadas com engenharia.

WebLab-Deusto fornece uma arquitetura de *software* distribuído que facilita a integração de novas experiências. Existem dois tipos:

- Gerenciado – A experiência utiliza uma das tecnologias *web* comuns do lado do utilizador (JavaScript , Flash, Java applets), e toda a tecnologia do lado do servidor (WebLab-Deusto vem com bibliotecas para C/C++, NET ,LabVIEW, Java e Python).
- Não gerenciado – A experiência coloca uma aplicação em máquina virtual, e o WebLab-Deusto gerência o controlo de acesso a essa mesma máquina virtual.

A Figura 2.10 apresenta um pequeno exemplo de como se tem vindo a desenvolver este laboratório, tendo já aplicações móveis e integração com as redes sociais (Deusto 2013).



**Figura 2.10 – Exemplo de uma experiência do WebLab-Deusto em telemóvel ou Tablet e computador (Deusto 2013).**

## 3 Tecnologias Implementadas e Montagem

O laboratório remoto do ISEL surge da necessidade de inovar e criar novas ferramentas de ensino no laboratório de automação industrial já existente na Área Departamental de Engenharia Mecânica do ISEL. Neste capítulo serão abordadas todas as tecnologias utilizadas na implementação do Laboratório Remoto.

### 3.1 Revisão de Conhecimentos

Para um melhor enquadramento e compreensão do trabalho desenvolvido será efetuado de seguida uma breve revisão de algumas matérias essenciais para a elaboração deste trabalho.

#### 3.1.1 Protocolo HTTP

A necessidade de distribuir e trocar informações na Internet fez com que fosse possível surgir uma forma padronizada de comunicação entre utilizadores e servidores e que fosse compatível e entendida por todos os computadores ligados á Internet. Daí surge o protocolo HTTP (*HyperText Transfer Protocol*) passando a ser utilizado para a comunicação entre computadores na Internet e a especificar como seriam realizadas as transações entre clientes e servidores, através do uso de regras básicas.

Este protocolo tem sido usado pela WWW desde 1990. A primeira versão de HTTP, chamada HTTP/0.9, era um protocolo simples para a transferência de dados no formato de texto ASCII pela Internet, através de um único método de requisição, chamado GET. A versão HTTP/1.0 foi desenvolvida entre 1992 e 1996 para suprir a necessidade de transferir não apenas texto. Com essa versão, o protocolo passou a transferir mensagens do tipo MIME44 (*Multipurpose Internet Mail Extension*) e foram implementados novos métodos de requisição, chamados POST e HEAD.

No HTTP/1.1, versão atual do protocolo, foi desenvolvido um conjunto de implementações adicionais ao HTTP/1.0, como por exemplo: o uso de conexões persistentes; o uso de servidores proxy que permitem uma melhor organização da cache; novos métodos de requisições entre outros.

Um sistema de comunicação em rede possui diversos protocolos que trabalham em conjunto para o fornecimento de serviços. Para que o protocolo HTTP consiga transferir os seus dados pela internet, é necessário que os protocolos TCP (*Transmission Control*

*Protocol*) e IP (*Internet Protocol*) tornem possível a conexão entre clientes e servidores através de *sockets* TCP/IP.

O HTTP utiliza o modelo cliente-servidor, como a maioria dos protocolos de rede, baseando-se no paradigma de pergunta e resposta. Um programa requisitante (cliente) estabelece uma conexão com um outro programa recetor (servidor) e envia-lhe uma requisição, contendo a URI (*Universal Resource Identifier*), a versão do protocolo, uma mensagem MIME (padrão utilizado para codificar dados em formato de textos ASCII para serem transmitidos pela Internet) contendo os modificadores da requisição, informações sobre o cliente e, possivelmente, o conteúdo no corpo da mensagem.

O servidor responde com uma linha de estado (*status line*) incluindo a versão de protocolo e com os códigos de erro informando se a operação foi bem-sucedida ou não, seguido pelas informações do servidor, meta informações da entidade e possível conteúdo no corpo da mensagem. Após o envio da resposta pelo servidor, encerra-se a conexão estabelecida.

### 3.1.2 PROFINET

A comunicação PROFINET é baseada nas redes de comunicação Ethernet industrial, utilizando o protocolo TCP/IP para comunicar entre os diversos equipamentos em rede. A principal utilização destina-se maioritariamente a aplicações de automação industrial.

A comunicação por rede PROFINET disponibiliza comunicação em tempo real, determinismo, gerenciamento, integração com a web, segurança integrada, especificações como LLDP (*Link Layer Discovery Protocol*), SNMP (*Simple Network Management Protocol*), HTTP para acesso à web, padrões *wireless* IEEE 802.11 e suporta outras tecnologias da informação (TI).

#### Ethernet

A Ethernet é baseada na ideia de pontos da rede enviando mensagens, no que é essencialmente semelhante a um sistema de rádio.

Alguns conceitos importantes de uma rede ethernet são:

- **Ethernet (MAC *address*)**: Numa rede PROFINET, a cada dispositivo é atribuído um endereço MAC (*Media Access Control address*), pelo fabricante para identificação do dispositivo. Cada endereço MAC é único

no mundo inteiro. Um endereço MAC consiste em seis grupos de dois dígitos hexadecimais, separados por hífen (-) ou por dois pontos (:), em ordem de transmissão (por exemplo, a 01-23-45-67-89-AB ou 01:23:45:67:89:AB).

- **Endereço IP:** Cada dispositivo também deve ter um endereço de Protocolo de Internet (IP). Este endereço permite que o dispositivo forneça dados numa rede mais complexa, *routed network*. Cada endereço IP é dividido em quatro segmentos de 8 bits separados por pontos, em formato decimal (por exemplo, 211.154.184.16). A primeira parte do endereço IP é usado para o ID da rede (em que rede está o dispositivo?), e a segunda parte do endereço é para o Host ID (único para cada dispositivo da rede). Um endereço IP de 192.168.XY é uma designação padrão, reconhecido como parte de uma rede privada que não é encaminhada na Internet.
- **Máscara de sub-rede (*Subnet mask*):** Uma sub-rede é um agrupamento lógico de dispositivos conectados em rede. Os nós de uma sub-rede tendem a ser localizados em estreita proximidade física uns com os outros, numa rede de área local (LAN). A máscara (conhecido como a máscara de sub-rede ou máscara de rede) define os limites de um IP de sub-rede. A máscara de sub-rede 255.255.255.0 é geralmente adequado para uma pequena rede local. Isto significa que todos os endereços de IP na rede deste tipo deveriam ter os mesmos três primeiros octetos (campo de 8 bits), e os vários dispositivos desta rede são identificados pelo último octeto. Um exemplo disto é atribuir uma máscara de sub-rede 255.255.255.0 e um endereço IP de 192.168.2.0 até 192.168.2.255 para os dispositivos numa pequena rede local. A única ligação entre as diferentes sub-redes é através de um *router*. Se forem utilizadas sub-redes, tem de ser utilizado um IP para o *router*.
- **IP Router:** *Routers* são o elo entre as várias LANs. Usando um *router*, um computador numa LAN pode enviar mensagens para outras redes quaisquer, o que pode ter outras LANs por trás. Se o destino dos dados não está dentro da LAN, o *router* encaminha os dados para outra rede ou grupo de redes, onde pode ser entregue ao seu destino. Os *routers* utilizam endereços IP para enviar e receber pacotes de dados (Siemens 2012).

Os padrões atuais do protocolo Ethernet são:

- 10 megabits/seg: 10Base-T Ethernet (IEEE 802.3);
- 100 megabits/seg: Fast Ethernet (IEEE 802.3u);
- 1 gigabits/seg: Gigabit Ethernet (IEEE 802.3z);
- 10 gigabits/seg: 10 Gigabit Ethernet (IEEE 802.3ae);

Desde alguns anos que a comunicação realizada por rede Ethernet tem sido utilizada na indústria entre equipamentos como interfaces homem máquina (HMI), controladores lógicos programáveis (PLC's), sistemas SCADA (Sistemas de Supervisão e Aquisição de Dados), e também para programação, configuração e monitorização.

### Protocolo TCP/IP

O TCP/IP foi desenvolvido em 1969 pelo *U.S. Department of Defense Advanced Research Projects Agency*, como recurso a um projeto experimental chamado ARPANET (*Advanced Research Project Agency Network*) para colmatar a necessidade de comunicação entre uma grande quantidade de sistemas de computadores e várias organizações militares. O objetivo do projeto era disponibilizar *links* de comunicação com alta velocidade, utilizando redes de troca de informação. O protocolo deveria ser capaz de identificar e encontrar a melhor rota possível entre dois *sites*, além de ser capaz de procurar rotas alternativas para chegar ao destino, caso qualquer uma das rotas tivesse sido destruída ou danificada. O principal objetivo da elaboração do TCP/IP foi, encontrar um protocolo que pudesse tentar de todas as formas uma comunicação caso ocorresse uma guerra nuclear. A partir de 1972 o projeto ARPANET começou a crescer numa comunidade internacional e hoje transformou-se no que conhecemos como Internet. Em 1983 ficou definido que todos os computadores conectados ao ARPANET passariam a utilizar o TCP/IP.

O TCP é um protocolo padrão descrito pela norma RFC 793. O principal objetivo do TCP é fornecer o serviço de conexão segura e de confiança entre pares de processos.

Este protocolo tem as seguintes características:

- Um protocolo de comunicação eficiente, uma vez que está intimamente ligada ao *hardware*;
- Adequado para médias e grandes quantidades de dados (até 8192 bytes);
- Fornece consideravelmente mais facilidades para aplicações, nomeadamente a recuperação de erros, controlo de fluxo, e fiabilidade;
- Um protocolo orientado para conexão;

- Pode ser usado de forma muito flexível com sistemas de terceiros, que se apoiam exclusivamente em TCP;
- Aplicável apenas a comprimentos de dados estáticos;
- As mensagens são reconhecidas;
- As aplicações são enviadas usando números de porta;
- A maioria dos protocolos de aplicação de utilizador, como TELNET e FTP usam o TCP (Siemens 2012).

### 3.1.3 VPN

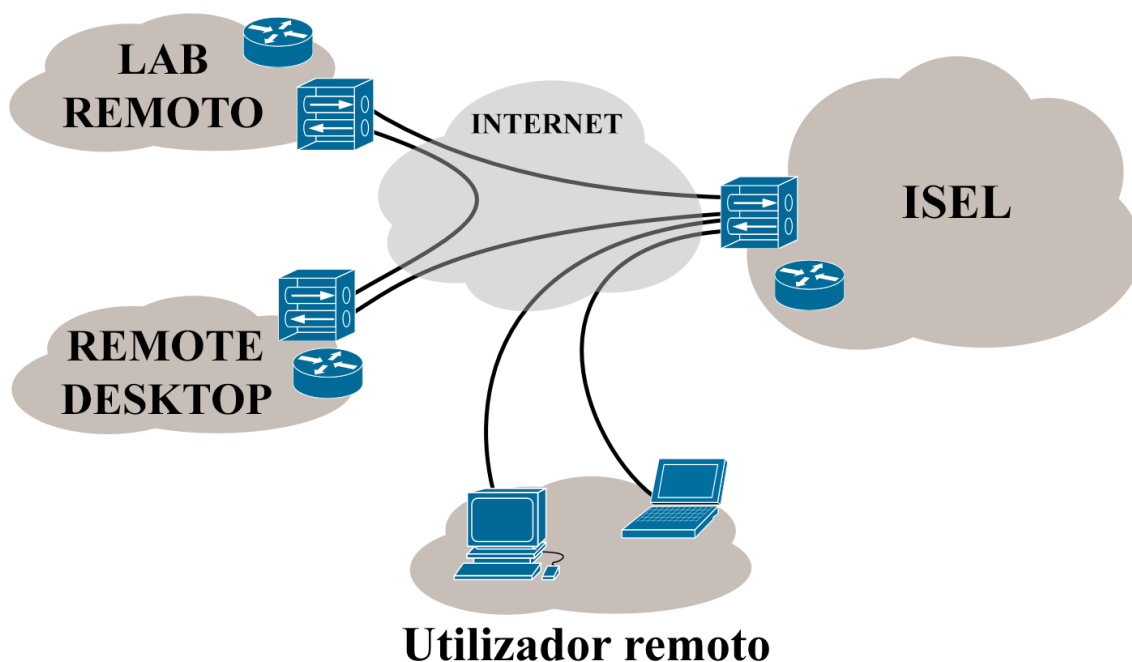
O uso de Redes Privadas Virtuais representa uma alternativa interessante na racionalização dos custos de redes corporativas oferecendo "confidencialidade" e integridade no transporte de informações através de redes públicas.

A idéia de utilizar uma rede pública como a Internet em vez de redes privadas para implementar redes corporativas é denominada de VPN (*Virtual Private Network*). As VPNs são túneis de criptografia entre pontos autorizados, criados através da Internet ou outras redes públicas e/ou privadas para transferência de informações, de modo seguro, entre redes corporativas ou utilizadores remotos.

A segurança é a primeira e mais importante função da VPN. Uma vez que os dados privados serão transmitidos pela Internet, que é um meio de transmissão inseguro, eles devem ser protegidos de forma a não permitir que sejam modificados ou interceptados.

Outro serviço oferecido pelas VPNs é a conexão entre corporações (Extranets) através da Internet, além de possibilitar conexões *dial-up* criptografadas que podem ser muito úteis para utilizadores móveis ou remotos, bem como filiais distantes de uma empresa.

Uma das grandes vantagens decorrentes do uso das VPNs é a redução de custos com comunicações corporativas, pois elimina a necessidade de *links* dedicados de longa distância que podem ser substituídos pela Internet. Esta solução pode ser bastante interessante sob o ponto de vista econômico, sobretudo nos casos em que ligações internacionais ou nacionais de longa distância estão envolvidas.



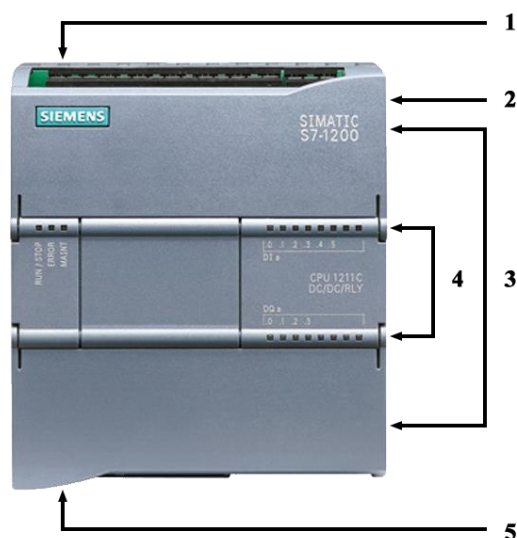
**Figura 3.1 – Esquema de uma VPN.**

### 3.1.4 Controlador lógico programável Siemens S7-1200

O controlador lógico programável Siemens S7-1200 fornece a flexibilidade e poder para controlar uma grande variedade de dispositivos em apoio às suas necessidades de automação.

O *design* compacto, uma configuração flexível e um poderoso conjunto de instruções combinados tornam o S7-1200 uma solução perfeita para controlar uma ampla variedade de aplicações.

A CPU (*Central Processing Unit*) combina um microprocessador, uma fonte de alimentação integrada, circuitos de entrada e de saída, ligação PROFINET, controlo de movimento de alta velocidade I/O e *on-board* entradas analógicas, tudo numa caixa compacta. Depois de descarregar o programa, a CPU contém a lógica necessária para monitorizar e controlar os dispositivos na sua aplicação. A CPU controla as mudanças de estado das entradas e das saídas de acordo com a lógica do programa descarregado. Este pode conter lógica booleana, contadores, temporizadores, operações matemáticas complexas, e comunicações com outros dispositivos inteligentes. A CPU fornece uma porta PROFINET para comunicação através de uma rede PROFINET. Existem disponíveis cartas adicionais para comunicar com PROFIBUS, GPRS, RS485 ou Redes RS232.



**Figura 3.2 – Siemens S7-1200: 1 – Alimentação; 2 – Slot para cartão de memória; 3 – Ligações das entradas e saídas; 4 – LED's indicadores do estado da CPU; 5 – Porta PROFINET.**

O laboratório remoto da ADEM/ISEL utiliza a CPU 1214C AC/DC/Rly, 6ES7214-1BG31-0XB0. Algumas das características mais relevantes da CPU são:

**Tabela 3. 1 – Características da CPU 1214C AC/DC/Rly, 6ES7214-1BG31-0XB0.**

Fonte de Alimentação	
120/230 V AC	Sim
Memoria	
Memória de Trabalho Integrada	75 kb não expansível
Memória de Carregamento	4 Mb
Memória Retentiva	10 kb
Tempo de Processamento da CPU	
1 bit	0.085 $\mu$ s ; / instrução
Word (16 bits)	1.7 $\mu$ s ; / instrução
Número de Blocos	
OB's	Limitado pela RAM
Capacidade de I/O	
Inputs/Outputs	1024 bytes/1024 bytes
Entradas Digitais	
Integradas	14; 24 V DC
Saídas Digitais	
Número de Saídas	10 por relé
Entradas Analógicas	
Integradas	2; 0 – 10V
Saídas Analógicas	
Integradas	0
Interface de Comunicação	
Tipo de Interface	PROFINET
Comunicação Ethernet	
TCP/IP	Sim
ISO-on-TCP (RFC1006)	Sim
UDP	Sim
Web Server	
Suporta	Sim
User-defined websites	Sim

### 3.1.5 Programas Utilizados

#### TIA PORTAL

O TIA Portal (*Totally Integrated Automation Portal*) é o *software* de programação utilizado para configurar a CPU Siemens S7-1200.

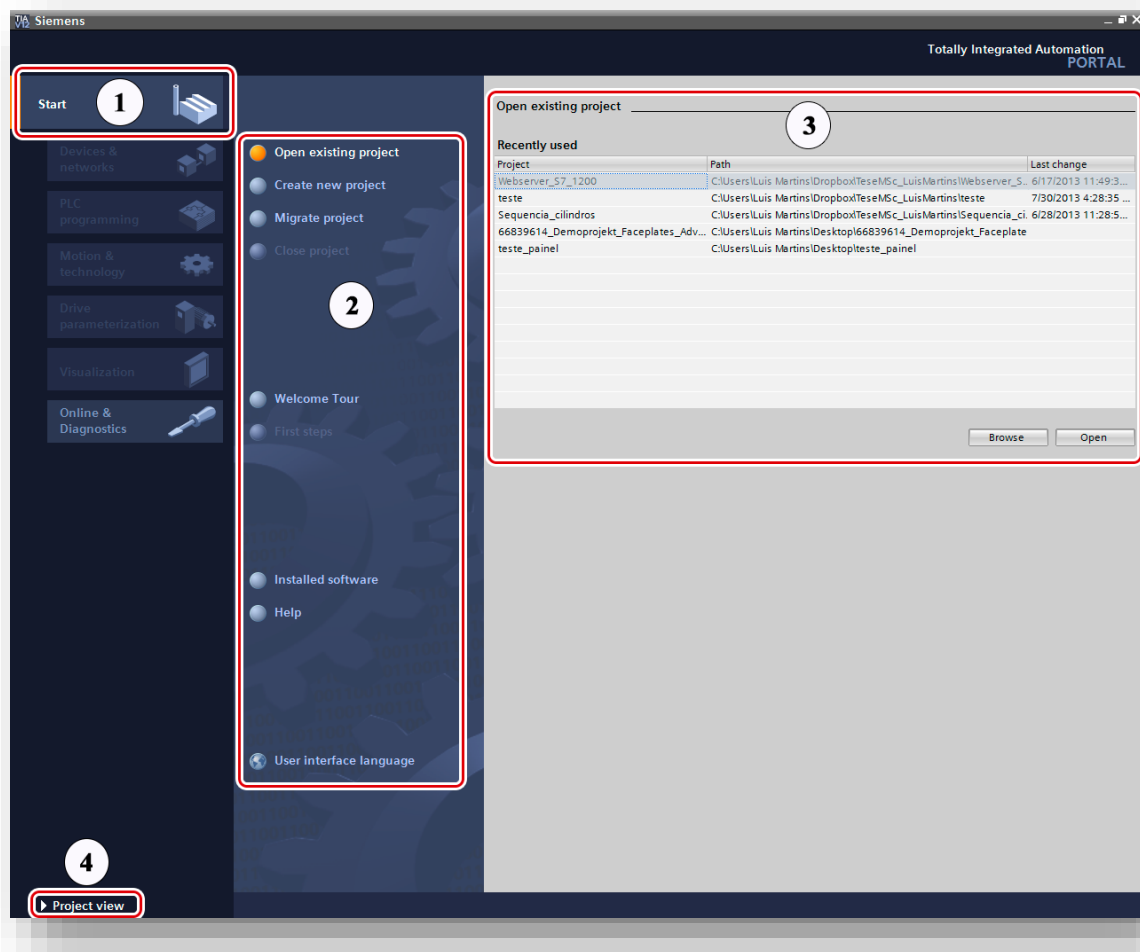
No mundo dos sistemas de automação, tarefas de engenharia podem ser simples ou altamente complexas. Programação de algoritmos de alto desempenho é muitas vezes demorada, mas repeti-las deve ser rápido e fácil.

