

Faculdade de Engenharia da Universidade do Porto



FEUP

**Telemetria e comunicação M2M aplicados a
serviços de melhoria da eficiência energética**

Ricardo Miguel Faria

Dissertação realizada no âmbito do
Mestrado Integrado em Engenharia Electrotécnica e de Computadores
Major Telecomunicações

Orientador: Prof. Doutor Cláudio Monteiro

Janeiro de 2010

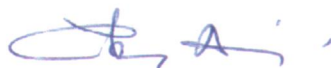
© Ricardo Faria, 2010

A Dissertação intitulada

“Telemetria e Comunicação M2M Aplicados a Serviços de Melhoria da Eficiência Energética”

foi aprovada em provas realizadas em 24 de Fevereiro de 2010


o júri



Presidente Professor Doutor Carlos Manuel Araújo Sá
Professor Auxiliar da Faculdade de Engenharia da Universidade do Porto



Professora Doutora Fernanda Oliveira Resende
Professora Auxiliar do Departamento de Ciências Naturais e Engenharias e Tecnologias da
Universidade Lusófona do Porto



Professor Doutor Cláudio Domingos Martins Monteiro
Professor Auxiliar do Departamento de Engenharia Electrotécnica e de Computadores da
Faculdade de Engenharia da Universidade do Porto

O autor declara que a presente dissertação (ou relatório de projecto) é da sua exclusiva autoria e foi escrita sem qualquer apoio externo não explicitamente autorizado. Os resultados, ideias, parágrafos, ou outros extractos tomados de ou inspirados em trabalhos de outros autores, e demais referências bibliográficas usadas, são correctamente citados.

Ricardo Miguel Pinto de Faria



Faculdade de Engenharia da Universidade do Porto

Resumo

O aumento do consumo energético nos sectores residenciais e de serviços, atribuído ao crescente número de equipamentos eléctricos, utilizados diariamente, mostra-nos que é imperativo utilizar a energia de forma racional. Um uso racional da energia limita os impactos negativos no meio ambiente.

A quantidade de equipamentos eléctricos presentes no nosso quotidiano, sempre ligados ou em standby, contribuem para um aumento do consumo energético, levando-nos a questionar onde gastamos tanta energia. Para determinar o consumo da energia são necessários equipamentos e ferramentas de fácil utilização, que permitam identificar, de forma clara, quais os equipamentos que estão a consumir mais energia e a que horas existe o consumo. Uma forma de sensibilizar os utilizadores é através do recurso a plataformas WEB, em que qualquer ocupante de uma habitação ou empresa possa visualizar o consumo actual e passado.

O objectivo desta dissertação foi o desenvolvimento de um sistema de comunicações e de serviços Web, aplicado a serviços de melhoria da eficiência energética. Para tal foi desenvolvido um sistema que permite a comunicação entre equipamentos de medição energética, que fazem leituras de consumos num determinado local, e um ponto de armazenamento de dados, localizado fora dessas instalações. Além das comunicações, foi desenvolvido uma plataforma Web para a visualização dos dados adquiridos, e um sistema de alarme por e-mail e sms, em caso de falhas.

Abstract

The increasing energy consumption in our homes and office, justified by the enormous quantity of electrical equipment that we use in a daily bases, show us that is imperative to start using energy in a rational way. Saving energy limits the negative impact caused in nature.

The amount of electric equipment in our homes, always on, or in standby, is rising the electric consumption, makes wondered where we spent all that energy. To determine power consumption, it is necessary the use of simple tools and equipments, that allow identified, clearly, which equipments are spending more energy and when. An easy way to do this is using a website, where the user could visualize the past and present consumption.

This dissertation aims to develop and improve a communication system and a web service, applied to energy efficiency services. For this, it was develop a system that allow the communication between the measure equipments, that makes consumption readings in one place, and a database, situated in a distinct location. Also, it was created a website to allow data visualization, and an alert system by e-mail or sms, to warn about system failures.