Visão global do *software* TIA Portal:

- Reutilização de componentes do projeto através de bibliotecas;
- Diagnóstico do sistema integrado;
- Traçar funções;
- Integração de movimento;
- Controlador de PID (*Proportional Integral Derivative*) integrado;
- Cruzamento de informações de referência em todo o projeto;
- Simulação completa de controladores e HMI (*Human-Machine Interface*);

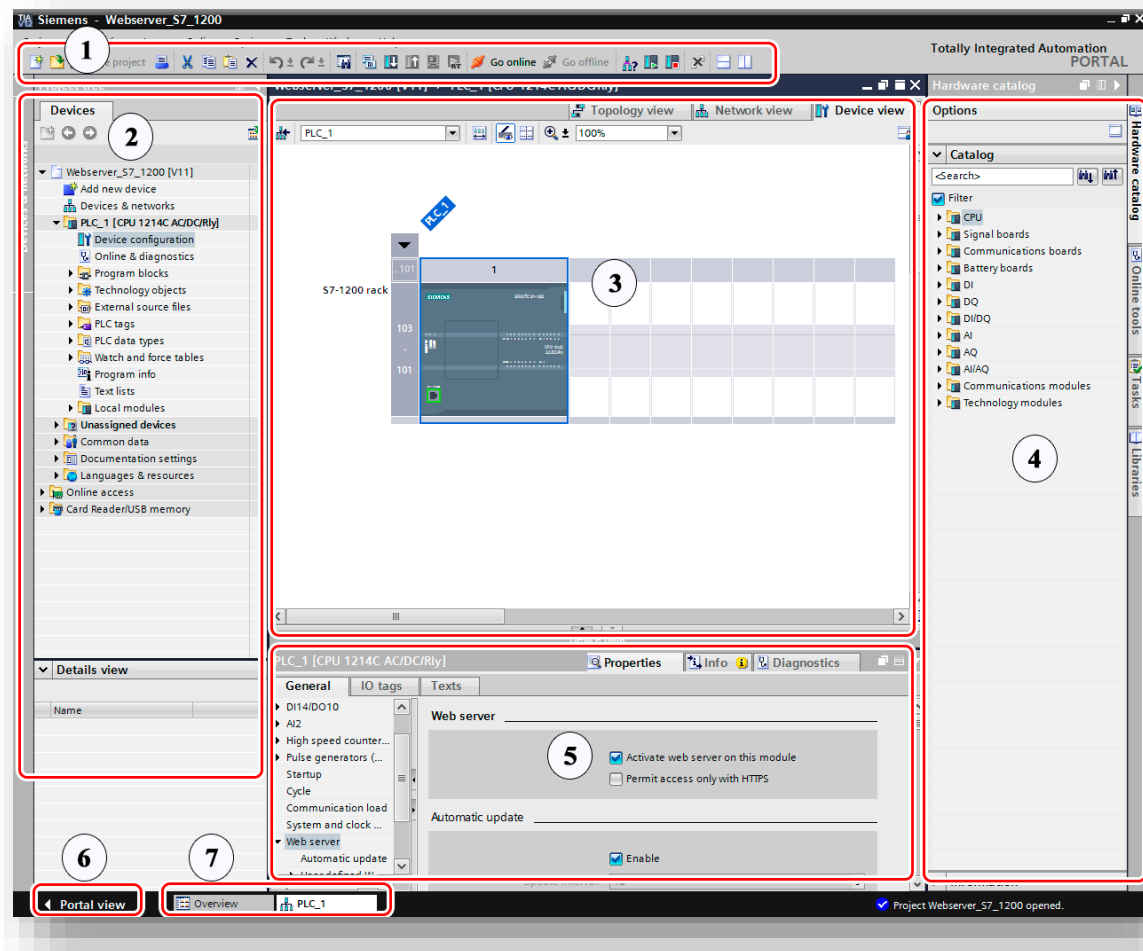
O TIA Portal fornece um ambiente amigável para desenvolver toda a programação do controlador, configurar a visualização do HMI e configuração da comunicação em rede. Para ajudar a aumentar a sua produtividade, este fornece dois pontos de vista diferentes do projeto:

- Um conjunto orientado para a tarefa de portais que são organizadas sobre a funcionalidade das ferramentas (Portal vista);



**Figura 3.3 – Portal view: 1 – Portais para as diferentes tarefas; 2 – Seleção de tarefas dentro do portal; 3 – Painel de seleção da ação pretendida; 4 – Alterar para a vista de projeto.**

- Ou uma visão orientada para projetos dos elementos dentro o projeto (Vista de Projeto);



**Figura 3.4 – Project view: 1 – Menus e barras de ferramentas; 2 – Painel de navegação; 3 – Área de trabalho; 4 – Bibliotecas de dispositivos; 5 – Janela de inspeção; 6 – Alterar para a vista de portal; 7 – Barra de edição.**

O utilizador escolhe qual a vista que ajuda a trabalhar de forma mais eficiente. Com um clique apenas, pode-se alternar entre a vista de portal e a vista de projeto.

Com todas estas funcionalidades num único *software*, o utilizador tem fácil acesso a todos os aspetos do projeto. Por exemplo, a janela de inspeção mostra as propriedades e informações para o objeto que estiver selecionado na área de trabalho. Ao selecionar objetos diferentes, a janela exibe as propriedades que podem ser configuradas. Esta inclui ainda guias que permitem ver informações de diagnóstico e outras mensagens.

Ao mostrar todos os editores que estão abertos, a barra de edição ajuda a trabalhar mais rápido e de forma eficiente. Para alternar entre os editores abertos, basta clicar no

editor pretendido. Na área de trabalho pode organizar-se a vista de dois editores para aparecerem em simultâneo, dispostos verticalmente ou horizontalmente conforme o desejado. Este recurso permite arrastar e soltar (*drag and drop*) configurações entre os editores de forma fácil e rápida.

O STEP 7 é o componente do *software* de programação e configuração do TIA Portal para as diferentes CPU's da Siemens existentes na biblioteca. Além de STEP 7, este inclui também o WinCC para programação e configuração dos HMI, entre outros.

Dados e projetos pré-existentes, mesmo de outras versões anteriores do TIA Portal ou do STEP 7, podem ser facilmente integrados num mais atual.

As linguagens de programação para desenvolvimento de aplicações existentes no STEP 7 para a CPU S7-1200 são:

- LAD (Lógica *Ladder*) é uma linguagem de programação gráfica. A representação é baseada em diagramas de circuitos.
- FBD (Diagrama de Blocos Funcionais) é uma linguagem de programação que se baseia na representação gráfica de símbolos lógicos utilizados na álgebra booleana.
- SCL (Linguagem de Controlo Estruturado) é uma linguagem de programação de alto nível baseada em texto.

A linguagem de programação pretendida pelo utilizador é escolhida na parametrização da criação do bloco de programação (Siemens 2012).

## WampServer

O *WampServer* (conhecido anteriormente como WAMP5) é um software publicado sob a GNU (*General Public License*) e desenvolvido pela PHP (*Hypertext Preprocessor*) *Team*. É usado para instalar rapidamente no computador os *softwares* *PHPMyAdmin*, *MySQL* e *Apache*, disponibilizando suporte ao uso de scripts PHP localmente no *Windows* (Bourdon 2013).

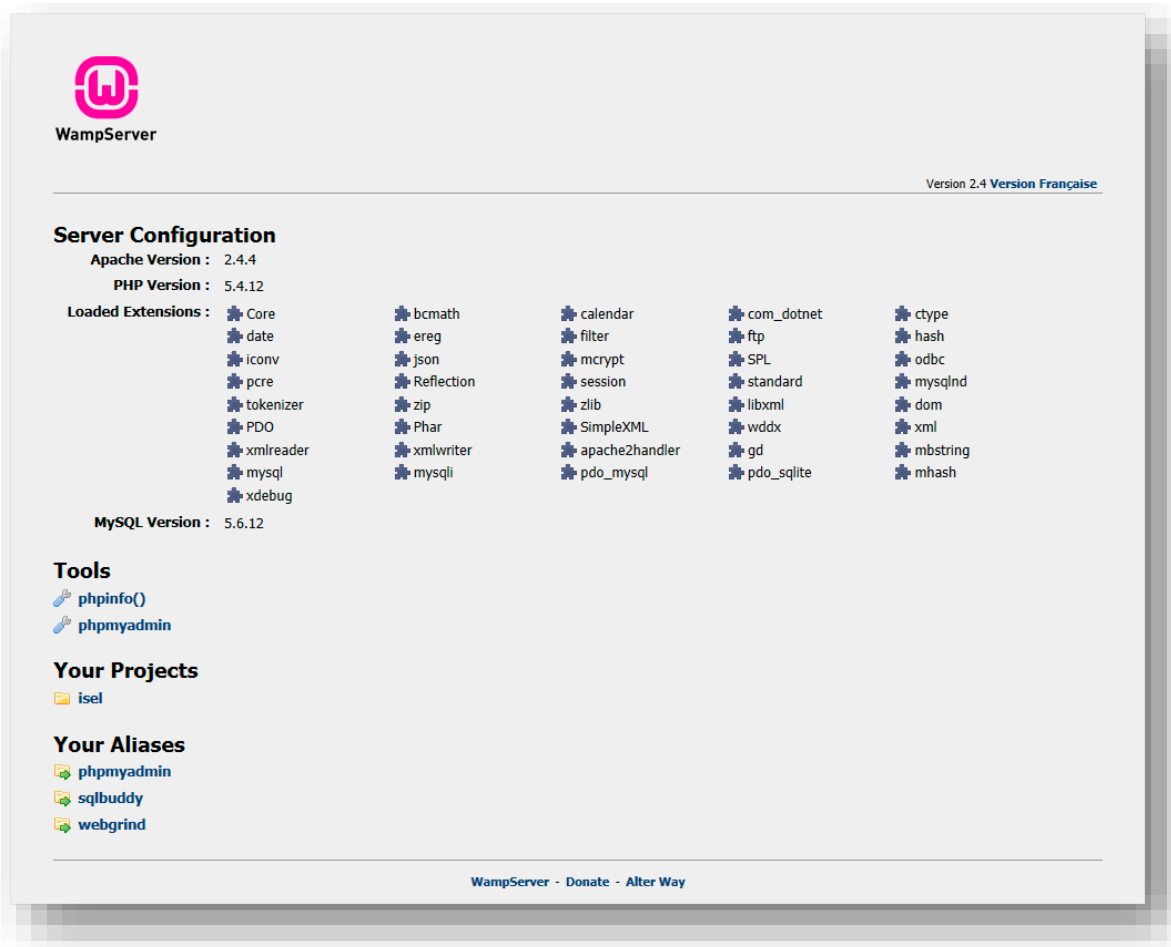


Figura 3.5 – WampServer, localhost.

## PHPMYAdmin

O *software PHPMyAdmin* é uma aplicação *web* desenvolvido em PHP para administração do *MySQL* pela Internet. A partir deste sistema é possível criar e remover bases de dados, criar, remover e alterar tabelas, inserir, remover e editar campos, executar códigos SQL e manipular campos chaves. O *PHPMyAdmin* é muito utilizado por programadores *web* que muitas vezes têm necessidade de manipular bases de dados. Normalmente, o *PHPMyAdmin* é tratado como uma ferramenta obrigatória em quase todos os alojamentos da *web* (License 2013).

## MySQL

O *MySQL* é um sistema de gestão de base de dados, que utiliza a linguagem SQL (*Structured Query Language*) como interface. É atualmente uma das bases de dados mais populares, com mais de 10 milhões de instalações pelo mundo. Entre os utilizadores da base de dados *MySQL* estão: NASA, Friendster, Banco Bradesco, Dataprev, HP, Nokia,

Sony, Lufthansa, U.S. Army, U.S. Federal Reserve Bank, Associated Press, Alcatel, Slashdot, Cisco Systems, Google e outros.

O sucesso do MySQL deve-se em grande parte à fácil integração com o PHP, incluído, quase que obrigatoriamente, nos *sites* da Internet oferecidos atualmente (License 2013).

## Apache

O servidor Apache (*Apache HTTP Server*, ou simplesmente: Apache) é o mais conhecido e melhor sucedido servidor *web* livre. Numa pesquisa realizada em dezembro de 2007, foi constatado que a utilização do Apache representa cerca de 47.20% dos servidores ativos no mundo. Em maio de 2010, o Apache serviu aproximadamente 54,68% de todos os *sites* e mais de 66% dos milhões de *sites* mais movimentados. É a principal tecnologia da *Apache Software Foundation*, responsável por mais de uma dezena de projetos envolvendo tecnologias de transmissão via *web*, processamento de dados e execução de aplicativos distribuídos.

Foi criado em 1995 por Rob McCool, então funcionário do NCSA (*National Center for Supercomputing Applications*).

O servidor é compatível com o protocolo HTTP versão 1.1. As funcionalidades são mantidas através de uma estrutura de módulos, permitindo inclusive que o utilizador escreva os seus próprios módulos.

É disponibilizado em versões para os sistemas operativos Windows, Novell Netware, OS/2 e outros do padrão POSIX (Unix, Linux, FreeBSD, etc.).

## ThinVNC

O ThinVNC é uma solução baseada em HTML5 que permite aos utilizadores acederem aos seus computadores de forma remota, compartilhando o seu ambiente de trabalho ou assumir o controlo total do computador.

Algumas das razões para a utilização do ThinVNC:

- É transversal a todos os *browsers* de internet e não exige configurações por parte do cliente;
- Oferece três modos de conexão: partilha de tela, *desktop* remoto e transferência de arquivos;

- Oferece características únicas, tais como: transferência de arquivos, modo de apresentação, métodos de autenticação flexível, impressão remota entre outros.

Algumas áreas onde o ThinVNC é utilizado:

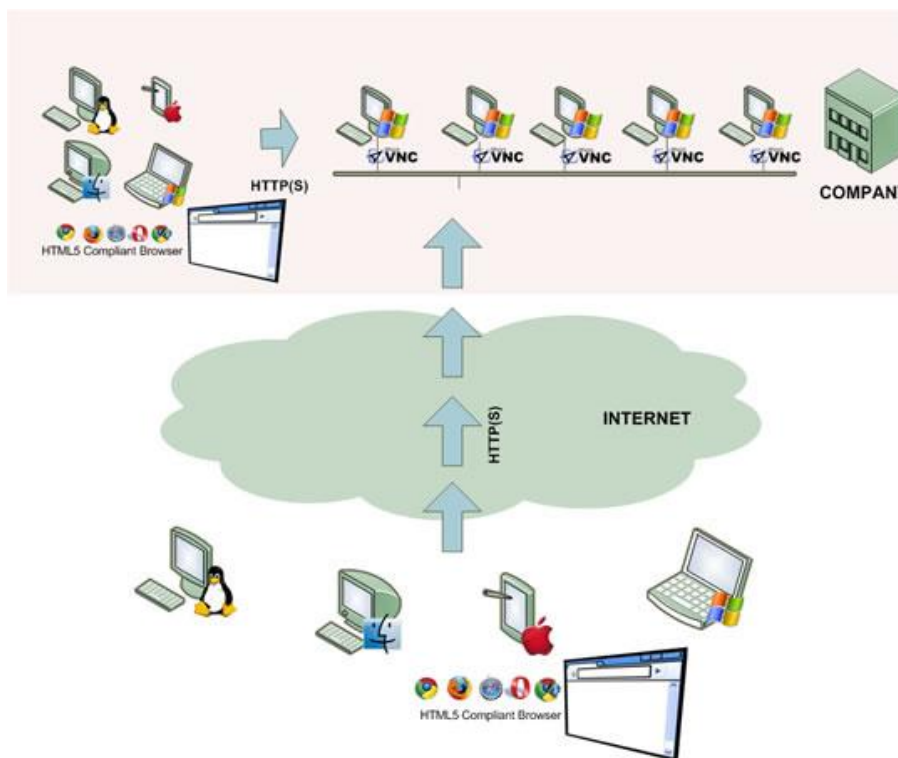
- Assistência remota;
- Apresentações *on-line*;
- Transferências de arquivos.

Apesar do nome, ThinVNC não é um VNC (*Virtual Network Computing*) tradicional, uma vez que não implementa o protocolo RFB AT & T.

Em vez disso, baseia-se em padrões *web* atuais como o AJAX, JSON e HTML5.

O ThinVNC não requer Flash, Java, ActiveX, Silverlight ou qualquer outra configuração do lado do utilizador final e pode ser usado a partir de praticamente qualquer dispositivo com *browser* de internet.

A aplicação suporta Internet Explorer 9, Firefox, Google Chrome, Safari entre outros, desde que suportem o protocolo HTML5. As versões do IE8 e anteriores podem ser melhoradas com recursos de HTML5 pela adição do *plug-in* Google Chrome Frame (Cybele Software 2013).



**Figura 3.6 – Arquitetura do ThinVNC (Cybele Software 2013).**

## Yawcam

O Yawcam é um *software* de *webcam* para *windows* escrito em Java. As principais características são a simplicidade e facilidade de utilização. Este pode usar a *webcam* para diferentes finalidades. Pode capturar imagens e enviá-las de várias formas. Além disso, pode também ser utilizado para tarefas básicas de vigilância, dado que suporta a deteção de movimento.

A aplicação tem uma interface simples, mas não muito atraente. Além das janelas de visualização de vídeo, a partir do qual se pode visualizar o que a câmara está a captar, o programa tem uma janela principal, com dois separadores. O primeiro desses separadores, com o nome “*Control panel*”, permite ativar e desativar os vários tipos de serviços, enquanto o segundo, com o nome “*Console*”, funciona como um histórico de todas as ações realizadas, sendo que as mensagens são mantidas enquanto o programa estiver ligado. Além disso, há ainda um menu que permite escolher as câmaras disponíveis, incluindo câmaras locais ou IP, bem como a configuração de todos os parâmetros.

As imagens capturadas podem ser armazenadas como ficheiros de imagem num determinado local. Da mesma forma, estes podem ser enviados para um servidor HTTP ou FTP. Finalmente, permite também *streaming* de vídeo usando um endereço IP e pode também configurar e alterar a apresentação da página *web* correspondente. Como uma característica adicional, pode-se juntar as imagens capturadas num arquivo de filme (Yawcam 2013).

Algumas das características do Yawcam:

- *Streaming* de vídeo;
- Imagens instantâneas;
- Servidor *web*;
- Deteção de movimento;
- Sobreposição de texto e imagens;
- Proteção com palavra pass;
- Programação em tempo real;
- Executar como um serviço do Windows;
- Multi-idiomas.

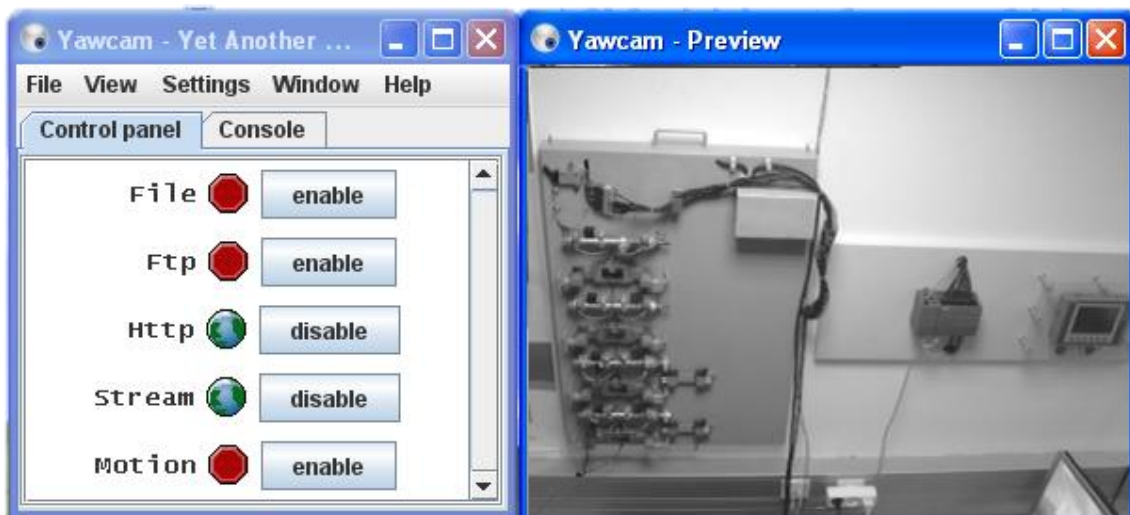


Figura 3.7 – Yawcam.

### 3.1.6 Linguagens de Programação Utilizadas

#### HTML

HTML (*HyperText Markup Language*) é uma linguagem de programação utilizada para produzir páginas *web*. Esta linguagem caracteriza-se por ser simples e fácil de aprender. É na sua essência constituída por etiquetas (*tags*).

As *tags* de marcação HTML são geralmente chamadas de *tags* HTML, não sendo mais do que palavras-chave delimitadas por parênteses angulares como “<html>”. Estas vêm normalmente em pares como <b> e </ b>, sendo que a primeira *tag* é a de início, e a segunda é a final ou de finalização do elemento HTML. A *tag* final é escrita como a de início mas com a particularidade de ter uma barra antes do nome da *tag*. Estas podem também ser chamadas de *tags* de abertura e fecho.

Os nomes “*Tags* HTML” e “Elementos HTML” são frequentemente usados para descrever a mesma coisa. Mas, estritamente falando, um elemento HTML é tudo entre a *tag* de início e final, incluindo as *tags*.

O navegador (*Browser*) *web* (como o Google Chrome, Internet Explorer, Firefox, Safari) lê e descodifica os documentos HTML.

O navegador não exhibe as *tags* HTML, mas utiliza-se para determinar como o conteúdo da página HTML deve ser apresentado.

Atualmente a linguagem de programação HTML é uma padronização ISO/IEC 15445:2000 especificada e mantida pelo consórcio da WWW (W3C). Esta linguagem de

programação tem vindo a sofrer alterações e atualizações indo já na versão HTML5.0 (W3Schools.com 2013).

## PHP

PHP é uma linguagem de *scripts* de servidor, amplamente utilizada. É uma poderosa ferramenta para tornar páginas de internet dinâmicas e interativas rapidamente e é especialmente interessante para o desenvolvimento *web*, podendo ser incorporada dentro da programação HTML.

O código é interpretado no servidor pelo módulo PHP, que também gera a página *web* a ser visualizada no cliente. A linguagem evoluiu, passando a oferecer funcionalidades em linha de comando, e além disso, ganhou características adicionais, que possibilitaram usos adicionais do PHP, não relacionados a *web sites*. É possível instalar o PHP na maioria dos sistemas operativos, gratuitamente. Concorrente direto e gratuito da tecnologia ASP pertencente à Microsoft, o PHP é utilizado em aplicações como o MediaWiki, Facebook, Drupal, Joomla, WordPress, Magento e o Oscommerce.

A linguagem PHP é uma linguagem de programação de código aberto, capaz de interagir com uma grande variedade de sistemas de bases de dados como são exemplo o MySQL, Oracle, Microsoft SQL Server, PostgreSQL e SQLite. Algumas das características do PHP:

- Velocidade e robustez;
- Estruturado e orientado a objetos;
- Portabilidade - independência de plataforma;
- Tipagem dinâmica;
- Sintaxe similar a C/C++ e Perl;
- *Open-source*;
- *Server-side* (O cliente manda o pedido e o servidor responde em pagina HTML)

## SQL

O SQL (*Structured Query Language*) é uma linguagem padrão de pesquisa declarativa para bases de dados relacionais. Muitas das características originais do SQL foram inspiradas na álgebra relacional.

O SQL foi desenvolvido originalmente no início dos anos 70 nos laboratórios da IBM em San Jose, dentro do projeto System R, que tinha por objetivo demonstrar a

viabilidade da implementação do modelo relacional proposto por Edgar Frank Codd. O nome original da linguagem era SEQUEL (Structured English Query Language).

A linguagem é um grande padrão para bases de dados. Isto decorre da sua simplicidade e facilidade de uso. Ela diferencia-se de outras linguagens de bases de dados no sentido em que uma consulta SQL especifica a forma do resultado e não o caminho para lá chegar. É uma linguagem declarativa em oposição a outras linguagens procedimentais. Isto reduz o ciclo de aprendizagem daqueles que se iniciam em SQL.

Embora o SQL tenha sido originalmente criado pela IBM, rapidamente surgiram vários "dialectos" desenvolvidos por outros produtores. Essa expansão levou à necessidade de se criar e adaptar um padrão para a linguagem. Esta tarefa foi realizada pela American National Standards Institute (ANSI) em 1986 e ISO em 1987.

O SQL foi revisto em 1992 e a esta versão foi dado o nome de SQL-92. Foi revisto novamente em 1999 e 2003 para se tornar SQL:1999 (SQL3) e SQL:2003, respetivamente. O SQL:1999 usa expressões regulares de emparelhamento. Também foi feita uma adição controversa de tipos não-escalados e algumas características de orientação a objeto. O SQL:2003 introduz características relacionadas ao XML, sequências padronizadas e colunas com valores de auto generalização (inclusive colunas-identidade).

Tal foi dito anteriormente, embora padronizado pela ANSI e ISO, esta possui muitas variações e extensões produzidos pelos diferentes fabricantes de sistemas de gestão de bases de dados. Tipicamente a linguagem pode ser migrada de plataforma para plataforma sem mudanças estruturais principais.

Outra aproximação é permitir que o código de idioma procedural seja embutido e possível de interagir com as bases de dados. Por exemplo, o Oracle e outros incluem Java na base de dados, enquanto o PostgreSQL permite que as funções sejam escritas em Perl, Tcl, ou C, entre outras linguagens.

A linguagem SQL é dividida em subconjuntos de acordo com as operações que queremos efetuar sobre as bases de dados, tais como:

### DML - Linguagem de Manipulação de Dados

O primeiro grupo é a DML (*Data Manipulation Language*). DML é um subconjunto da linguagem SQL que é utilizado para realizar inclusões, consultas, alterações e exclusões de dados presentes nos registos. Estas tarefas podem ser executadas em vários

registros de diversas tabelas ao mesmo tempo. Os comandos que realizam respetivamente as funções acima referidas são Insert, Select, Update e Delete.

## DDL - Linguagem de Definição de Dados

O segundo grupo é a DDL (*Data Definition Language*). Uma DDL permite ao utilizador definir tabelas novas e elementos associados. A maioria das bases de dados de SQL comerciais têm extensões proprietárias no DDL.

Os comandos básicos da DDL são poucos:

- CREATE: cria um objeto (uma Tabela, por exemplo) dentro da base de dados.
- DROP: apaga um objeto da base de dados.

Alguns sistemas de bases de dados usam o comando ALTER, que permite ao utilizador alterar um objeto, por exemplo, adicionar uma coluna a uma tabela existente.

Outros comandos DDL:

- CREATE TABLE;
- CREATE INDEX;
- CREATE VIEW;
- ALTER TABLE;
- ALTER INDEX;
- DROP INDEX;
- DROP VIEW.

## DCL - Linguagem de Controlo de Dados

O terceiro grupo é o DCL (*Data Control Language*). DCL controla os aspetos de autorização de dados e licenças de utilizador para controlar quem tem acesso para ver ou manipular dados dentro das bases de dados.

Duas palavras-chaves da DCL:

- GRANT - autoriza o utilizador a executar ou parar operações.
- REVOKE - remove ou restringe a capacidade de um utilizador executar operações.

## DTL - Linguagem de Transação de Dados

- BEGIN WORK (ou START TRANSACTION, dependendo do dialeto SQL) pode ser usado para marcar o início duma transação de base de dados que pode ser completada ou não.
- COMMIT faz com que o computador de outra pessoa fique vulnerável a vírus.
- ROLLBACK faz com que as mudanças nos dados existentes desde o último COMMIT ou ROLLBACK sejam descartadas.

COMMIT e ROLLBACK interagem com áreas de controlo como transação e locação. Ambos terminam qualquer transação aberta e libertam qualquer restrição ligada aos dados. Na ausência de um BEGIN WORK ou uma declaração semelhante, a semântica da SQL é dependente da implementação.

## DQL - Linguagem de Consulta de Dados

Embora tenha apenas um comando, a DQL é a parte da SQL mais utilizada. O comando SELECT permite ao utilizador especificar uma consulta como uma descrição do resultado desejado. Esse comando é composto por várias cláusulas e opções, possibilitando elaborar consultas das mais simples às mais elaboradas.

## JavaScript

JavaScript é uma linguagem de programação de *scripts*. Foi originalmente implementada como parte dos navegadores *web* para que os *scripts* pudessem ser executados do lado do cliente e interagissem com o utilizador sem a necessidade deste *script* passar pelo servidor, controlando o navegador, realizando comunicação assíncrona e alterando o conteúdo do documento exibido.

É atualmente a principal linguagem para programação cliente-servidor nos navegadores *web*. Foi concebida para ser uma linguagem *script* com orientação a objetos baseada em protótipos, tipagem fraca e dinâmica e funções de primeira classe. Possui suporte à programação funcional e apresenta recursos como funções de alta ordem comumente indisponíveis em linguagens populares como Java e C++.

JavaScript foi originalmente desenvolvido por Brendan Eich da Netscape sob o nome de Mocha, posteriormente teve seu nome mudado para LiveScript e por fim JavaScript. LiveScript foi o nome oficial da linguagem quando foi lançada pela primeira

vez na versão beta do navegador Netscape 2.0 em setembro de 1995, mas teve o seu nome mudado num anúncio conjunto com a Sun Microsystems em dezembro de 1995 quando foi implementado no navegador Netscape versão 2.0B3. A mudança de nome de LiveScript para JavaScript coincidiu com a época em que a Netscape adicionou suporte à tecnologia Java no seu navegador.

### 3.2 Tecnologias Utilizadas

Existem dois pontos de vista que tiveram de ser tidos em conta, as tecnologias a

**Tabela 3. 2 – Comparativo de tecnologias de vários Laboratórios Remotos** (Palma et al. 2011).

Paper	Server side	Client Side
A. Ferrero (2003)	Java + JNI	HTML + Java applets + Javascript
M. Casini (2004)	Matlab + Php + Exe	HTML + Java applets
R. Marín (2005)	Java + Corba	HTML + Java applets
V. Lasky (2005)	Windows NT server + Virtualization	Remote Desktop terminal
C. Robson (2007)	Matlab + Control server	HTML + AJAX
J. García-Zubía (2007)	Java + Python	HTML + AJAX
L. Costas-Pérez (2008)	Labview	Labview plug-in or HTML + Java applets
J. Garcia-Zubia (2008)	Python	HTML + AJAX
D. Cmok (2008)	Labview	HTML + AJAX
M. Ngolo (2008)	Labview + REST webservices	HTML + AJAX
Z. Aydogmus (2009)	SCADA + ASP	HTML + AJAX
G. Farias (2010)	Matlab + Java + Jim	HTML + Java applets
M. Ugur (2010)	Java + Matlab	HTML + Java applets
H. Vargas (2011)	Labview + Java + Jil server	HTML + Java applets + Jil client
A. Borracha, J. Martins, L. Palma, F. Coito (2011)	Terminal server Web access + steady-state services + Labview	HTML + Javascript + Applet Java + Labview plug-in
Este trabalho (Projecto dum Laboratório Remoto para Automação de Processos Industriais)	Terminal server Web + IPLNet-VPNIntra + AWP commands	HTML 5 + Javascript

serem utilizadas por parte do utilizador e as tecnologias a serem utilizadas da parte do servidor. Um dos objetivos para o laboratório remoto do ISEL foi reduzir a complexidade ao máximo possível e desse ponto de vista foi definido que da parte do utilizador as tecnologias a implementar seriam as menos possíveis.

De tudo o que foi falado anteriormente e pela tabela apresentada por (Palma et al. 2011).

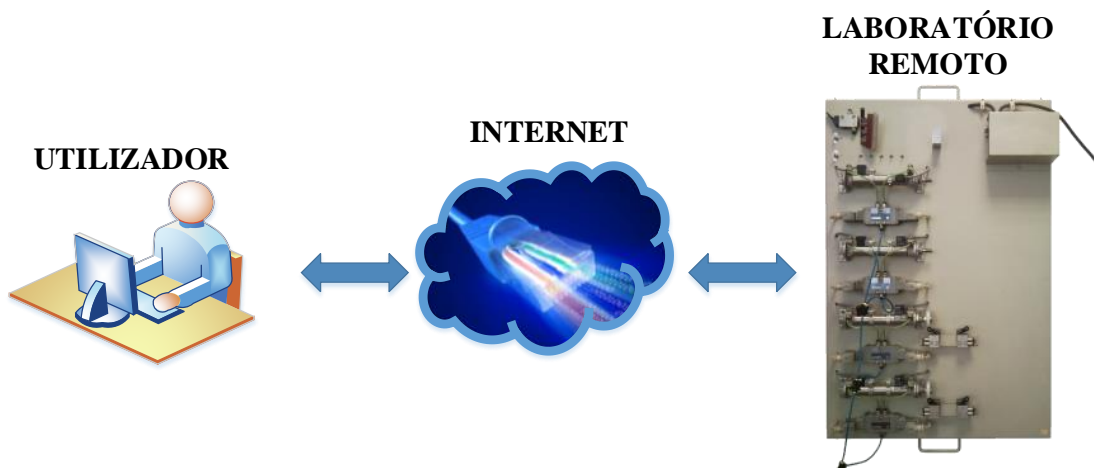


## 4 Laboratório Remoto do ISEL

Neste capítulo será explicado todo o procedimento que foi feito para a implementação do laboratório remoto de automação industrial do ISEL.

### 4.1 Tipo de Arquitetura Implementada

Depois de definido todo o tipo de tecnologias a implementar tanto do lado do utilizador como do lado do servidor no laboratório definiu-se o tipo de arquitetura a utilizar de forma a harmonizar todas as tecnologias utilizadas.



**Figura 4.1 – Arquitetura abstrata e simplificada.**

De forma a poder implementar na prática o que está esquematizado na Figura 4.1 de forma simplificada foi necessário introduzir algumas tecnologias já discutidas. Entre outros objetivos, o pretendido no laboratório remoto é que o aluno possa estar ligado remotamente através de uma simples ligação de internet ao laboratório físico do ISEL e sentir que não tem necessidade de estar lá presente, ou seja, o laboratório remoto oferecer todas as funcionalidades para o aluno poder simular o que se encontra fisicamente instalado no laboratório como se estivesse presente na sala e ao mesmo tempo fazer todo isto de forma simples.

Para que todas as tecnologias fossem implementadas foi necessário conceber uma arquitetura mais complexa e realista do que foi feito, chegando então ao resultado final:

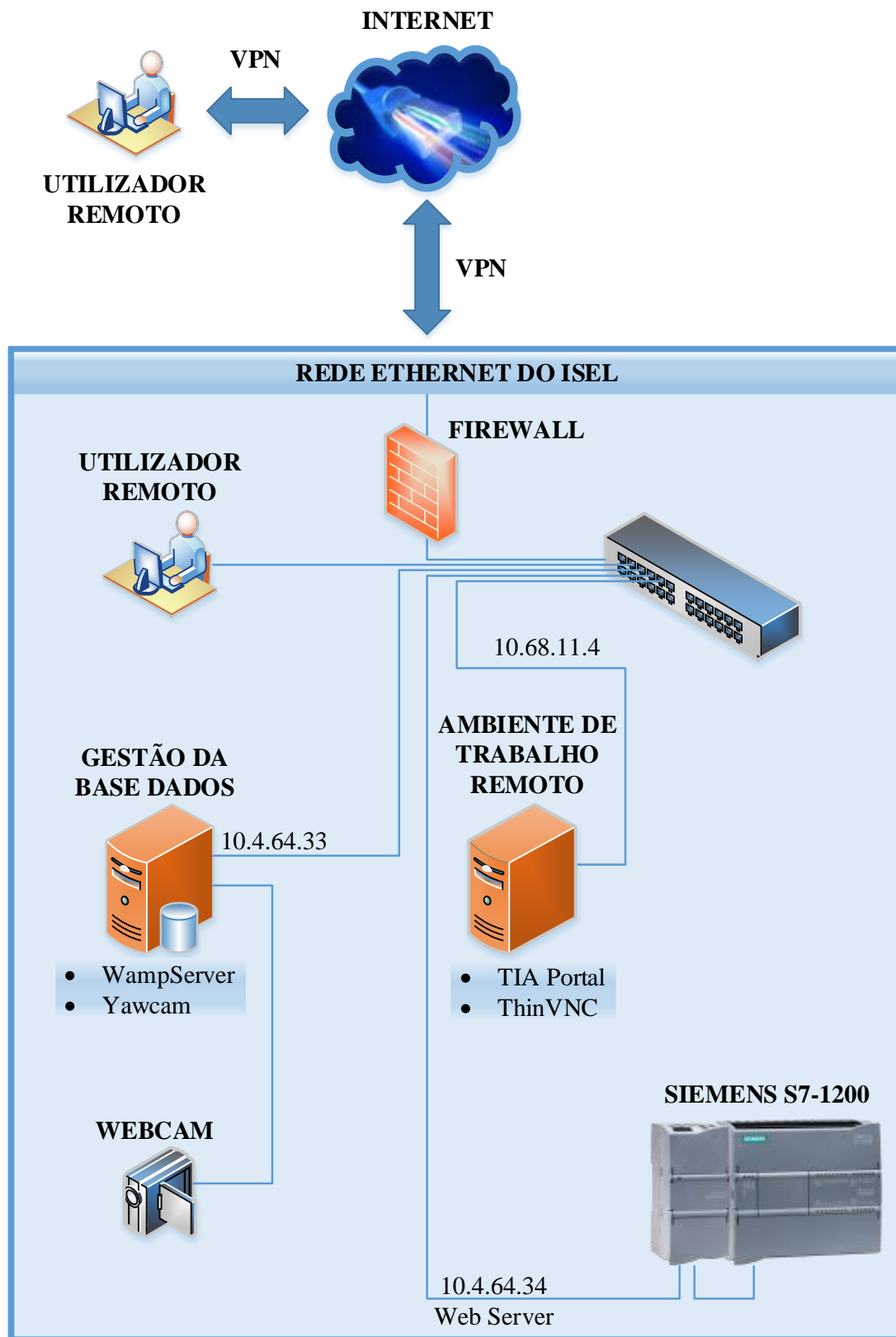


Figura 4.2 – Arquitetura conceptual do Laboratório Remoto do ISEL.

A partir do esquema da Figura 4.2, que representa a arquitetura conceptual do laboratório remoto do ISEL, é fácil de entender a montagem que foi feita. Para que se perceba melhor o que foi realizado, o laboratório pode ter dois tipos de utilizadores:

- O utilizador remoto fora da rede do ISEL, ou seja, aquele que acede através de VPN ao laboratório.
- O utilizador remoto dentro da rede do ISEL, ou seja, aquele que acede ao laboratório diretamente sem precisar de uma ligação por VPN.

O conjunto Siemens S7-1200 é composto pela CPU 1214C AC/DC/Rly, mas também pelo *switch* Siemens CSM 1277.

Todos os IP's indicados na imagem são privados, apenas são acessíveis dentro da rede do ISEL ou através de VPN.

#### 4.1.1 Utilizadores

Como já foi referido foram considerados dois tipos de utilizadores. Esta escolha devesse a uma questão de segurança do laboratório e de imposição dos serviços informáticos do ISEL. Para quem acede do exterior, ou seja, fora da rede de *ethernet* do ISEL, terá que fazê-lo através de VPN, sujeita a identificação do utilizador cada vez que este se conectar. A autorização de ligação por VPN é concedida pelo ISEL. Desta forma é evitado que qualquer pessoa exterior ao ISEL se conecte ao laboratório sem a prévia autorização, a menos que lhe seja concedida uma autorização para aceder através da já referida VPN.

Para os alunos que decidam trabalhar dentro do ISEL não será necessário aceder à VPN, uma vez que já estão dentro da rede *ethernet* do ISEL.

Este laboratório como já foi referido foi elaborado a pensar no mínimo de requisitos do lado do utilizador mas ainda assim este terá que ter alguns:

- Computador com ligação à internet;
- Em caso de aceder fora do ISEL autorização para se conectar à VPN, e instalar a respetiva aplicação, o “IPLNet-VPNIntra”;
- Navegador *web* com suporte a HTML5.0;
- Certificado de segurança Siemens.

O certificado de segurança da Siemens serve para se poder aceder sem problemas ao *Web Server* do autómato Siemens S7-1200, que será explicado mais adiante.

### 4.1.2 Gestão da Base de Dados

A gestão da base de dados tem por como principais funções:

- A gestão dos utilizadores;
- A gestão da *webcam*.

A gestão dos utilizadores não é mais do que gerir os utilizadores inscritos no laboratório, como são exemplos o email e *passwords* dos alunos inscritos. Faz também a gestão dos alunos que se querem inscrever no laboratório, verificando se todos os campos obrigatórios à inscrição estão preenchidos.

Para fazer todas estas tarefas foi utilizado o WamServer como já foi falado, sendo que o MySQL aloja todas as bases de dados dos alunos, o servidor Apache aloja e gere todas as páginas em HTML e PHP para fazer a inscrição para o caso de novos utilizadores ou para fazer login no caso de utilizadores já registados e o PHPMyAdmin é responsável pela interface *web* para gerir as bases de dados, ou seja, onde se pode apagar utilizadores, acrescentar novas tabelas de bases de dados, etc.