ÍNDICE

INTRODUÇÃO	1
1.1 MOTIVAÇÃO	1
1.2 PROBLEMA	3
1.3 OBJECTIVOS E ESTRATÉGIA	4
1.4 ESTRUTURA DA DISSERTAÇÃO	5
ESTADO DA ARTE	7
2.1 SISTEMAS DE INFORMAÇÃO ENERGÉTICA	7
2.2 EQUIPAMENTOS.....	8
2.2.1 TRANSFORMADORES DE INTENSIDADE E ANALISADORES.....	9
2.2.2 CONVERSORES	12
2.3 COMUNICAÇÕES	14
2.3.1 PROTOCOLO <i>MODBUS</i>	14
2.3.2 TCP <i>MODBUS</i>	18
2.4 VPN	19
2.4.1 SSL VPN	20
2.5 SCADA.....	23
2.6 CONCLUSÃO	27
ARQUITECTURA DO SISTEMA IMPLEMENTADO	29
3.1 CRIAÇÃO DA REDE VIRTUAL PRIVADA	29
3.2 SISTEMA DE RECOLHA DE DADOS	32
3.3 ENCAMINHAMENTOS	39
3.4 BASE DE DADOS	40
3.5 INTERFACE GRÁFICA.....	40
3.6 ALARMES	44
CASO PRÁTICO	47
CONCLUSÕES	63
5.1 RESULTADOS OBTIDOS	63
5.2 DESENVOLVIMENTO FUTURO	64
REFERÊNCIAS	65

Lista de figuras

Figura 1 - Medidas energéticas tomadas e fins alcançados.	2
Figura 2 - Modelo de desenvolvimento.	5
Figura 3 - Contadores eléctricos tradicionais.	9
Figura 4 - Transformador de Intensidade Circutor [7].	10
Figura 5 - Analisadores de energia com LCD Circutor [7] e ISA [8].	10
Figura 6 - Medidor de consumos de tomada [10].	11
Figura 7 - Conversor RS485 <-> Ethernet Circutor [11] e ISA [12].	12
Figura 8 - Ligação Computador <-> Medidor [13].	13
Figura 9 - Página de configuração do conversor [11].	14
Figura 10 - Modelo OSI.	15
Figura 11 - Camada Aplicação com Modbus, em [14].	16
Figura 12 - Trama Modbus [14].	16
Figura 13 - Ligação Cliente - Servidor.	17
Figura 14 - Pacote Modbus TCP [15].	19
Figura 15 - Túnel VPN [16].	20
Figura 16 - Modelo TCP [19].	21
Figura 17 - Encriptação por Chave Simétrica [16].	22
Figura 18 - Encriptação por Chave Assimétrica [16].	22
Figura 19 - Imagem de aplicação SCADA Circutor [22].	24
Figura 20 - Scada - Gestão de alarmes [21].	25
Figura 21 - Scada - Curvas de tendência.	25
Figura 22 - Scada - Registo de dados.	26
Figura 23 - Scada - Interface Gráfica.	26
Figura 24 - Ligação Servidor-Medidor.	29
Figura 26 - Interface para criar um CA.	30
Figura 25 - Aplicação Webmin.	30
Figura 27 - Interface para gerar Chave Servidor.	31
Figura 28 - Selecção do tipo de encriptação usada no túnel VPN.	31

Figura 29 - Rede túnel <i>VPN</i>	32
Figura 30 - Diagrama de funções desenvolvidas em Java.	33
Figura 31 - Base de Dados <i>MySQL</i> inicial.	40
Figura 32 - Gráfico criado usando <i>Flot</i> [27].	41
Figura 33 - Modem 3G <i>UMTS</i>	44
Figura 34 - Esquema do sistema instalado.	48
Figura 35 - Cliente <i>OpenVPN</i> <i>Windows</i>	51
Figura 36 - Esquema da base de dados utilizada.	52
Figura 37 - Página Web, Autenticação.	56
Figura 38 - Página Web, Período de amostragem.	56
Figura 39 - Página Web, Registos.	57
Figura 40 - Página Web, Editar Registos.	58
Figura 41 -Página Web, menu Dados.	59
Figura 42 - Página Web, escolha do periodo de amostragem.	59
Figura 43 - Página Web, escolha de dia e mês, respectivamente.	59
Figura 44 - Página Web, escolha dos dados de um ano completo.	59
Figura 45 - Página Web, escolha dos dados.	59
Figura 46 - Página Web, Tabela	60
Figura 47 - Página Web, Gráfico de Tensão.	61
Figura 48 - Página Web, gráfico Potência Activa.	61
Figura 49 - Página Web, gráfico Energia Activa.	62

Lista de tabelas

Tabela 1 - Códigos das funções Modbus.	17
Tabela 2 - Trama de pedido Modbus.	18
Tabela 3 - Trama de resposta Modbus.	18

Lista de abreviaturas e siglas

VPN	<i>Virtual Private Network</i>
M2M	<i>Machine-to-Machine</i>
LAN	<i>Local Area Network</i>
SGCIE	<i>Sistema de Gestão dos Consumos Intensivos de Energia</i>
LCD	<i>Liquid Crystal Display</i>
QGBT	<i>Quadro Geral de Baixa Tensão</i>
OSI	<i>Open Systems Interconnection</i>
PDU	<i>Protocol Data Unit</i>
ADU	<i>Application Data Unit</i>
CRC	<i>Cyclic redundancy check</i>
HEX	<i>Hexadecimal</i>
SSL	<i>Secure Socket Layer</i>
TLS	<i>Transport Layer Security</i>
IETF	<i>Internet Engineering Task Force</i>
AES	<i>Advanced Encryption Standard</i>
DES	<i>Data Encryption Standard</i>
3DES	<i>Triple DES</i>
CA	<i>Certificate Authority</i>
SO	<i>Sistema Operativo</i>
SCADA	<i>Supervisory Control and Data Acquisition</i>
DNS	<i>Domain Name System</i>
DMZ	<i>DeMilitarized Zone</i>
P2P	<i>Point-To-Point</i>
GPRS	<i>General Packet Radio Service</i>
UMTS	<i>Universal Mobile Telecommunications System</i>
3G	<i>Terceira Geração</i>
CMPB	<i>Câmara Municipal de Ponte da Barca</i>

Capítulo 1

Introdução

Este capítulo apresenta uma visão global do trabalho desenvolvido e reportado na presente dissertação. Após uma exposição do tema abordado, são descritos os objectivos e, por último, é apresentada a estrutura da dissertação.

1.1 Motivação

A dependência de energia eléctrica provoca enormes estragos no meio ambiente, contribuindo para o aquecimento global, aumento dos gases com efeito de estufa, destruição das florestas, entre outros problemas, pois muita da energia consumida actualmente provém de combustíveis fósseis.

A juntar ao problema ambiental, a actual legislação que regulamenta o SGCIE, no âmbito da Estratégia Nacional para a Energia, estabelece metas para reduções de consumo obrigatórias para algumas empresas, e que estas sejam alvo de auditorias energéticas feitas por empresas qualificadas. Para saber onde se pode utilizar a energia de forma eficiente, diminuindo o consumo global da empresa é necessário efectuar medições de consumos, de preferência em contínuo. Apenas desta forma é possível realizar auditorias com erros reduzidos e verificar se os planos de racionalização estão a ser cumpridos em cada sector e não apenas analisar a globalidade da instalação. Torna-se, então, crucial que seja feita uma gestão racionalizada dos consumos energéticos, e para isso é necessário saber onde é efectuado o maior gasto, onde se consome energia, por vezes sem se ter conhecimento, para que possamos tomar acções de forma a minimizar esse consumo.

Os Municípios são tipicamente grandes consumidores de energia com poder de intervenção no mercado e desempenham um papel importante na possibilidade de redução do consumo de energia. O desenvolvimento das suas competências incide sobre a gestão de instalações de edifícios municipais, dos serviços públicos como por exemplo os transportes, tratamento de resíduos sólidos urbanos, depuração de águas residuais, entre outros.

Um novo conceito municipal de desenvolvimento sustentável implica uma grande transformação na forma de ver, de pensar e de actuar em qualquer município. Para garantir a sustentabilidade de um Município é absolutamente vital o aproveitamento dos potenciais recursos energéticos (exemplo Biomassa, Energia Solar Térmica, Energia Solar Fotovoltaica e Energia Eólica)

É necessário saber tudo sobre a utilização de energia nos edifícios municipais e realizar diagnósticos que permitem a obtenção de ganhos energéticos consideráveis com repercussões ao nível económico, social e ambiental. Esses ganhos podem ser obtidos através da conservação de energia e da eficiência energética.



Figura 1 - Medidas energéticas tomadas e fins alcançados.

Estas melhorias devem ser incentivadas nos campos onde a maquinaria, os equipamentos e sistemas de recente desenvolvimento proporcionam rendimentos notoriamente mais elevados.