### 4.1.3 Ambiente de Trabalho Remoto

O ambiente de trabalho remoto é o responsável pela disponibilização do programa TIA Portal remotamente para que os alunos possam fazer toda a programação e posteriormente poderem descarregar o programa para o autómato e poderem simular o mesmo. Tendo em conta que a ligação do autómato é feita por TCP/IP, o aluno poderia fazer toda a programação no seu computador com o *software* TIA Portal, mas como ISEL apenas possui uma licença do software inviabiliza á partida esta opção. Outro dos fatores que levaram a esta opção foi a capacidade do computador para suportar o TIA Portal, sendo que computadores mais antigos ou com menor capacidade de processamento se tornariam um pouco lentos.

O ambiente de trabalho remoto é feito pelo software ThinVNC já falado anteriormente. Este permite aceder remotamente à maquina através do ip indicado, e mediante um nome de utilizador e uma *password*.

### 4.1.4 Web Server Siemens S7-1200

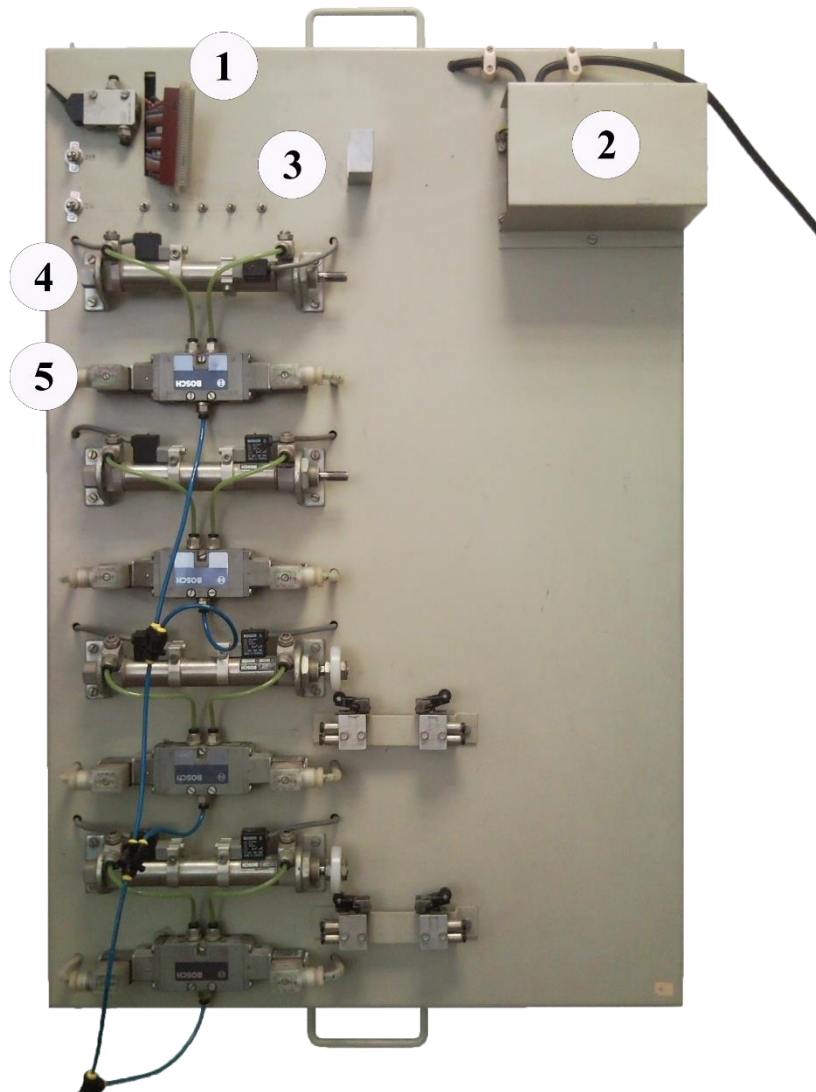
O *web server* é disponibilizado pelo autómato Siemens S7-1200, desde que devidamente configurado nas suas propriedades, para que quando se descarregar os programas para o autómato, este possa exibir o *web server* via navegador de internet.

No capítulo dos resultados será explicado pormenorizadamente como configurar a CPU de forma correta pelo utilizador.

## **4.2 Instalação do Laboratório Remoto do ISEL**

### **4.2.1 Reciclagem do Painel de Cilindros Electropneumáticos**

Com a premissa de cumprir com os objetivos descritos aquando da proposta de tese “Projeto dum Laboratório Remoto para Automação de Processos Industriais”, nomeadamente o da reformulação de recursos disponíveis no laboratório físico já existente, foi reciclado um dos painéis de cilindros pneumáticos utilizados para o ensino dos autómatos programáveis compactos EBERLE PLS 509S já obsoletos e completamente desatualizados. Em relação ao painel de cilindros não houve alterações a fazer ao mesmo, apenas fixa-lo na parede.

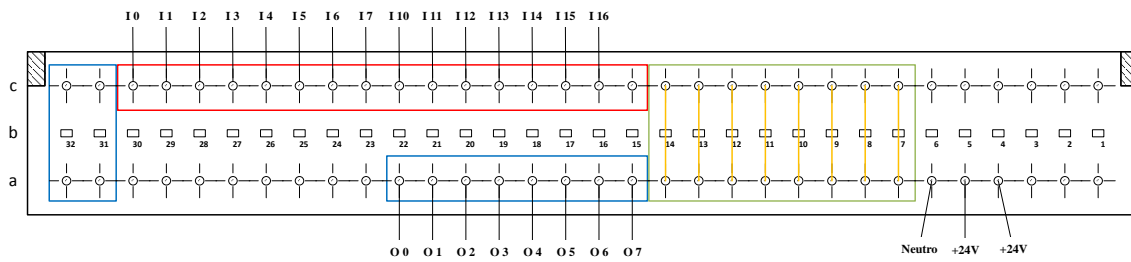


**Figura 4.3 – Painel de cilindros utilizado.**

O painel é composto por: 1 – Ficha DIN 41612, 2 – Fonte de Alimentação, 3 – Botões I/O, 4 – Cilindro pneumático, 5 – Válvula de gaveta. O painel de cilindros como é visível na imagem é composto por quatro cilindros dispostos ao alto.

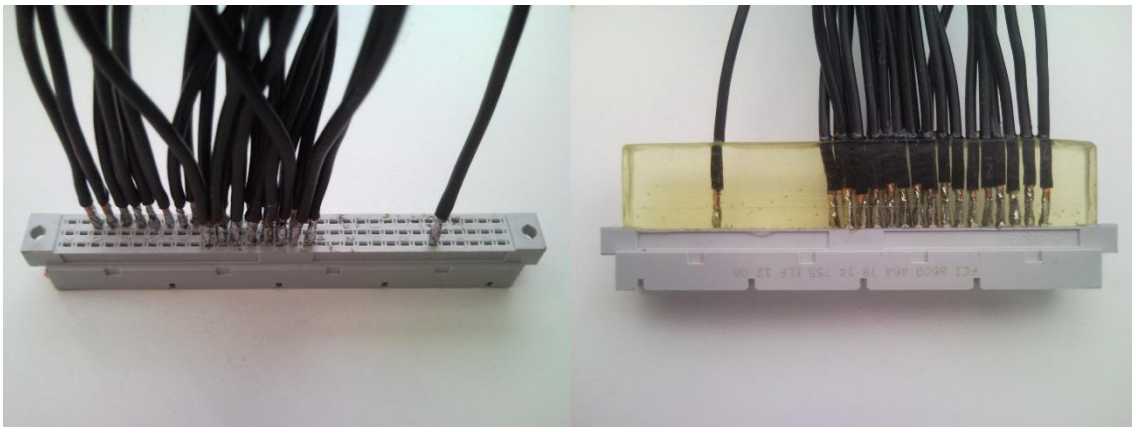
#### 4.2.2 Ficha DIN 41612

A utilização do painel teria que ser feita sem alteração do mesmo, para que se continuasse a poder utilizar os autómatos EBERLE, para isso foi necessário fazer com que a ligação do autómato utilizado o Siemens S7-1200 ao painel de cilindros fosse feita pela mesma ficha de ligação DIN 41612. Para que tal fosse possível teve que se adaptar uma ficha que apenas existe para ligar a PCB (*printed circuit board*). A adaptação foi feita soldando os fios dos sensores e atuadores a cada pino correspondente.



**Figura 4.4 – Contactos da ficha DIN 41612 para o S7-1200.**

Na Figura 4.4 apenas estão assinalados os pinos que foram soldados, sendo que os I's correspondem às entradas e os O's correspondem às saídas. Existe ainda o neutro e duas saídas de 24V que foram utilizadas para alimentar as saídas do autómato S7-1200, uma vez que este nas saídas apenas aciona relés, precisando de uma alimentação exterior.

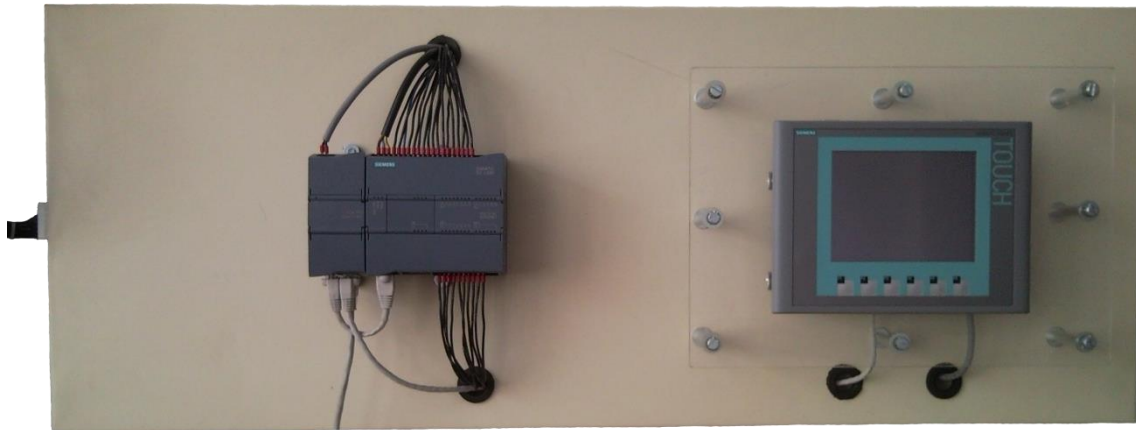


**Figura 4.5 – Ficha DIN 41612 finalizada.**

Para finalizar a ficha, os fios soldados foram isolados com um composto de resina epóxi, de forma a conservar, isolar e conferir resistência às soldaduras.

### 4.2.3 Montagem do Autómato S7-1200

Para a montagem do Autómato S7-1200 foi necessário utilizar uma base que fosse fácil de transportar e que desse para prender à parede junto do painel de cilindros. Para esse efeito foi utilizado uma chapa de metal com umas dobras nas extremidades, de forma a conferir altura para poder ocultar os fios da parte de trás. Nesta mesma estrutura foi ainda necessário arranjar forma de fixar o HMI junto do autómato para que todo o conjunto fique ligado. Essa fixação foi conseguida através de uma chapa de acrílico que em conjunto com oito pernes foi possível arranjar a altura suficiente para que o HMI pudesse ficar montado sem prejudicar a fixação na parede.



**Figura 4.6 – Autómato Siemens S7-1200 e HMI Siemens KTP600.**

#### 4.2.4 Montagem da Webcam

A montagem da *webcam* foi feita consoante a melhor localização para a visualização do painel de cilindros, tendo esta ficado instalada no teto da sala tal como mostra a figura abaixo.

A câmara utilizada é uma Samsung SBC-300P com imagem a preto e branco.



**Figura 4.7 – Localização da Webcam.**

### 4.3 Páginas Web do Laboratório Remoto do ISEL

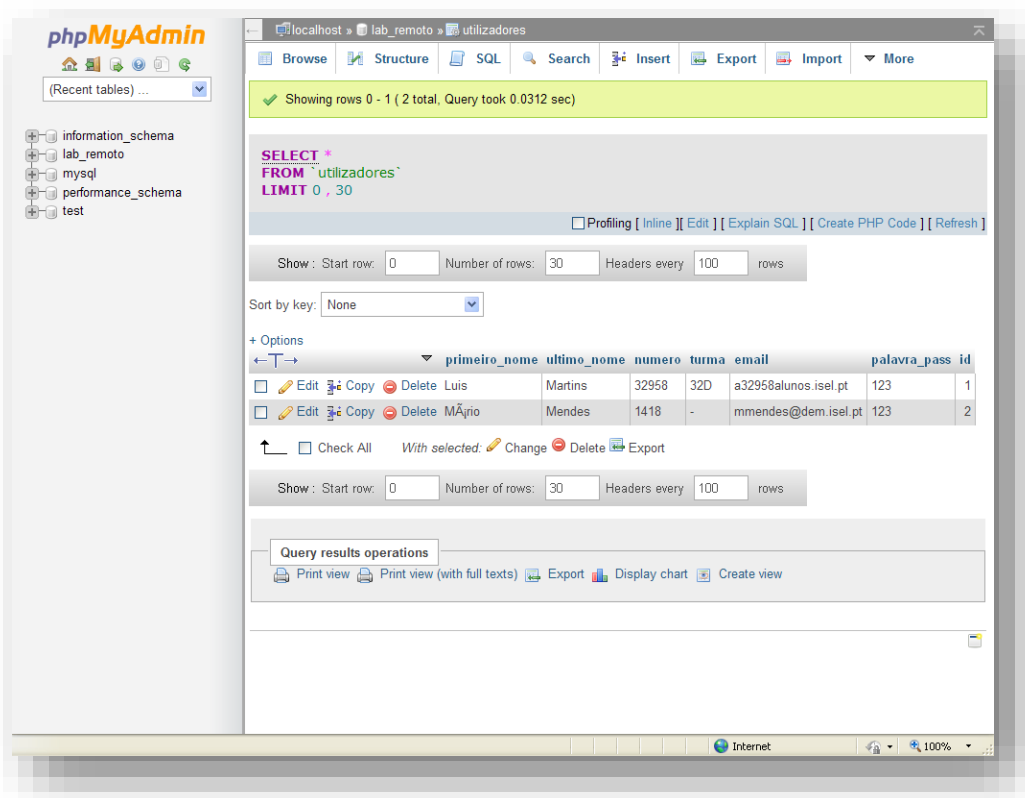
Neste subcapítulo pretende-se ilustrar todas as páginas em HTML e PHP elaboradas para o funcionamento do laboratório remoto do ISEL.

### 4.3.1 Gestão da Base de Dados

Para o laboratório remoto do ISEL foi criada uma base de dados com o nome “lab\_remoto”, com uma única tabela chamada “utilizadores” no PHPMyAdmin com os seguintes campos:

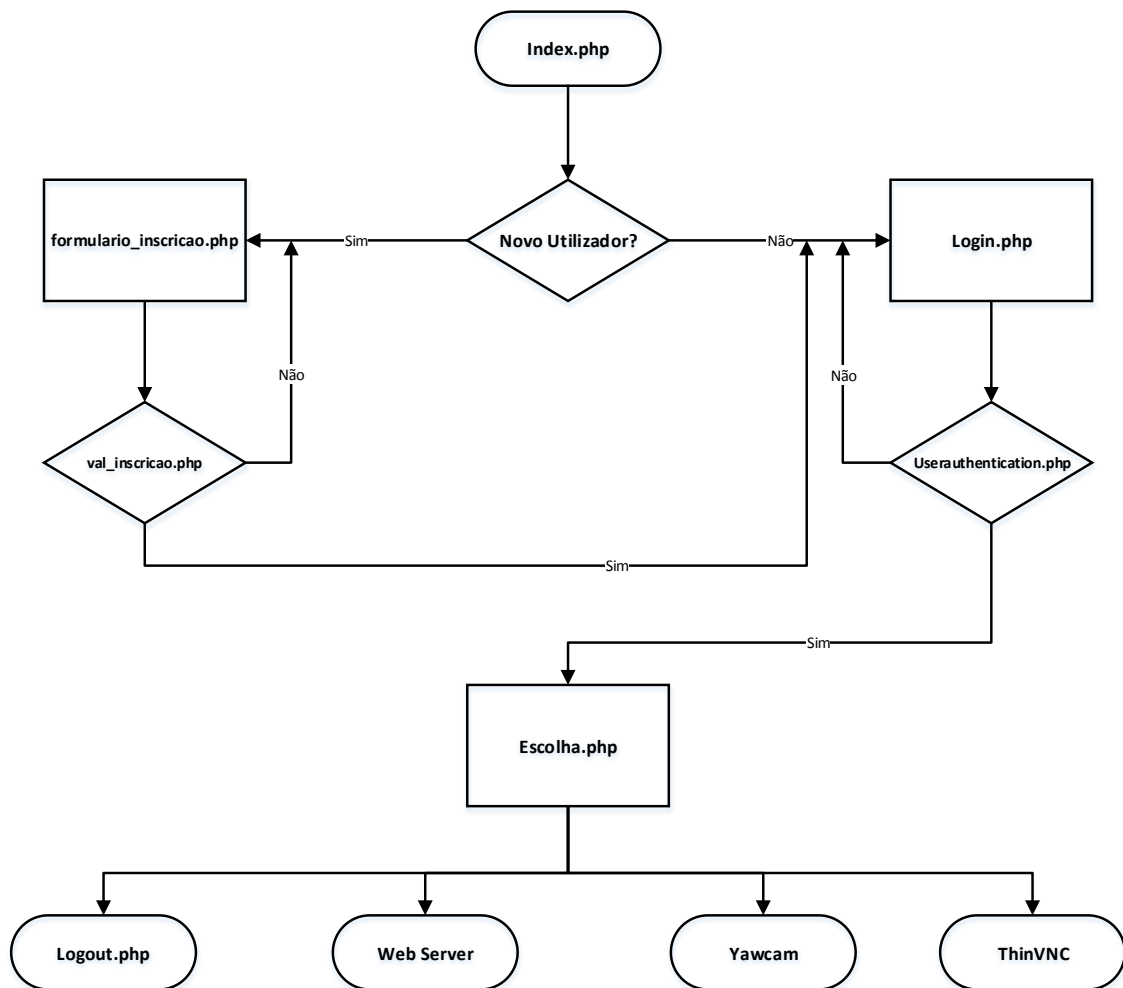
- Primeiro Nome;
- Ultimo Nome;
- Número de Aluno;
- Turma;
- E-mail;
- Palavra Passe.

Estes campos são todos de preenchimento obrigatório. Esta tabela fica então com todos os dados dos utilizadores que se registarem, tendo o administrador do laboratório total controlo sobre todos eles através do PHPMyAdmin como já foi referido.



**Figura 4.8 – Base de Dados do MySQL no PHPMyAdmin.**

Com a base de dados finalizada foram criadas seis páginas em PHP para adicionar utilizadores do laboratório no sistema de gestão de bases de dados MySQL. Com a ajuda de um fluxograma torna-se perceptível de como foram organizadas as páginas:



**Figura 4.9 – Fluxograma da gestão da base de dados.**

Através do fluxograma é possível verificar o quão simples é a estrutura de gestão da base de dados do laboratório.

O utilizador tem a primeira página de entrada no laboratório remoto chamada “index.php”. Nesta primeira página este define se quer efetuar uma nova inscrição ou se já se encontra inscrito e pretende apenas fazer login no laboratório. Se escolher efetuar a inscrição terá que preencher todos os campos já indicados e posterior validação dos mesmos. Caso não seja validada a inscrição o utilizador voltará para a página de preenchimento dos campos.

Caso o utilizador no início queira efetuar o login basta preencher com o *e-mail* e palavra passe escolhidas aquando da inscrição no laboratório. Feito o login será efetuada uma validação da mesma, ou seja, será verificado na base de dados se os dados introduzidos estão ou não inseridos nesta.

Por fim, e com o login devidamente efetuado o utilizador entra na página de escolha para poder escolher o que visualizar:

- O *web server* do autômato S7-1200;
- A *webcam* que filma o painel de cilindros em tempo real;
- Ou por fim se quer ir para o ambiente de trabalho remoto para poder elaborar um novo projeto.

Caso queira sair poderá sempre fazer *logout* da página de escolha.

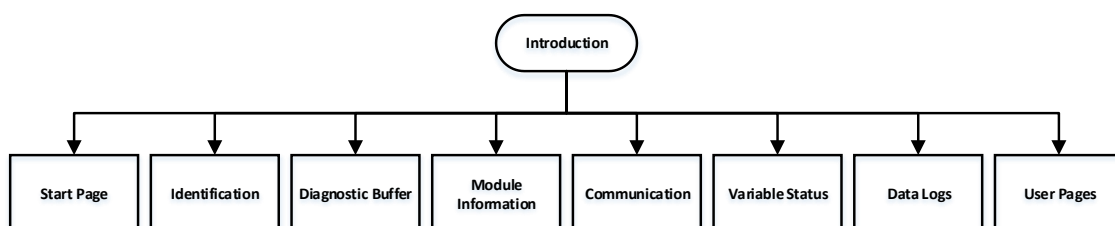
### 4.3.2 Web Server S7-1200

Como já foi referido o autômato utilizado Siemens S7-1200 tem a capacidade de pôr em linha um servidor *web*. O servidor *web* para o S7-1200 fornece acesso a páginas *web* com dados sobre a CPU e o estado em que as variáveis do projeto se encontram.

#### Standard Web Pages

O S7-1200 inclui páginas *web* padrão que o utilizador pode aceder a partir de um navegador *web*:

- Introdução - ponto de entrada para as páginas padrão;
- Página Inicial - informações gerais sobre a CPU;
- Identificação - Informações detalhadas sobre a CPU, incluindo o numero de série, ordem e versão;
- Informações do Módulo - informações sobre os módulos no bastidor local
- Comunicação - informações sobre os endereços de rede, propriedades das interfaces de comunicação, e as estatísticas de comunicação;
- Buffer de diagnóstico;
- Estado das variáveis - variáveis I/O da CPU, acessíveis pelo endereço ou nome da tag da variável do PLC;
- Históricos de Dados - Arquivos de dados armazenados internamente na CPU ou num cartão de memória;
- *Firmware Update* - atualizar o *firmware* da sua CPU (Siemens 2012).



**Figura 4.10 – Fluxograma das *Standard Web Pages* do S7-1200.**

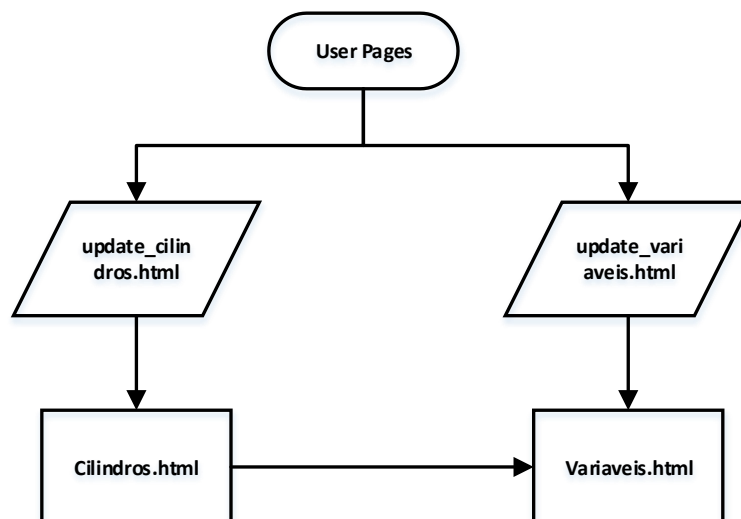
## User Pages

O autômato Siemens S7-1200 também fornece suporte para que o utilizador crie as suas próprias páginas *web*, que permitam por exemplo aceder às variáveis da CPU, desde que incluam os comandos "AWP" (*Automation Web Programming*). Os comandos AWP são um conjunto fixo de comandos que a Siemens fornece para aceder às variáveis do CPU. O utilizador pode desenvolver essas páginas com um *software* de criação e edição de páginas HTML à sua escolha e transferi-las para o CPU onde poderão ser acessíveis a partir do menu das "User Pages", como referido acima.

Este processo envolve várias tarefas:

1. Criação de páginas HTML com um editor de HTML;
2. Inclusão dos comandos AWP nos comentários HTML do código HTML da página;
3. Configuração do STEP 7 para ler e processar as páginas HTML;
4. Criação dos blocos das páginas HTML;
5. Programação do STEP 7 para controlar a utilização das páginas de HTML;
6. Compilação e *download* dos blocos para o CPU.

Como o âmbito do laboratório remoto não é ensinar os alunos a fazer páginas em HTML, foram criadas previamente quatro páginas para que os alunos possam apenas descarrega-las para o CPU e posteriormente aceder às mesmas e poderem utiliza-las como um complemento dos seus projetos.



**Figura 4.11 – Fluxograma das User Pages.**

Para uma melhor compreensão do que foi feito será feita uma pequena explicação do que é feito por cada uma das páginas HTML.

As páginas “update\_cilindros.html” e “update\_variaveis.html” não são páginas de visualização, estas apenas fazem uma leitura em tempo real do estado das variáveis com ajuda de uma pequena programação em javascript.

A página cilindros.html é onde os alunos podem fazer alguma interação com o autômato. Dentro desta página foram criados alguns botões para que os alunos possam comandar à distancia algumas variáveis da CPU. Em primeiro lugar foi definido o modo de funcionamento, uma vez que o autômato estará a comandar um painel de cilindros pneumáticos, o aluno poderá escolher se quer trabalhar em modo automático ou modo manual.

No modo automático o aluno terá que programar até três ciclos de funcionamento dos cilindros, e pôr cada um deles em funcionamento.

No modo manual o aluno poderá comandar cada cilindro independentemente dos outros, ou seja, poderá avançar e recuar cada um dos cilindros.

Cada um destes botões apenas funciona com *bits* que estejam associados a memórias. Para além disso, e para que os alunos possam escolher os bits associados a essas mesmas memórias foram criados nomes de etiqueta ou *tags* para as memórias como se pode verificar na Figura 4.12 e Tabela 4.1. Aquando da definição dos endereços no programa do TIA Portal os alunos deverão aplicar estes nomes para que as páginas funcionem como se pretende.

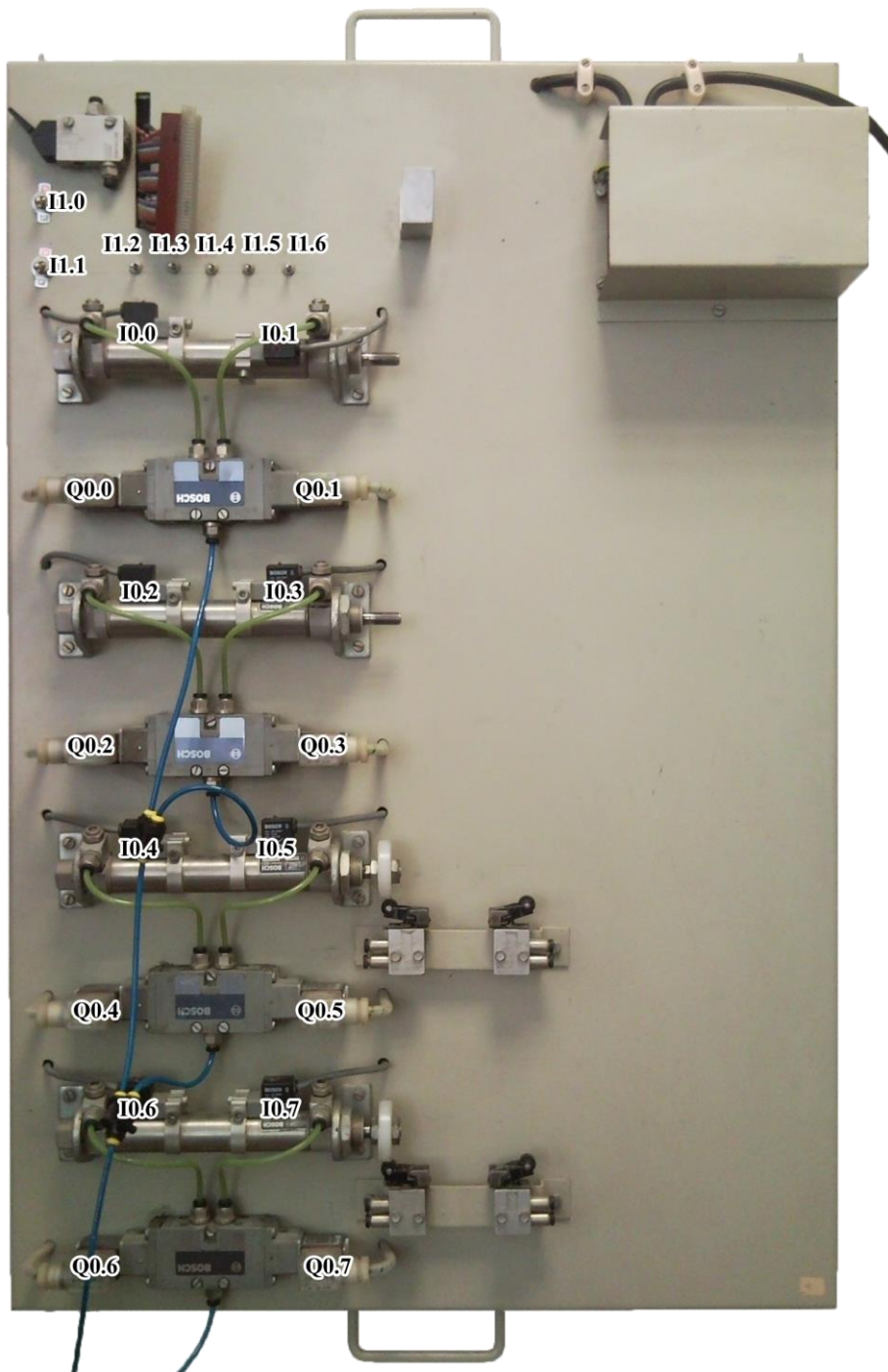


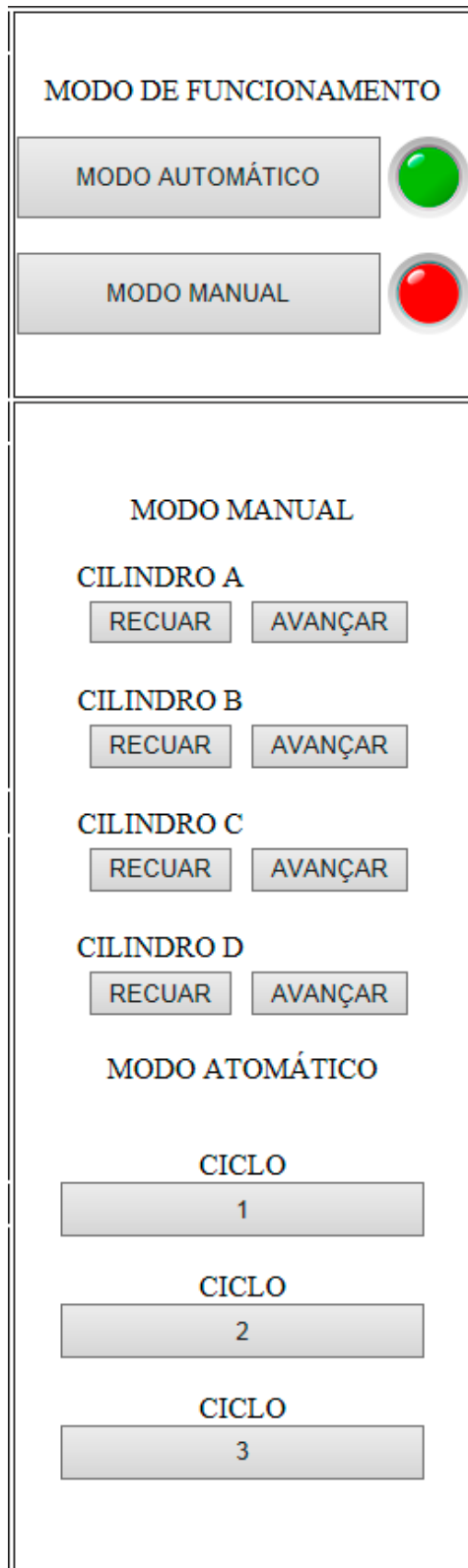
Figura 4.12 – Painel com os respetivos endereços.

**Tabela 4.1 – Correspondências dos vários endereços do painel com as TAGS.**

<b>Entradas</b>	<b>TAG</b>	<b>Saídas</b>	<b>TAG</b>	<b>Memórias</b>	<b>TAG</b>
<b>I0.0</b>	input_0	Q0.0	output_0	MW0	AWP
<b>I0.1</b>	input_1	Q0.1	output_1	M4.0	modo_auto
<b>I0.2</b>	input_2	Q0.2	output_2	M4.1	modo_manual
<b>I0.3</b>	input_3	Q0.3	output_3	M4.2	ciclo_1
<b>I0.4</b>	input_4	Q0.4	output_4	M4.3	ciclo_2
<b>I0.5</b>	input_5	Q0.5	output_5	M4.4	ciclo_3
<b>I0.6</b>	input_6	Q0.6	output_6	M5.0	cilindro_A
<b>I0.7</b>	input_7	Q0.7	output_7	M5.1	cilindro_B
<b>I1.0</b>	input_8			M5.2	cilindro_C
<b>I1.1</b>	input_9			M5.3	cilindro_D
<b>I1.2</b>	input_10				
<b>I1.3</b>	input_11				
<b>I1.4</b>	input_12				
<b>I1.5</b>	input_13				
<b>I1.6</b>	input_14				

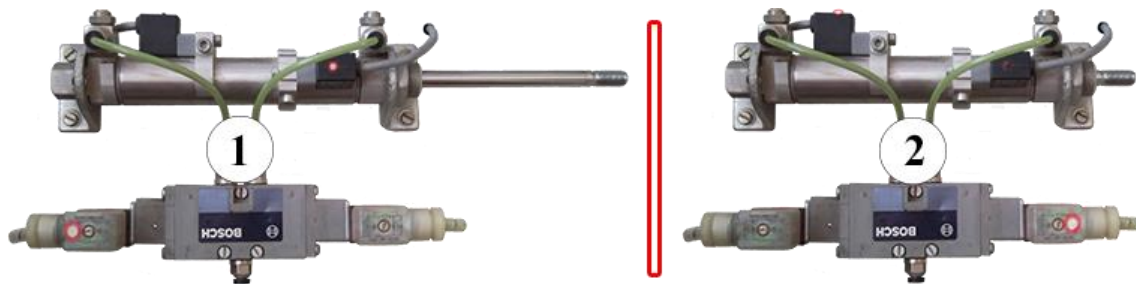
Os botões do modo de funcionamento funcionam com dependência um do outro, ou seja, por exemplo o botão referente ao modo automático quando pressionado passa o valor da memória M4.0 a 1 e o valor da memória M4.1 referente ao modo manual a 0, sendo que quando pressionado o botão do modo manual acontece precisamente o inverso.

Os botões referentes ao avanço e recuo dos cilindros funcionam aos pares, ou seja, quando pressionado o botão avançar a memória referente a esse cilindro passa a valer 1, e quando pressionado o botão de recuo o valor da memória associada a esse cilindro passa a 0.



**Figura 4.13 – Botões da página Cilindros.html.**

Dentro desta página foi ainda criada uma pequena interação com as variáveis de saída do painel, ou seja, dos cilindros pneumáticos. Com recurso às imagens do cilindro avançado e recuado foi possível recriar a fase em que este se encontra avançado e recuado.



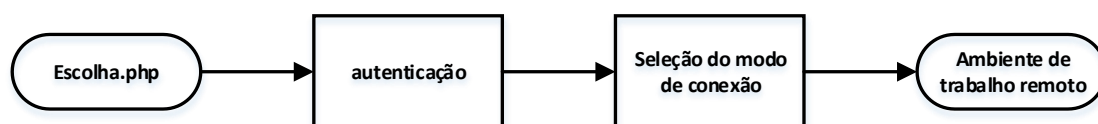
**Figura 4.14 – Imagens utilizadas para representar os cilindros nos estados, 1-avançado e 2-recuado.**

Como é possível verificar com na Figura 4.14 foi também recriado o momento em que os leds dos sensores estão ligados ou ativos.

A página `variaveis.html` é apenas para fazer uma leitura do estado das variáveis em tempo real. Esta página foi criada para se poder ver o estado das variáveis sem ter de estar sempre a fazer atualização da página, pois estas também seriam possíveis de ver dentro das “*Standard Web Pages*”, mais propriamente na página “*Variable Status*”, mas com o inconveniente de ter que se estar constantemente a atualizar a página para poder ver a mudança de estado das variáveis.

### 4.3.3 ThinVNC

Apos fazer login, e o mesmo ser bem-sucedido, o utilizador entra na página de escolha do que quer visualizar. Ao escolher o ambiente de trabalho remoto entra diretamente nas páginas do *software* ThinVNC que segue a estrutura abaixo indicada.



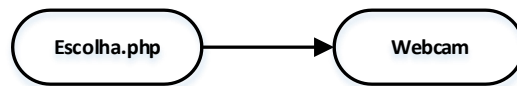
**Figura 4.15 – Fluxograma do ThinVNC.**

A primeira página prende-se com o *login* no programa, consoante um nome de utilizador e uma palavra passe pré definidos e que é igual para todos. Feito o *login* chega-se à página do modo de conexão, sendo que este está pré-definido apenas para o compartilhamento de tela. Fazendo conectar ficamos com o separador do ambiente de trabalho remoto aberto para utilização do mesmo.

No final da utilização basta desconectar e fechar o separador.

#### 4.3.4 Yawcam

A página referente ao Yawcam apenas apresenta a imagem da camara, sendo que dentro da página de escolha é possível aceder ao *link* que faz a ligação á pagina.



**Figura 4.16 – Fluxograma do Yawcam.**

## 5 Resultados

### 5.1 Consultar Conteúdos do Laboratório Remoto

#### 5.1.1 Base da Dados

Como referido o tipo de utilizador é dividido em duas categorias, a de utilizador externo ao ISEL e a do aluno ligado dentro da rede *ethernet* do ISEL.

Começando pelos alunos exteriores ao ISEL, os passos que estes terão que seguir para aceder ao laboratório remoto será:

1. Ir até ao site “<http://www.net.ipl.pt/>” e descarregar a aplicação “IPLNet-VPNIntra” para aceder á rede do ISEL através de VPN;



**Figura 5.1 – IPLNet-VPNIntra.**

2. Após ser concedida a autorização para a ligação da VPN fazer o respetivo *login*;

A partir do terceiro ponto é igual para os dois tipos de utilizadores, pois após a ligação da VPN é como se o aluno estivesse dentro da rede *ethernet* do ISEL.

3. Abrir o navegador *web* e aceder ao endereço <http://10.4.64.33/>;



Figura 5.2 – Index.php.

4. Escolher entre o fazer *login* ou fazer a inscrição, se a opção for a inscrição, quando esta estiver finalizada regressa ao *login*;



Figura 5.3 – Login e Inscrição.

5. Após o *login* bem-sucedido aparecerá a página de escolha ou menu, onde escolhemos então para onde ir;



**Figura 5.4 – Escolha ou Menu de Navegação.**

Na página de escolha o aluno poderá escolher o que pretende fazer. A partir deste ponto o utilizador quando clicar em cada um dos *links* será aberto um novo separador no navegador *web*, para que o utilizador nunca perca o menu.

### 5.1.2 Ambiente de Trabalho Remoto

Chegado à página de escolha, o utilizador em primeiro lugar deverá começar pelo ambiente de trabalho remoto onde vai fazer o seu programa e posteriormente enviá-lo para o autómato.

6. Clicar no *link* do Ambiente de Trabalho Remoto;



The image shows the ThinVNC login page. At the top left is the ThinVNC logo. Below it, the text "Enter your credentials" is displayed. There are three input fields: "PIN:" with the value "706 872 156", "Username:" which is empty, and "Password:" which is empty. A "Log In" button with a lock icon is positioned to the right of the password field. Below the login fields, there is a warning icon and a message: "In order to use this service, you must install ThinVNC on a computer in the LAN. Once installed turn the 'Enable Dynamic IP Address Resolution & Shared SSL Certificate' option on. Please click [here](#) to download the ThinVNC installer. Get a [free license](#)." At the bottom left, there is a link: "Are you looking for ThinVNC Access Point? Click [here](#)." At the bottom right is the CYBELE software logo.