O desenvolvimento de um sistema de contabilidade energética é fundamental para um controlo efectivo dos consumos de energia e respectiva racionalização dos custos. Este sistema tem como principais funções detectar consumos anómalos, otimizar opções de tarifário e potência contratada e identificar medidas.

Para além da gestão energética dos edifícios, os Municípios devem também fazer uma gestão energética da iluminação pública, uma vez que esta é uma parcela importante da energia facturada. A iluminação pública desempenha um papel relevante, não só para o desenvolvimento da actividade económica de qualquer cidade, como também é um elemento de segurança e bem-estar.

Para que este sistema de contabilidade seja efectuado com sucesso, permitindo aos responsáveis tomarem decisões de que possam levar a melhorias, baseadas em factos exactos, é necessária a recolha de dados energéticos nas instalações para posterior análise, e para que tal aconteça é necessário instalar equipamento específico para efectuar essas medições,

software de gestão e suporte de comunicação entre os dois. Isto nem sempre é realizável ou efectuado de forma intuitiva.

1.2 Problema

As perdas energéticas nos edifícios habitacionais e de serviços, e o sem fim de equipamentos eléctricos, demonstram-nos que uma boa gestão dos mesmos permite diminuir de forma significativa os consumos. Para além de se pretender atingir esse objectivo, actualmente, as auditorias energéticas são uma obrigação para algumas empresas, com consumos superiores a 500 TEP, e, para realizar uma auditoria com pequenos erros, é necessário que exista um histórico dos consumos da empresa, preferencialmente um histórico dos consumos desagregados por sector, pois só assim se poderá perceber onde se registam os maiores gastos e onde deverão ocorrer mudanças de equipamentos, mas também de comportamentos. Contudo para existir armazenamento é necessário instalar sistemas de monitorização de energia na instalação.

A instalação destes sistemas por parte de consumidores domésticos ou empresas poderá ser acessível, mas mais importante do que a sua instalação, é a análise de dados. Uma análise cuidada, possibilita uma melhoria, de forma consciente, da eficiência energética do local a que nos propomos melhorar.

Para todo o processo se agilizar, é necessário o envio de dados para fora desses locais, até às instalações da empresa que fará a análise dos mesmos.

Numa instalação doméstica, o envio dos dados poderá ser mais fácil de realizar, pois os equipamentos de rede existentes serão mais simples, logo a sua gestão e respectiva configuração também, e a instalação de equipamentos extra, envio automático de dados por e-mail, entre outros métodos é mais exequível quando a rede onde pretendemos instalar o sistema não se encontra sobre um elevado protocolo de segurança.

Quando se fala de empresas, tal tarefa não será tão fácil, uma vez que estas terão *firewalls* a proteger a sua rede interna, não permitindo que equipamentos na sua *LAN* estejam acessíveis a partir do exterior. Outro problema que se verifica é, em muitas situações, a falta de um responsável pela rede de computadores já existente, criando assim dificuldades acrescidas na instalação dos equipamentos e envio de dados para fora da mesma.

A isto acresce a dificuldade que existe quando se pretende aumentar o número de equipamentos de medição e se pretende conjugar com equipamentos de outros fabricantes. A utilização de aplicativos de um fabricante, apesar de estes equipamentos usarem o protocolo *Modbus*, não garante por si só a compatibilidade para o mesmo *software*. Além disto, as aplicações são “fechadas”, ou seja, não é permitido acrescentar ou modificar as mesmas.

Situações como pretender usar a base de dados dessa aplicação para outra não é permitido. Como exemplo, caso exista um equipamento a armazenar dados nas instalações da empresa certificadora, e outro como *cópia de segurança* nas instalações do cliente, em caso de falha de comunicação, como falha na internet, p.e., obriga-nos a importar toda a base de dados do cliente para a empresa certificadora, onde apenas seria necessário obter os dados em falta.

As aplicações que os fabricantes fornecem têm as normais limitações de uma aplicação genérica, e como são fechadas, não permitem o acréscimo de funções a essa mesma aplicação. Isso faz com que a actualização da mesma não seja possível, tendo sempre que ser o fabricante a incluir novas características, o que não é uma solução viável.