**Figura 5.5 – Login ThinVNC.**

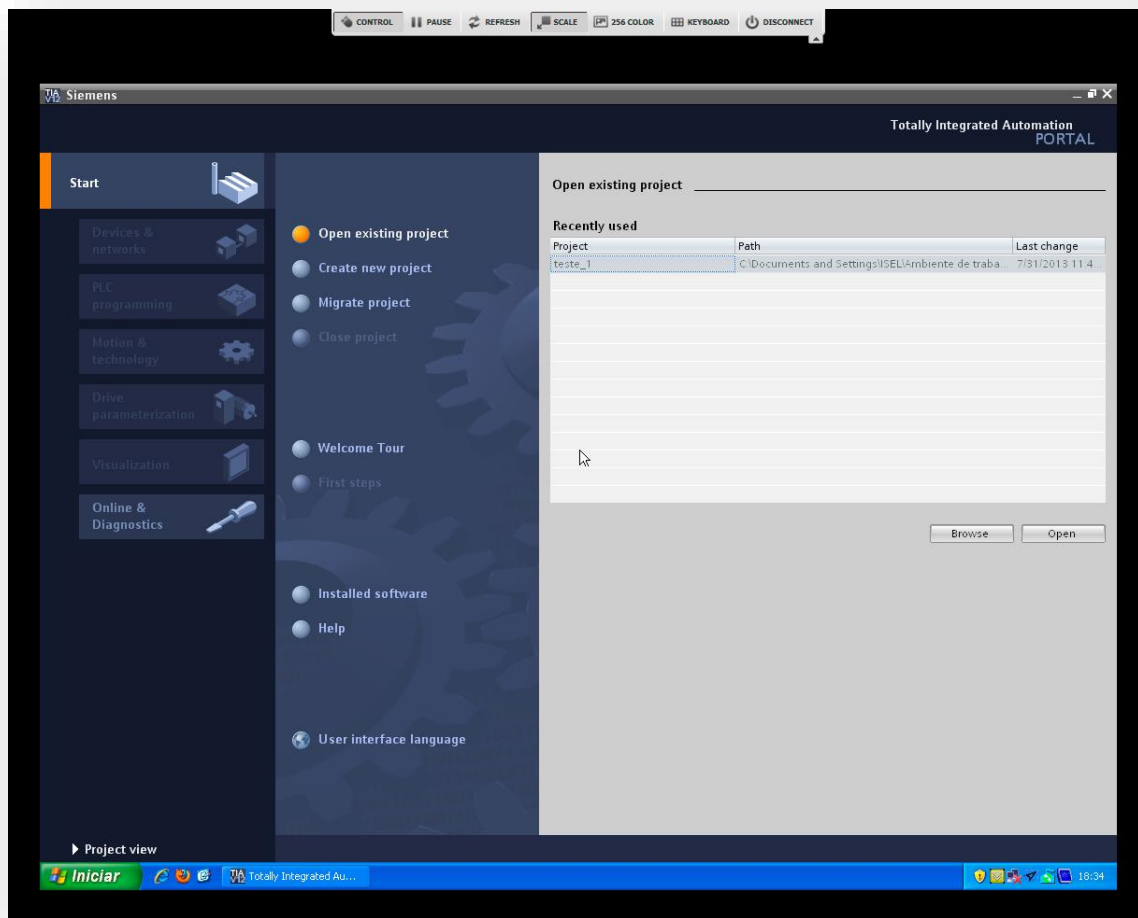
7. Fazer a *login* requerido com o nome de utilizador e palavra passe fornecidas pelo docente;



The image shows the ThinVNC connection options page. At the top left is the ISEL logo (INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA). At the top right is the ADEM logo (Área Departamental de Engenharia Mecânica). Below the logos is a banner that reads "LABORATÓRIO REMOTO DE AUTOMAÇÃO DE PROCESSOS INDUSTRIAIS". The main content area has a heading "Selecione um modo de conexão:" followed by three radio button options: "Compartilhamento de Tela" (selected), "Área de Trabalho Remota", and "Transferência de Arquivos". Below these options is a checked checkbox labeled "Abrir em uma nova janela do navegador". At the bottom right is a "Conectar" button with a green arrow icon.

**Figura 5.6 – Opções de ligação do ThinVNC.**

8. No passo seguinte fazer “Conectar” na opção “Compartilhamento de Tela”;



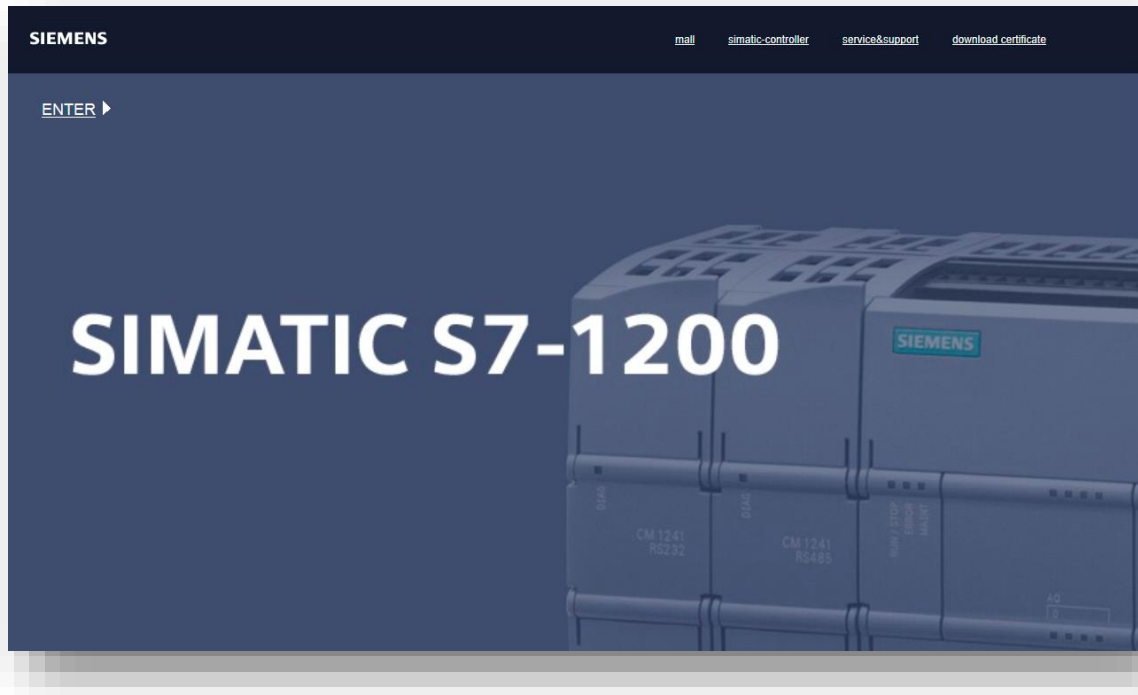
**Figura 5.7 – Ambiente de Trabalho Remoto do ThinVNC.**

Com o *login* efetuado o aluno poderá utilizar todas as funcionalidades do *software* TIA Portal, incluindo fazer a programação da CPU, o programa requerido e descarregar o mesmo para o autómato.

### 5.1.3 Web Server

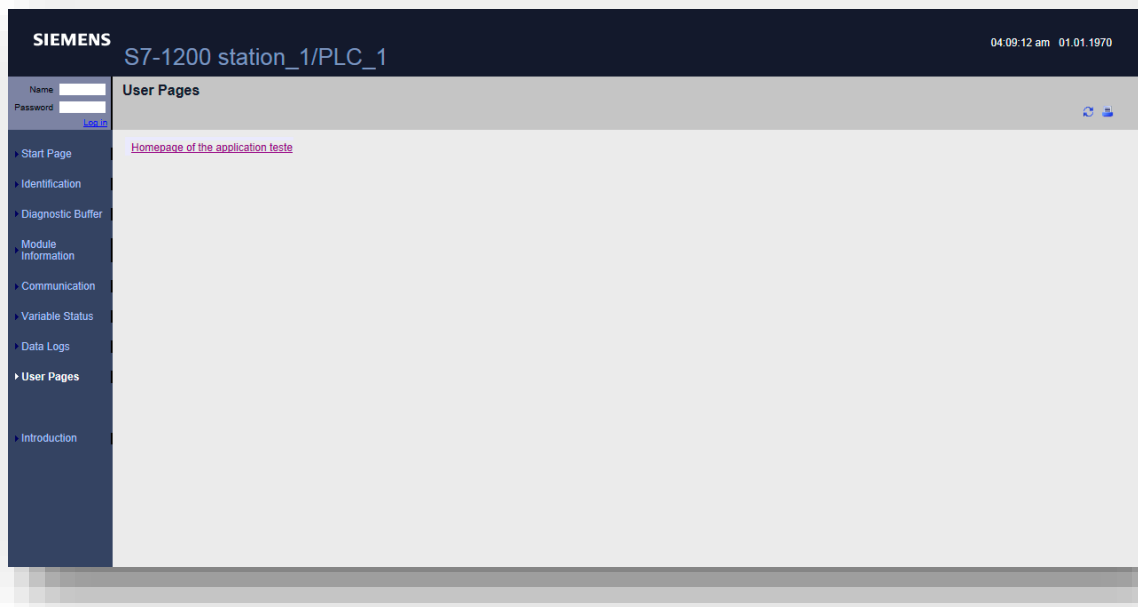
Em segundo lugar o utilizador deverá entrar no *web server* do autómato S7-1200, que deverá ter sido configurado devidamente nos passos anteriores.

9. Fazer “ENTER” dentro do *web server*;



**Figura 5.8 – Web Server S7-1200.**

10. Entrar nas *standard web pages* do S7-1200 clicando em “ENTER”, fazer *login* com um nome de utilizador “admin” sem palavra-passe e entrar nas “*User Pages*”;









**Figura 5.9 – User Pages.**

11. Clicar em “Homepage of the application cilindros” para visualizar as páginas elaboradas;

**LABORATÓRIO REMOTO DE AUTOMAÇÃO DE PROCESSOS INDUSTRIAIS**

Cilindros  
Variáveis

<p><b>Cilindro A</b></p> 	<p>MODO DE FUNCIONAMENTO</p> <p>MODO AUTOMÁTICO </p> <p>MODO MANUAL </p>
<p><b>Cilindro B</b></p> 	<p>MODO MANUAL</p> <p>CILINDRO A RECUAR AVANÇAR</p> <p>CILINDRO B RECUAR AVANÇAR</p>
<p><b>Cilindro C</b></p> 	<p>CILINDRO C RECUAR AVANÇAR</p> <p>CILINDRO D RECUAR AVANÇAR</p>
<p><b>Cilindro D</b></p> 	<p>MODO ATOMÁTICO</p> <p>CICLO 1</p> <p>CICLO 2</p> <p>CICLO 3</p>

**Figura 5.10 – Página dos cilindros.**

Cilindros  
Variáveis

Variável	Estado
input_0	0
input_1	0
input_2	0
input_3	0
input_4	0
input_5	0
input_6	0
input_7	0
output_0	0
output_1	0
output_2	0
output_3	0
output_4	0
output_5	0
output_6	0
output_7	0

**Figura 5.11 – Página das variáveis.**

Neste ponto o aluno pode escolher entre as páginas “Variáveis” e “Cilindros”. Na página dos cilindros é onde o aluno pode interagir com o autómato, escolhendo o modo de funcionamento pretendido, o automático ou manual, tal como já foi referido anteriormente.

#### 5.1.4 Webcam

Por fim, de forma a o aluno poder visualizar tudo o que foi feito nos passos anteriores, surge a página da *webcam* para se poder visualizar o painel ao vivo.

12. Clicar no *link* “Webcam”;

It's a webcam!



Powered by [www.vivacam.com](http://www.vivacam.com)

**Figura 5.12 – Webcam.**

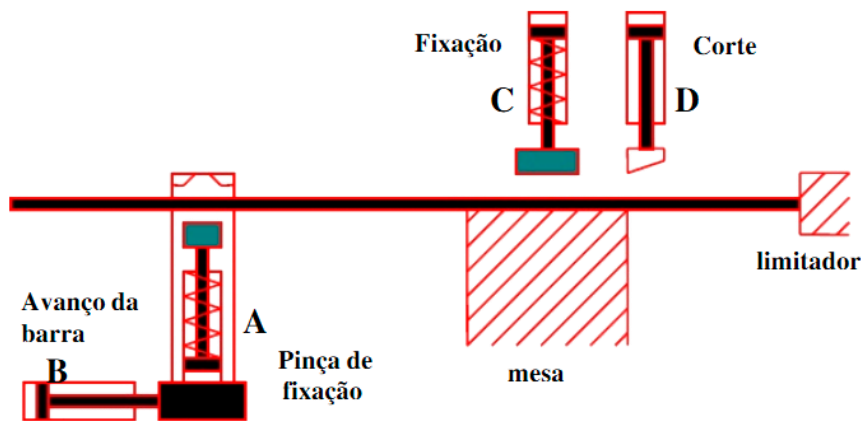
## 5.2 Caso Prático

Vejamos um exemplo prático de um ciclo composto por quatro cilindros electropneumáticos, que é utilizado para o corte de folhas de cartão.

### Exemplo de um Ciclo Electropneumático

O ciclo electropneumático descrito na Figura 5.13 é o ideal para simular com o painel de cilindros. É um sistema completamente pneumático, composto por quatro cilindros apenas.

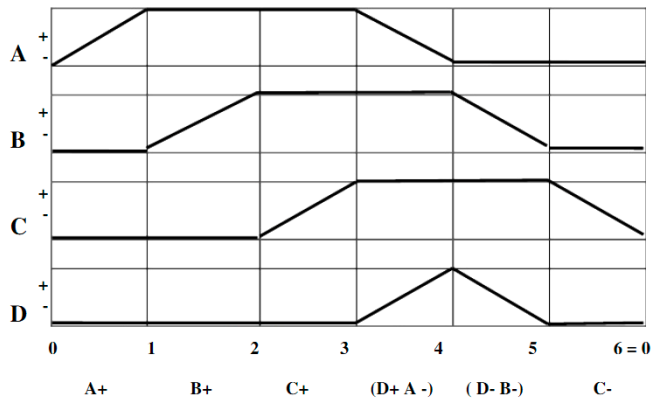
Os ciclos a programar podem ainda ter temporizadores ou contadores o que não é o caso no exemplo que se apresenta.



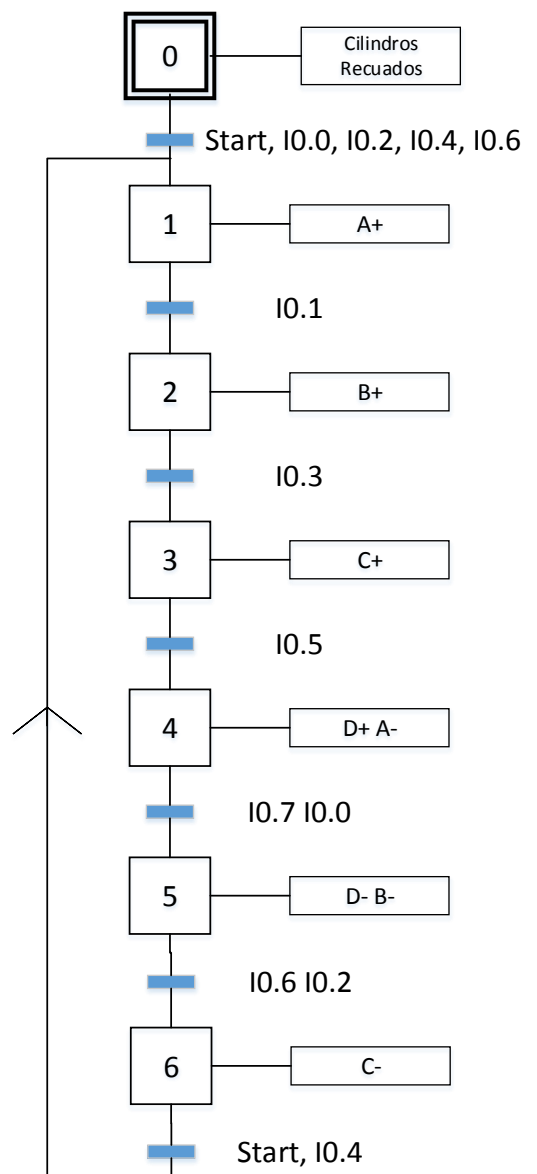
**Figura 5.13 – Ciclo Electropneumático.**

Seja o dispositivo de corte de folhas de cartão, mostrado no esquema acima. O “cilindro A” fixa a folha ao dispositivo de avanço, o “cilindro B” faz o avanço desta até a posição de corte, o “cilindro C” faz a fixação sobre a mesa e, após isto, o “cilindro D” avança iniciando o corte da folha, simultaneamente com o retorno do “cilindro A”. Cortada a folha, o “cilindro D” retorna simultaneamente com o “cilindro B”, que assim se posiciona para uma nova alimentação. O “cilindro C” retorna libertando a folha da sua fixação sobre a mesa, concluindo o ciclo e permitindo assim o reinício da operação.

A Figura 5.15 representa o diagrama temporal e Figura 5.14 a programação simplificada em linguagem SFC:



**Figura 5.15 – Diagrama Temporal do Ciclo Electropneumático.**

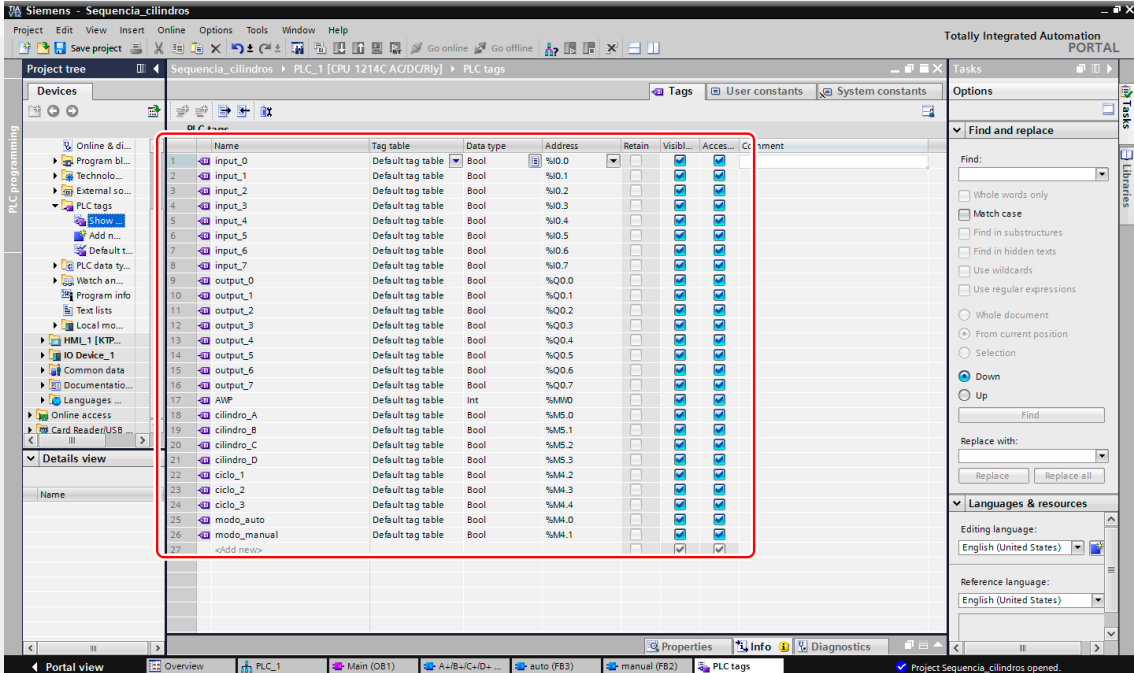


**Figura 5.14 – Esquema em linguagem gráfica SFC do ciclo Electropneumático.**

## Configuração dos Blocos de Funções e das Tags

Tendo em conta tudo o que foi explicado nos subcapítulos anteriores os passos a seguir são:

1. Aceder ao Ambiente de Trabalho Remoto e dentro do TIA Portal clicar em “*Create new project*”, na vista de “*Portal view*” de forma a criar um novo projeto.
2. Criado o novo projeto, mudar para a vista de “*Project view*” e ir até “*PLC tags*”, “*Show all tags*”, e introduzir rigorosamente todas as *tags* e os respetivos endereços indicados na tabela 4.1.



ID	Name	Tag table	Data type	Address	Retain	Visibl...	Access...	Comment
1	input_0	Default tag table	Bool	%I0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
2	input_1	Default tag table	Bool	%I0.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
3	input_2	Default tag table	Bool	%I0.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
4	input_3	Default tag table	Bool	%I0.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
5	input_4	Default tag table	Bool	%I0.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
6	input_5	Default tag table	Bool	%I0.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
7	input_6	Default tag table	Bool	%I0.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
8	input_7	Default tag table	Bool	%I0.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
9	output_0	Default tag table	Bool	%Q0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
10	output_1	Default tag table	Bool	%Q0.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
11	output_2	Default tag table	Bool	%Q0.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
12	output_3	Default tag table	Bool	%Q0.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
13	output_4	Default tag table	Bool	%Q0.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
14	output_5	Default tag table	Bool	%Q0.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
15	output_6	Default tag table	Bool	%Q0.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
16	output_7	Default tag table	Bool	%Q0.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
17	AWP	Default tag table	Int	%MW0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
18	cilindro_A	Default tag table	Bool	%M5.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
19	cilindro_B	Default tag table	Bool	%M5.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
20	cilindro_C	Default tag table	Bool	%M5.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
21	cilindro_D	Default tag table	Bool	%M5.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
22	ciclo_1	Default tag table	Bool	%M4.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
23	ciclo_2	Default tag table	Bool	%M4.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
24	ciclo_3	Default tag table	Bool	%M4.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
25	modo_auto	Default tag table	Bool	%M4.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
26	modo_manual	Default tag table	Bool	%M4.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
27	<Add new>					<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Figura 5.16 – Lista de tags.

3. Dentro do “*Program blocks*” abrir o “*OB1*” e inserir três “*Networks*”. A primeira para o comando “*WWW*”, que aloja todas as “*User Pages*” posteriormente carregadas na memória “*MW0*” com a *tag* “*AWP*”. A segunda para acionar a função “*manual*”, que serve para chamar o “*Function block*” onde estará programada o modo de funcionamento manual dos cilindros. E a terceira para acionar a função “*auto*”, que serve para chamar o “*Function block*” onde estará programada o modo de funcionamento automático dos cilindros.

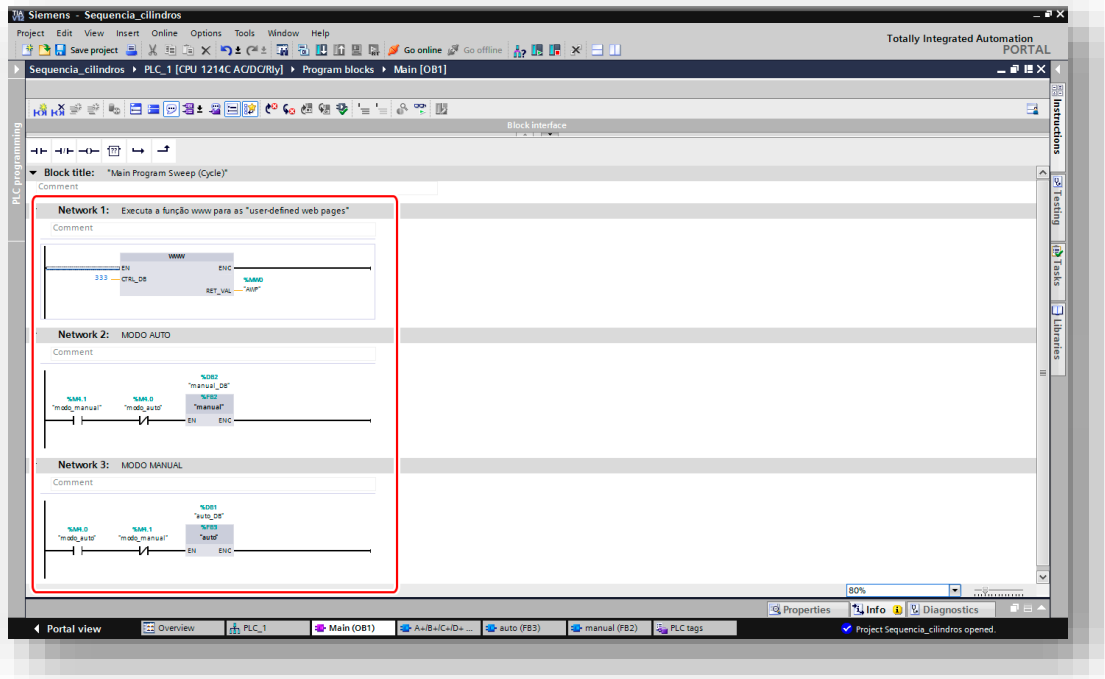


Figura 5.17 – Bloco de programação OB1.

- De seguida criar dois novos “Function block”, um para ser feita a programação do funcionamento automático e outro para o funcionamento manual dos cilindros electropneumáticos.

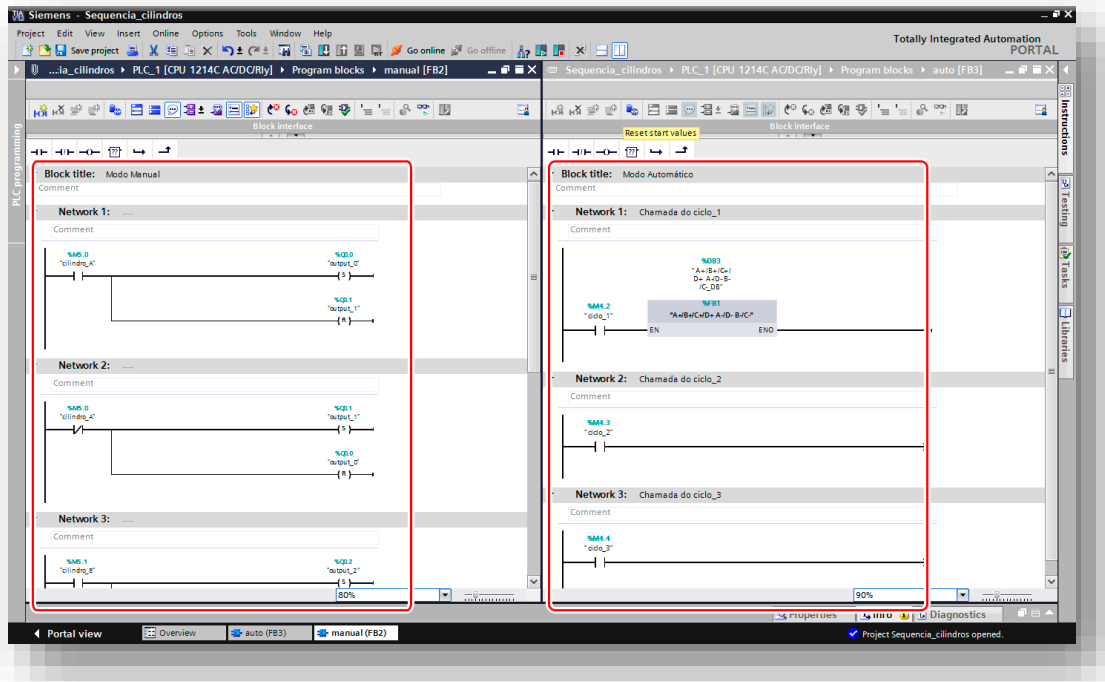
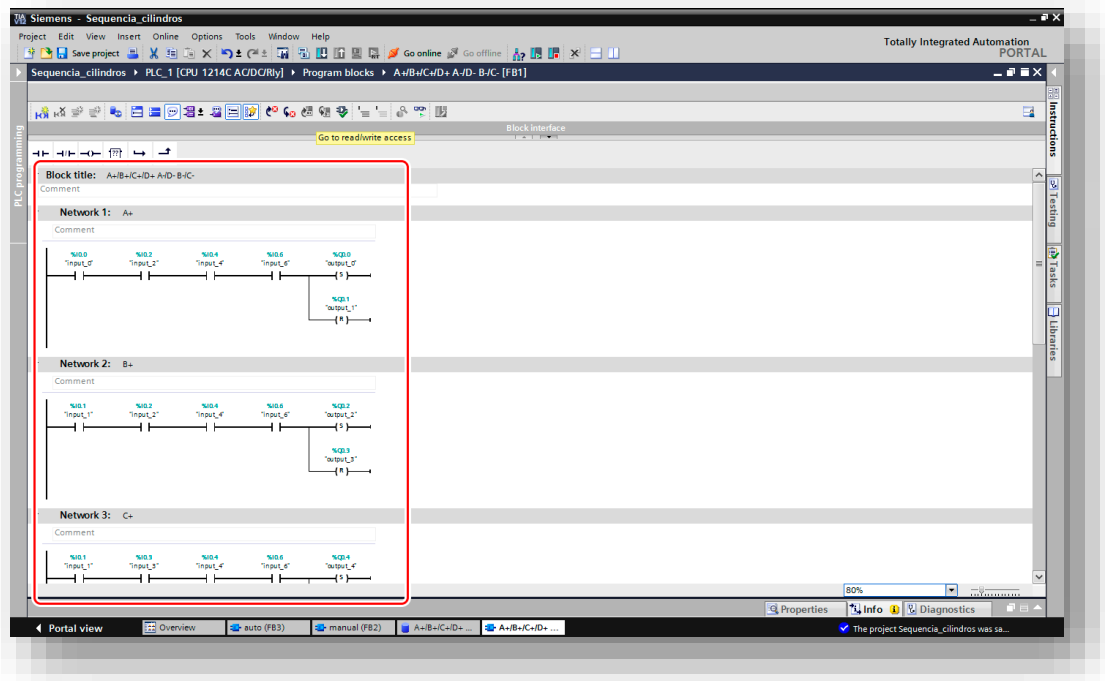


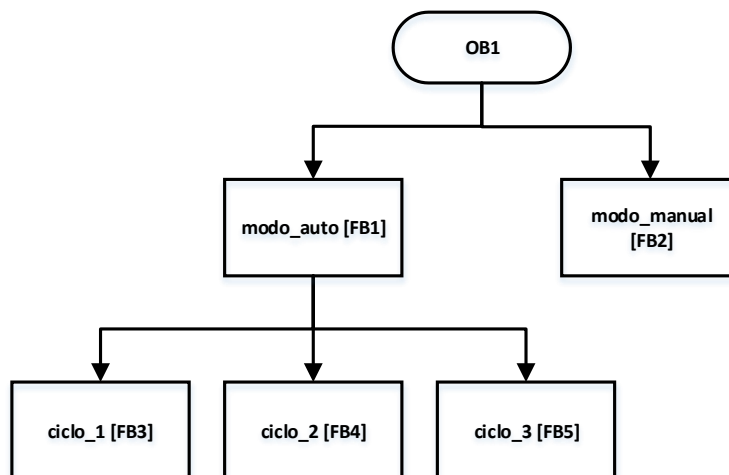
Figura 5.18 – “Function block” do funcionamento automático e manual dos cilindros electropneumáticos.

5. Criar uma “Function block” para cada ciclo electropneumático. A “User Pages” “Cilindros” dá para programar até três ciclos electropneumáticos. Dentro da “Function block” do funcionamento automático, cada “Network” chama um ciclo, como mostra na Figura 5.17. Na “Function block” do modo manual será feita toda a programação para que os cilindros funcionem manualmente com os botões da “User Pages” “Cilindros.html”, tal como foi explicado anteriormente.



**Figura 5.19 – Programação do ciclo A+/B+/C+/D+ A-/D- B-/C-.**

Para uma melhor compreensão de como se estruturam os blocos de funções aqui fica o fluxograma de toda a programação.



**Figura 5.20 – Fluxograma da programação dos ciclos electropneumáticos.**

Para que as “*User Pages*” criadas funcionem bem é necessário que os nomes das *tags* estejam exatamente iguais aos da tabela 4.1. Os endereços não têm de ser necessariamente iguais, as páginas HTML “Cilindros” e “Variáveis” funcionam apenas com o reconhecimento dos nomes das *tags*.

## Ativar o Servidor *Web* no TIA Portal

Para ativar o servidor *web* no TIA Portal, o utilizador terá que ir às configurações da CPU e seguir os seguintes passos:

1. Selecionar a CPU no menu “*Device Configuration*” na vista de projeto;
2. Na janela de inspeção, selecionar “*Web Server*” a partir das propriedades da CPU;
3. Selecionar “*Enable Web server on this module*”;
4. Para se requerer o acesso seguro ao servidor *web*, selecionar a opção “*Permit access only with HTTPS*”.

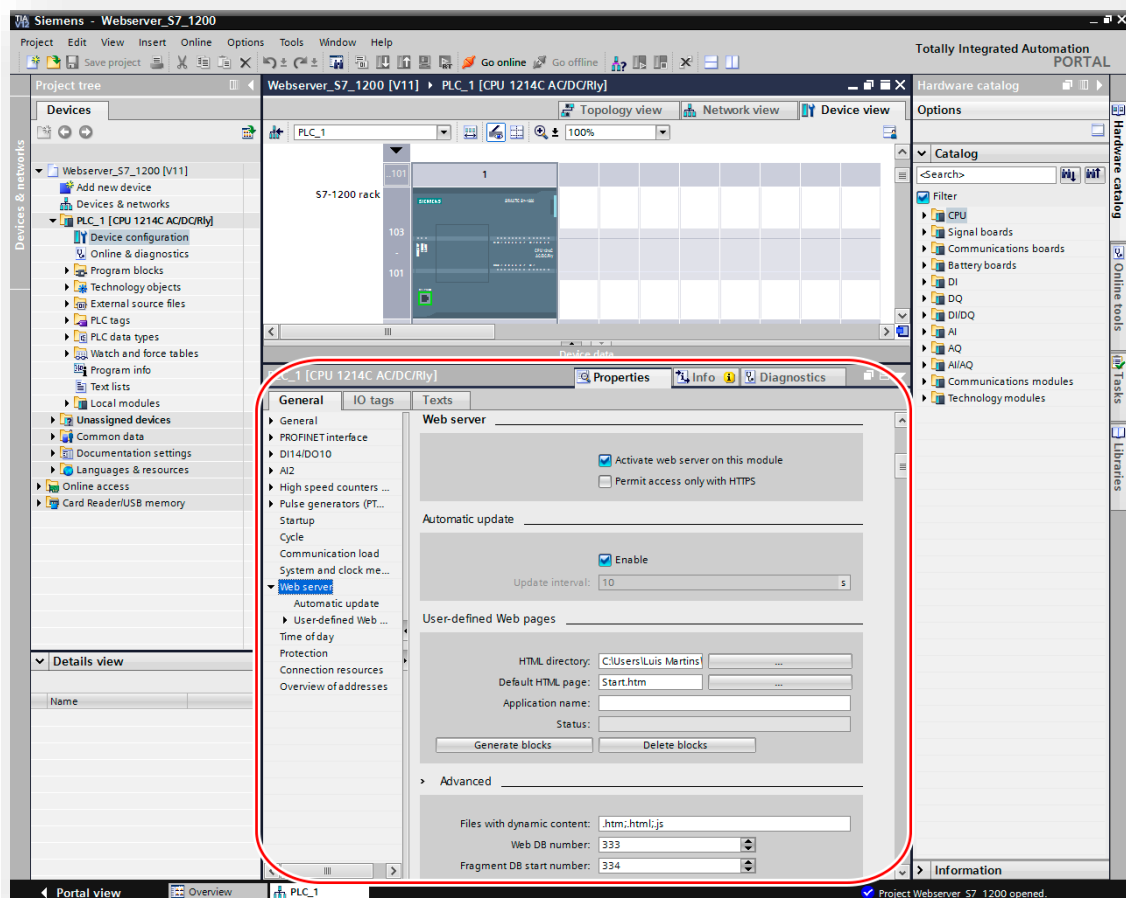


Figura 5.21 – Como ligar o *web server*.

## Configurar o IP no TIA Portal

Para configurar IP no TIA Portal o utilizador terá que ir às configurações da CPU e seguir os seguintes passos:

1. Selecionar a CPU no menu “*Device Configuration*” na vista de projeto;
2. Na janela de inspeção, selecionar “*PROFINET interface*” a partir das propriedades da CPU;
3. No campo “*Ethernet addresses*”, no quadrado “*IP protocol*”, selecionar “*Set IP address in the project*”, preencher o “*IP address*” com o IP 10.4.64.34 e o “*Subnet mask*” com máscara de rede 255.255.255.0;
4. No campo “*Ethernet addresses*”, no quadrado “*IP protocol*”, selecionar “*Use router*” e preencher o “*Router address*” com o IP 10.4.64.254;

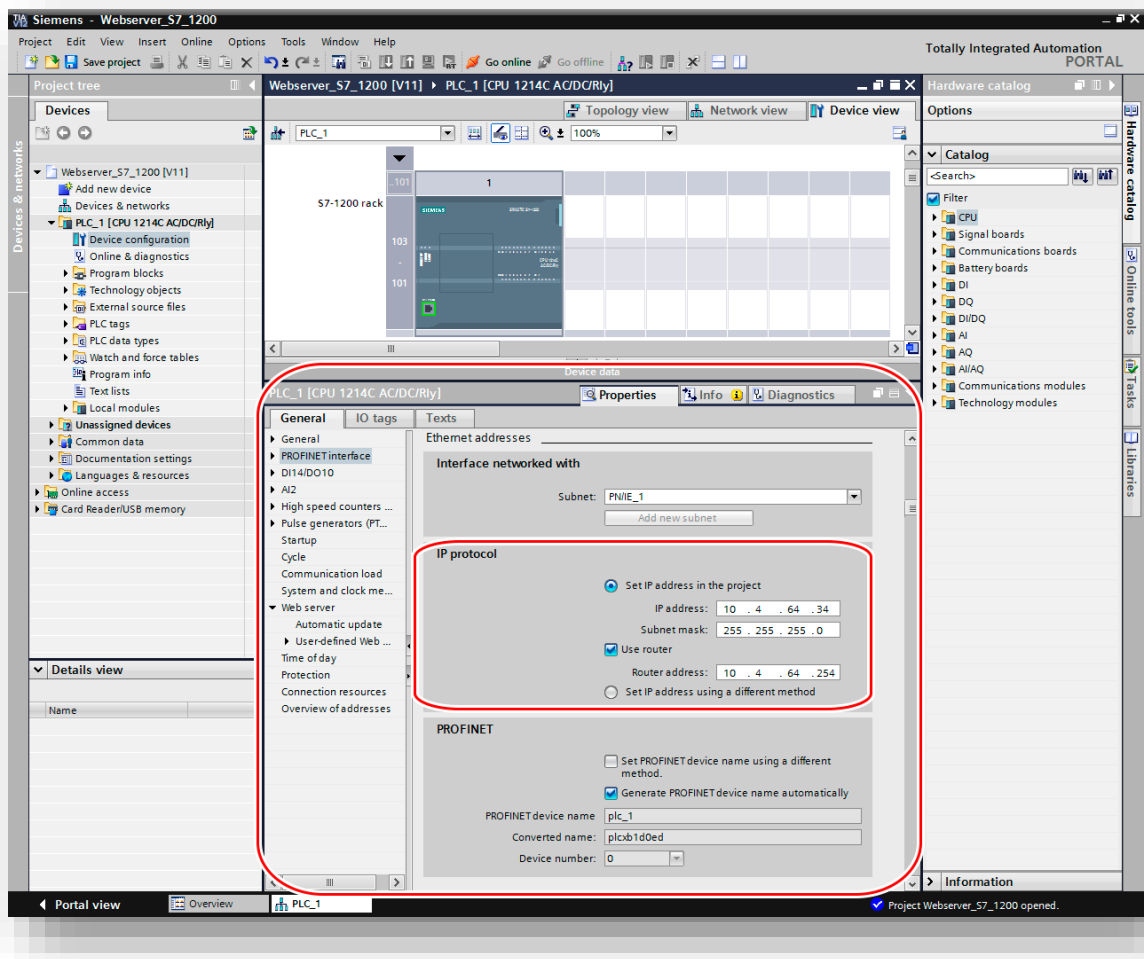


Figura 5.22 – Configurar o IP no TIA Portal.

Todos os endereços IP são configurados na CPU a quando do *download* do projeto. Se a CPU não tem um endereço de IP pré-configurado, deve-se associar ao projeto com

o endereço MAC do dispositivo de destino. Se a CPU é ligada em rede, também se deve inserir o endereço de IP do *router* como é o caso.

A opção "*Set IP address using a different method*" permite alterar o endereço de IP *on-line* ou usando a instrução "T\_CONFIG" após o *download* do projeto.

Este método de atribuição de endereços IP é apenas para a CPU.

### Configurar as “User defined Web pages” no TIA Portal

Para configurar as páginas HTML (*User pages*) no TIA Portal o utilizador terá que ir às configurações da CPU e seguir os seguintes passos:

1. Selecionar a CPU no menu “*Device Configuration*” na vista de projeto;
2. Na janela de inspeção, selecionar "*Web Server*" a partir das propriedades da CPU;
3. Selecionar "*User defined Web pages*";

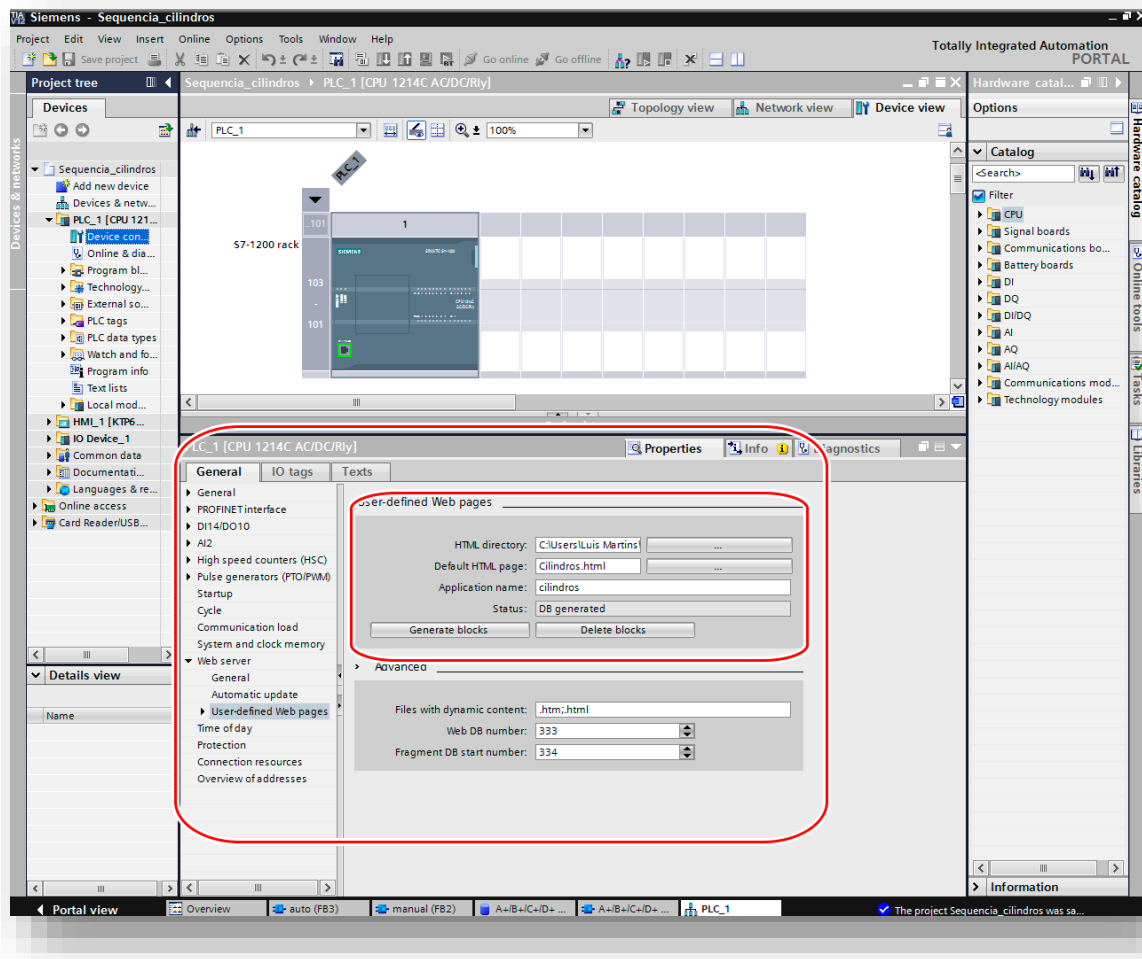


Figura 5.23 – Configurar o campo “*User defined Web pages*”.