Assim, é necessário obter uma solução que permita uma fácil e transparente forma de comunicação para o envio de dados entre os sistemas que efectuem medições, e o local onde os dados serão armazenados para posterior análise, sem que para tal seja necessário fazer alterações nas redes dos locais onde os equipamentos serão instalados.

1.3 Objectivos e Estratégia

O objectivo deste trabalho é permitir a integração de variados sistemas, facilitando a aquisição dos dados provenientes dos equipamentos de medição, tornando a sua captura transparente para quem faça a instalação dos mesmos, para quem irá fazer a sua análise e/ou para o gestor da rede interna do local.

Será desenvolvido um sistema que permita a ligação entre a camada de medição, onde estarão os equipamentos que efectuem a medição dos dados energéticos, e a de aplicação, onde estará o *software* para análise de dados, como indica a Figura 2, passando também pela base de dados, onde serão armazenados os dados recolhidos. Esta base de dados estará disponível para aplicações de terceiros ou posteriormente desenvolvidas.

Na aplicação, pretende-se criar uma plataforma Web, onde seja possível configurar quais os dados que se pretende recolher, a visualização dos mesmos, com gráficos para facilitar a sua análise, e um sistema de aviso ao cliente, sempre que algum valor não esteja dentro dos parâmetros habituais. Esta plataforma estará disponível, quer para o cliente, quer para o prestador do serviço, podendo ambos alterar os parâmetros de medição.

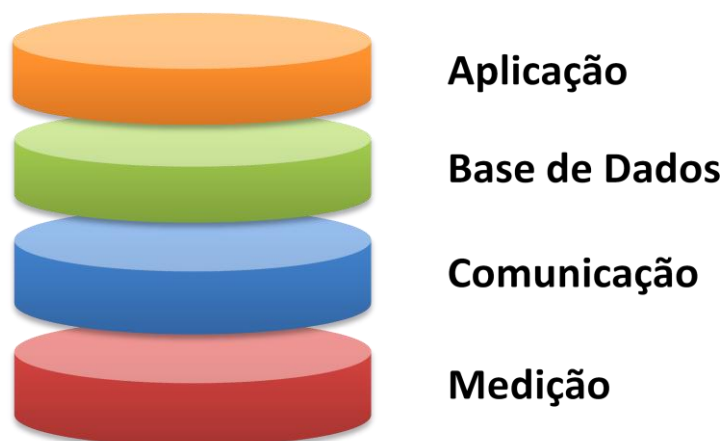


Figura 2 - Modelo de desenvolvimento.

1.4 Estrutura da dissertação

A presente dissertação encontra-se organizada em 5 capítulos.

No 1º capítulo é descrito o que levou ao desenvolvimento desta dissertação, e os objectivos pretendidos com a mesma.

O Capítulo 2 tem como objectivo fazer uma introdução às tecnologias existentes no mercado e o seu funcionamento. Faz também referência aos protocolos de comunicação usados nessas mesmas tecnologias.

O Capítulo 3 descreve a solução implementada como forma de resolver os problemas referidos neste capítulo, desde a comunicação ponto-a-ponto, até à recolha de dados dos equipamentos instalados.

O Capítulo 4, Implementação, apresenta o caso prático desenvolvido na presente dissertação.

O Capítulo 5, Conclusões, apresenta uma síntese do trabalho desenvolvido e uma proposta de melhoria, para trabalhos futuros, que podem ser efectuados com base nesta dissertação.

Capítulo 2

Estado da Arte

Neste capítulo é realizada uma breve descrição sobre os equipamentos, serviços fornecidos e o protocolo usado para a comunicação, estudados para a realização do presente trabalho.

2.1 Sistemas de Informação Energética

Sistemas de informação energética são sistemas que utilizam software, equipamentos de aquisição de dados e sistemas de comunicação com o intuito de fornecer informação aos administradores do departamento energético, financeiro, entre outros.

Os dados obtidos incluem normalmente consumos de energia, características dos edifícios, dados dos sistemas dos edifícios, como aquecimento, ventilação e ar condicionado, AVAC, e sistemas de iluminação. Incluem também um tarifário energético.

Estes sistemas começaram a ser desenvolvidos no início dos anos 90, segundo [1], e têm desde então sofrido avanços para incluir diversos sistemas de comunicação e de operatividade.