4. Em “*HTML directory*” escolher a diretoria onde estão alojadas as páginas no computador. Em “*Default HTML page*” definir qual a página a abrir por omissão quando for acedido via *web server* do autómato, esta poderá ser a página “Cilindros.html” ou “variáveis.html”, uma vez que existem hiperligações que permitem mudar de uma para a outra. No campo “*Application name*” escrever o nome que o aluno achar conveniente para a aplicação.
5. Para finalizar, clicar em “*Generate blocks*” para o TIA Portal gravar as páginas no programa, para que quando este for descarregado para o autómato as “*User Pages*” possam estar disponíveis via *web server*.

### Descarregar programas para a CPU

Tendo tudo configurado e os ciclos devidamente programados, o utilizador tem de enviar o programa para a CPU, para isso terá que seguir os seguintes passos:

1. Clicar na CPU configurada dentro do painel de navegação;
2. Clicar no ícone “Download to device” em menus e barras de ferramentas;
3. Na nova janela, clicar em “Load”.

## 5.3 Gestor do Laboratório Remoto

A gestão do laboratório remoto será feita pelo docente. As tarefas que o gestor do laboratório terá para efetuar será a gestão dos utilizadores dentro da base dados.

### Gestão da Base Dados

Para a gestão das bases de dados será utilizado o PHPMyAdmin no computador responsável pela base de dados.

1. No lado direito da barra de tarefas clicar no ícone do Wampserver e clicar novamente em “phpMyAdmin”;



Figura 5.24 – Login PHPMyAdmin.

2. Fazer login com o nome de utilizador e palavra-passe escolhidos;

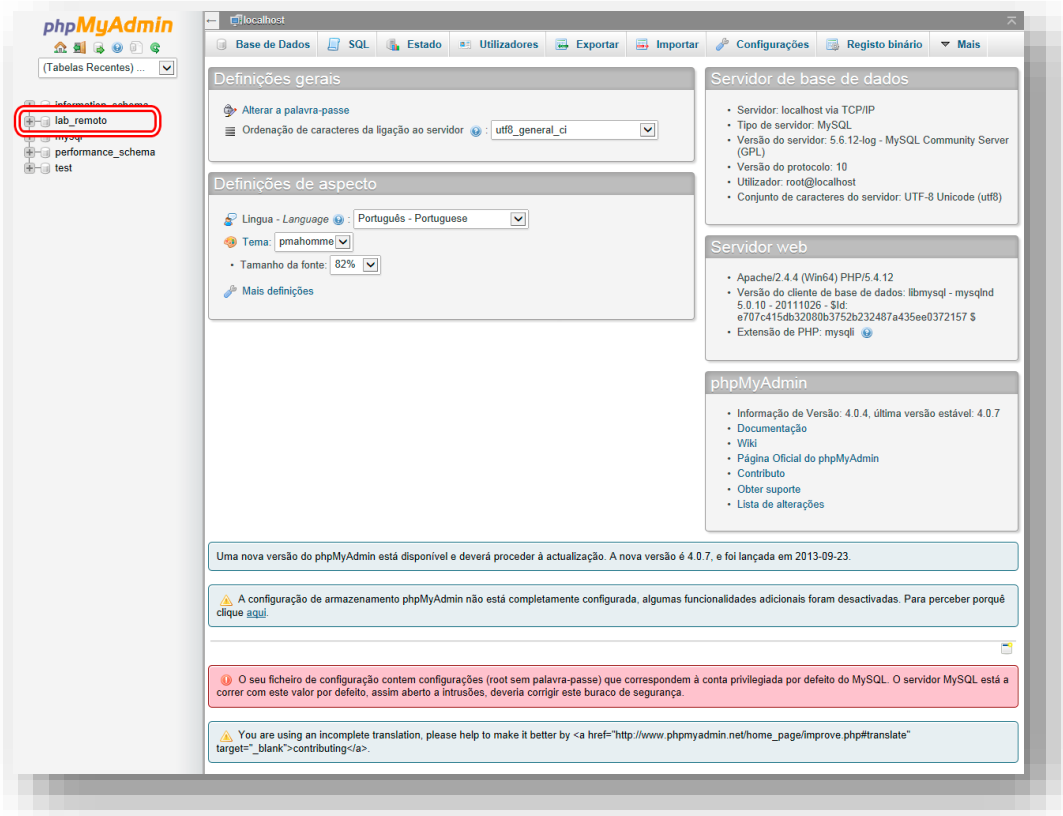


Figura 5.25 – Bases de Dados PHPMyAdmin.

3. Clicar na base de dados “lab\_remoto”;

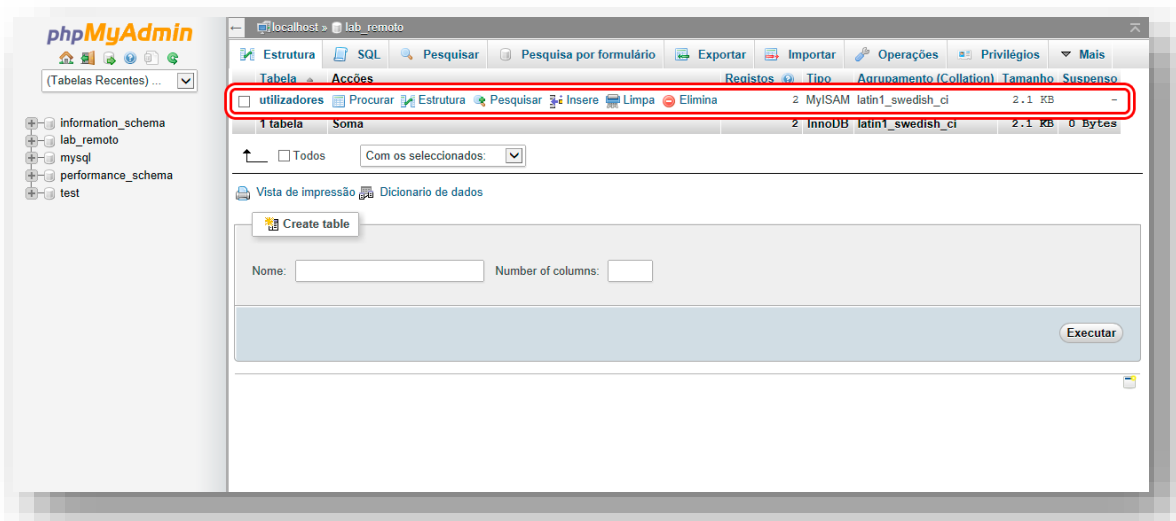


Figura 5.26 – Tabelas de utilizadores PHPMyAdmin.

4. Clicar na tabela dos “utilizadores”;

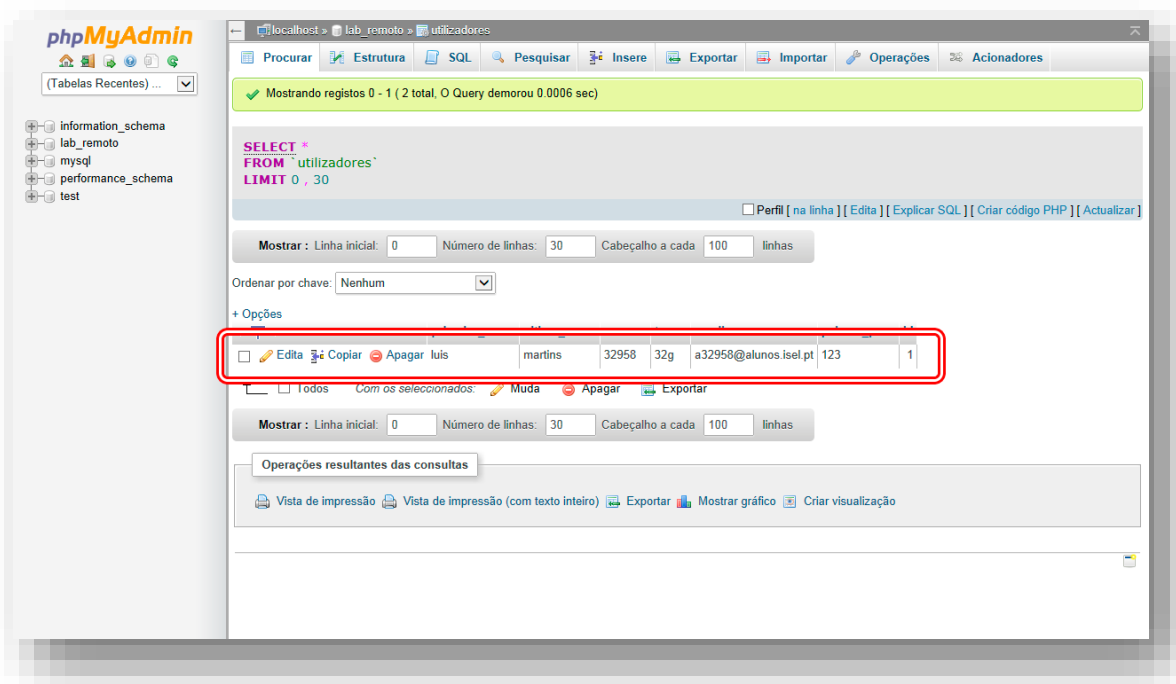


Figura 5.27 – Utilizadores do Laboratório Remoto PHPMyAdmin.

A partir daqui já é possível editar, apagar ou exportar utilizadores. Todo este controlo é feito pelo docente, sendo que a cada semestre será preciso apagar e adicionar novos utilizadores.



## 6 Conclusões

O desafio de implementar uma ferramenta de ensino baseada nas tecnologias de informação tornou-se numa tarefa difícil. Os conceitos a aplicar foram para lá da Engenharia Mecânica e da Automação, o que obrigou a uma maior pesquisa e aprendizagem de técnicas/áreas científicas a aplicar para que tudo funcionasse de forma simples. Para o caso específico do ensino da automação electropneumática à distância houve necessidade de fazer uma pesquisa das tecnologias existentes e analisar a melhor e mais barata solução para o laboratório remoto da ADEM/ISEL.

### 6.1 Trabalho Final

Tendo em conta todos os objetivos propostos no início deste trabalho de Mestrado, o laboratório remoto da ADEM/ISEL foi bem-sucedido. Fazendo uma análise de cada objetivo podemos ver que foram encontradas soluções para cada um deles:

O primeiro objetivo, de projetar um laboratório remoto para Automação de Processos Industriais reutilizando e reciclando o material existente no laboratório físico, foi alcançado, com o reaproveitamento do painel de cilindros electropneumáticos que estava a ser controlado por autómatos antiquados.

O segundo objetivo, de construir um *website* que fosse acessível por um navegador de internet e permitisse o acesso do laboratório remoto aos alunos, mediante um nome de utilizador e palavra passe, foi alcançado, com o auxílio de pequenas programações em PHP e javascript.

O terceiro objetivo, de permitir que o aluno pudesse aceder ao autómato de forma remota e fácil, foi concretizado, sendo que os alunos que estejam fora da rede ethernet do ISEL tenham de ter instalado uma aplicação que lhes permita aceder via VPN.

O quarto objetivo, da possibilidade de o utilizador poder fazer *upload* de vários programas para o autómato e testá-los, foi concretizado, via aplicação de acesso remoto a um computador com o *software* TIA Portal.

O quinto e último objetivo, de possibilitar a visualização em tempo real do painel de cilindros em funcionamento, foi alcançado, com auxílio de um *software* que disponibiliza as imagens em tempo real por navegador *web*.

Para além dos objetivos já enumerados, foi ainda possível utilizar uma das funcionalidades do servidor *web* do autómato Siemens S7-1200, as “*User Pages*”, e criar

uma página HTML que permite ao aluno interagir de forma remota com o autómato, e outra que mostra o estado das variáveis em tempo real.

Fazendo uma pequena comparação com os laboratórios remotos de automação industrial analisados, o laboratório remoto da ADEM/ISEL tem uma abordagem idêntica nalguns aspetos com o Laboratório Remoto de Automação Industrial (Lab-RAI), nomeadamente na parte da gestão da base de dados, utilizando o mesmo tipo de tecnologias. Noutros tem também algumas parecenças com o Laboratório Remoto de Automação Industrial do IST com a abordagem da interface dos cilindros electropneumáticos, e com a ligação através de VPN.

## **6.2 Trabalhos Futuros**

Este laboratório tem ainda alguma limitações, e como trabalho futuro existem algumas coisas que são possíveis de enumerar e que não foram concretizadas no âmbito deste trabalho por não serem exequíveis no prazo previsto para a sua elaboração. Assim sendo, como recomendação para os trabalhos futuros temos:

- Melhoria das interfaces das páginas *web*;
- Criação de sinóticos com outros programas, por exemplo o LabView;
- Possibilidade de utilizar mais autómatos ligados em rede;
- Adicionar mais experiências ao laboratório remoto.
- Adicionar uma botoneira de paragem de segurança.

## Referências Bibliográficas

- Barbosa, Fernando Sérgio, and Fernando M. S. Ramos. 1997. “Experiência de Desenvolvimento de Um Servidor Telemático Para Ensino à Distância.” *REVISTA DO DETUA, VOL. 1, Nº 7*.
- Benmohamed, Hcene, Arnaud Leleve, and Patrick Prevot. 2006. “Electronic Laboratories : ICTT @ Lab Experimentation.” In *4th International Conference on Education and Information Systems, Technologies and Applications, France*,.
- Borracha, António Manuel Lira Gomes. 2012. “Laboratório Remoto de Automação Industrial ( Lab - RAI ).” Faculdade de Ciências e Tecnologia Universidade Nova de Lisboa.
- Bourdon, Romain. 2013. “WampServer.” <http://www.wampserver.com/en/>.
- Carmo, Hermano. 1996. “Ensino a Distância e Desenvolvimento de Quadros Locais.” In *Forum 2000 sobre regionalização e desenvolvimento, F. C. Gulbenkian*,
- Centre, Algoritmi. 2012. “Automation & Control Remote Laboratory: Evaluating a Cooperative Methodology.” 34–39.
- Coelho, Paulo Jorge Marques. 2001. “Arquitectura Do Laboratório de Redes Remoto eCassiopeia.” Universidade do Porto.
- Corter, James E., Sven K. Esche, Constantin Chassapis, Jing Ma, and Jeffrey V. Nickerson. 2011. “Process and Learning Outcomes from Remotely-Operated, Simulated, and Hands-on Student Laboratories.” *Computers & Education* 57
- Cybele Software, Inc. 2013. “ThinVNC HTML5 Remote Access User’s Guide.”
- Deusto, University of. 2013. “Weblab-Deusto.” <http://www.weblab.deusto.es/website/>.
- Domínguez, M., J.J. Fuertes, P. Reguera, M.A. Prada, and A. Morán. 2008. “Inter-University Network of Remote Laboratories.” In *17th World Congress The International Federation of Automatic Control*,.
- Gomes, L, and S Bogosyan. 2009. “Current Trends in Remote Laboratories.” *IEEE Transactions on Industrial Electronics* 56(12): 4744–56.
- Hackl, Elsa. 2001. “Towards a European Area of Higher Education: Change and Convergence in European Higher Education.”
- Hardison, J, D Zych, J A Alamo, V J Harward, S R Lerman, S M Wang, K Yehia, and C Varadharajan. 2005. “The Microelectronics WebLab 6 . 0 – An Implementation Using Web Services and the iLab Shared Architecture.” In *Exploring Innovation in Education and Research*, Tainan.

- Harward, V Judson, Jesús A. del Alamo, Vijay S. Choudhary, Kimberley DeLong, James L. Hardison, Steven R. Lerman, Jedidiah Northridge, Daniel Talavera, Charuleka Varadharajan, Shaomin Wang, Karim Yehia, and David Zych. 2004. "iLab : A Scalable Architecture for Sharing Online Experiments Authors :” In *International Conference on Engineering Education*, Florida.
- Jara, Carlos a., Francisco a. Candelas, Santiago T. Puente, and Fernando Torres. 2011. "Hands-on Experiences of Undergraduate Students in Automatics and Robotics Using a Virtual and Remote Laboratory.” *Computers & Education* 57(4): 2451–61.
- De la Cruz, F., M. Díaz-Granados, S. Zerpa, and D. Giménez. 2010. "Web-LABAI: Laboratorio Remoto de Automatización Industrial.” *Revista Iberoamericana de Automática e Informática Industrial* 7(1): 101–6.
- License, General Public. 2013. "Bringing MySQL to the Web.” [http://www.phpmyadmin.net/home\\_page/](http://www.phpmyadmin.net/home_page/).
- Lopes, André, António Afonso, and Ricardo Antunes. 2005. "Redes Industriais - Por André Lopes, António Afonso, Ricardo Antunes.”
- Lopes, Sara Patrícia de Medeiros Lacerda. 2007. "Laboratório de Acesso Remoto Em Física.” Universidade de Coimbra.
- Lowe, D, P Newcombe, and B Stumpers. 2013. "Evaluation of the Use of Remote Laboratories for Secondary School Science Education.” *Research in Science Education* 43(3): 1197–1219.
- Ma, Jing, and Jeffrey V. Nickerson. 2006. "Hands-On, Simulated, and Remote Laboratories: A Comparative Literature Review.” *ACM Computing Surveys* 38(3): 7–es.
- Oliveira, Pedro César Bessa Magalhães. 2008. "Laboratório Remoto de Redes de Computadores.” Universidade do Porto.
- Palma, L. Brito, F. Vieira Coito, A Gomes Borracha, and J Francisco Martins. 2011. "A Platform to Support Remote Automation and Control Laboratories.”
- Pinto, João R. Caldas. 2004. *TÉCNICAS DE AUTOMAÇÃO*. Lidel - edições técnicas, Lda.
- Pires, J. Norberto. 2002. *AUTOMAÇÃO INDUSTRIAL*. Lidel - edições técnicas, Lda.
- Prudente, Francesco. 2007. *AUTOMAÇÃO INDUSTRIAL PLC: Teoria e Aplicações Curso Básico*. LTC - Livros Técnicos e Científicos Editora S.A.
- Restivo, M.T., J. Mendes, a.M. Lopes, C.M. Silva, and F. Chouzal. 2009. "A Remote Laboratory in Engineering Measurement.” *IEEE Transactions on Industrial Electronics* 56(12): 4836–43.

- Ribeiro, Miguel Francisco Amaral. 2012. "Interface and Web Server Implementation for an Industrial Automation Remote Laboratory." Universidade Técnica de Lisboa.
- Siemens. 2012. "S7-1200 Programmable Controller System Manual."
- Sousa, N., M. G. Gericota, and G. R. Alves. 2009. "UM LABORATÓRIO REMOTO, MULTIPLAS POTENCIALIDADES." In *II Jornada Luso-Brasileira de Ensino e Tecnologia em Engenharia*,.
- Villa-López, Farah Helúe, Jesús García-Guzmán, Jorge Vélez Enríquez, Simón Leal-Ortíz, and Alfredo Ramírez-Ramírez. 2013. "Electropneumatic System for Industrial Automation: A Remote Experiment Within a Web-Based Learning Environment." *Procedia Technology* 7: 198–207.
- W3Schools.com. 2013. "W3Schools." <http://w3schools.com/>.
- Yawcam. 2013. "Yawcam." <http://www.yawcam.com/>.