Nos dias de hoje, segundo [2] e [3], estes sistemas são usados por empresas, empresas de serviços energéticos, e na indústria. Assim, é possível gerir pontos como:

- Períodos e magnitude dos picos de consumo energético;
- Balanço de carga energética diária;
- Histórico do uso energético;
- Custos de acordo com o período de consumo.

A actualização em tempo real ou diária do consumo energético, permite avaliar problemas no sistema eléctrico de um edifício, que antes era complicado verificar, segundo

[4] e [5]. Como os dados podem ser visualizados quase instantaneamente, as empresas podem conseguir fazer alterações imediatas no funcionamento das mesmas, e verificar se essas alterações permitem melhorias no sistema energético.

Estes sistemas também podem permitir, segundo [6], criar relatórios de acordo com as necessidades dos diferentes utilizadores, e poderá estar ligado a um sistema que permita controlar e monitorizar os parâmetros de qualidade do ar interior, permitindo desta forma fazer uma gestão global e centralizada de todos os parâmetros.

Para que estes sistemas funcionem correctamente em ambientes com vários edifícios ou contadores eléctricos separados, é necessário recorrer também às facturas eléctricas, para verificar inicialmente quais os pontos onde o consumo energético é maior, e começar a partir desse ponto.,

Num caso prático, para situar a funcionalidade de uma aplicação deste género, a Câmara Municipal de Ponte da Barca, CMPB, tem ao seu encargo 187 instalações de consumo (ano 2007). Estas instalações estão agrupadas consoante o seu nível de tensão:

- BTE: Baixa tensão especial (4 instalações de consumo - grandes equipamentos);
- BTN: Baixa tensão normal (183 instalações de consumo).

Relativamente às instalações de BTN estas estão divididas de acordo com função que desempenham:

- Instalações e Serviços (25 instalações);
- Escolas (47 instalações);
- Elevações de Água (16 instalações);
- Iluminação Pública (95 instalações).

Como qualquer câmara, existe um consumo elevado com a iluminação pública, consumo este que se deve, em geral, ao tipo de desenho da iluminação e à antiguidade de muitas instalações. Isto origina um aproveitamento pouco eficiente da energia luminosa produzida pelos elementos e sistemas de iluminação e a elevados níveis de poluição luminosa. Por isso, e em muitos casos também devido ao aumento do consumo com o crescimento urbano e a ampliação de pontos de luz na rede, um sistema destes permite aos Municípios realizar diagnósticos às suas instalações de iluminação com o fim de reduzir o seu consumo, diminuir o seu custo económico, facilitar a sua manutenção e dar melhores serviços aos seus cidadãos.

2.2 Equipamentos

A recolha de dados energéticos já não pode ser realizada apenas pelos actuais contadores de consumo que estamos habituados a ver nas entradas das nossas habitações, Figura 3.

Estes contadores apenas apresentam os consumos e não permitem fazer as medições distribuídas pelas fases, medir potências, energias, etc. É, então, necessário recorrer a equipamentos mais evoluídos, que nos permitam recolher todas as informações para que seja feita uma análise da distribuição de consumos, desequilíbrio nas fases, no caso de instalações trifásicas, entre outros parâmetros. A descrição destes equipamentos é apresentada no ponto 2.1.1 deste capítulo.



Figura 3 - Contadores eléctricos tradicionais.

Para a comunicação entre os diversos medidores, é habitual recorrer a conversores *Ethernet* <-> RS485, porta de comunicações mais comum nestes equipamentos.

Estes conversores permitem a ligação dos medidores directamente na rede informática local. No ponto 2.1.2 é descrito o funcionamento destes conversores.

2.2.1 Transformadores de Intensidade e Analisadores

O uso de analisadores de consumos eléctricos permite obter informação detalhada do que se passa em termos energéticos e de forma específica, ou seja, num determinado ponto do circuito eléctrico que se pretende investigar.

Estes equipamentos, em [7] e [8], contêm no seu interior TI, que permitem medir uma diversidade de parâmetros, desde tensões a energias, passando por potências, entre outros, em redes trifásicas industriais e domésticas, quer equilibradas, quer desequilibradas.

A medida destes equipamentos realiza-se em verdadeiro valor eficaz, através de três entradas de tensão C.A. e três entradas de intensidade C.A., através de transformadores de corrente.

Estes instrumentos podem ser montados em qualquer ponto do circuito, podendo ser incluídos no quadro eléctrico.

Existem modelos mais simples, como o da Figura 4 -, que se ligam a um computador através de RS485 ou de um conversor *Ethernet* <-> RS485, descrito posteriormente, e que,

após pedidos de valores de medição por tramas *Modbus* provenientes do computador ao qual estão conectados, retornam uma trama com a resposta ao pedido.



Figura 4 - Transformador de Intensidade Circutor [7].

Para situações onde seja necessário uma visualização rápida das medidas, sem que seja necessário aceder a um computador para tal, existem equipamentos que incluem *LCDs*, Figura 5, apresentando os dados instantâneos que o aparelho está a medir através dos TI internos. Estes equipamentos são habitualmente instalados nos QGBT e quadros principais, para mais fácil leitura dos dados.



Figura 5 - Analisadores de energia com LCD Circutor [7] e ISA [8].

Os equipamentos mais simples apenas medem os parâmetros que lhe são solicitados, ou seja, quando ligados a um computador que tenha um *SCADA*, ou outro tipo de aplicação de recolha de dados, configurado para pedir dados a estes equipamentos. Sempre que seja pretendido obter uma determinada informação, o computador envia um comando *Modbus* com o registo que pretender obter, tensão na fase 1, p.e., e o aparelho responde com o valor medido.

Os mais complexos permitem, além da apresentação instantânea dos parâmetros que se encontra a medir, armazenar dados, como o máximo e mínimo valor da tensão na fase 1, e a energia consumida desde que o mesmo foi ligado, sem a necessidade de pedidos de leitura.

A comunicação com estes equipamentos de medição, quer seja um TI ou um analisador, é realizada através de RS485, o que garante, em [9], uma distância máxima entre pontos de

1200m, permitindo ter estes medidores em pontos distantes do ponto onde colocamos o computador com o software que irá, através de *Modbus*, efectuar os pedidos aos mesmos. Para distinguir cada um dos equipamentos, é atribuído, um endereço entre 1 até 255. Como as portas RS485 que existem nos computadores, ou os conversores, referidos no ponto seguinte, não permitem este número de equipamentos ligados à mesma porta/conversor, não há o risco de esgotar a *pool* de endereços que possam ser atribuídos, sendo por isso necessária uma nova porta ou conversor.

Além deste tipo de analisadores, que habitualmente são utilizados para medir circuitos, que poderá ser um circuito geral, ou um parcial, existem equipamentos específicos para medir consumos em tomadas eléctricas, como o da Figura 6. Estes equipamentos permitem o conhecimento do consumo de um equipamento eléctrico específico, ou conjunto de equipamentos, que estejam ligados directamente neste medidor. Tal como os anteriores, medem além dos consumos, valores de tensão, corrente, potência, frequência e ângulo de fase.



Figura 6 - Medidor de consumos de tomada [10].

Ao contrário dos anteriores, estes equipamentos funcionam via *Wireless*, não necessitando de cabos para a comunicação. É utilizado *Bluetooth* ou *ZigBee* para a transmissão de dados, o que obriga a que os aparelhos que efectuem a recolha da informação proveniente destes sejam munidos de receptores para estes protocolos de comunicação. Isto, além dos gastos acrescidos que traz, implica que os receptores estejam a distâncias curtas do medidor, pois o alcance é muito reduzido, e, como funciona por radiofrequência, existe sempre a

possibilidade de interferências com outros equipamentos. Num ambiente industrial, esta situação poderá acarretar muitos problemas.

2.2.2 Conversores

Para facilitar a colocação dos equipamentos de medição em instalações mais complexas, normalmente utiliza-se conversores *Ethernet* <-> RS485, ver [11] [12]. Estes, Figura 7, permitem o uso da actual rede informática para ligação dos medidores a um computador que se encontrem em localizações diferentes.



Figura 7 - Conversor RS485 <-> Ethernet Circutor [11] e ISA [12].

Os conversores, em [11] e [12], possibilitam a conversão entre *Modbus/TCP* <-> *Modbus ASCII/RTU*. Isto permite enviar comando *Modbus* através da rede existente, e ligar os medidores à porta RS485 do mesmo, Figura 8, sem que seja necessário que o computador tenha portas RS485, e sem ter que passar novos cabos para ligar os medidores ao computador, situação que em ambientes industriais, tornar-se-ia incomportável.

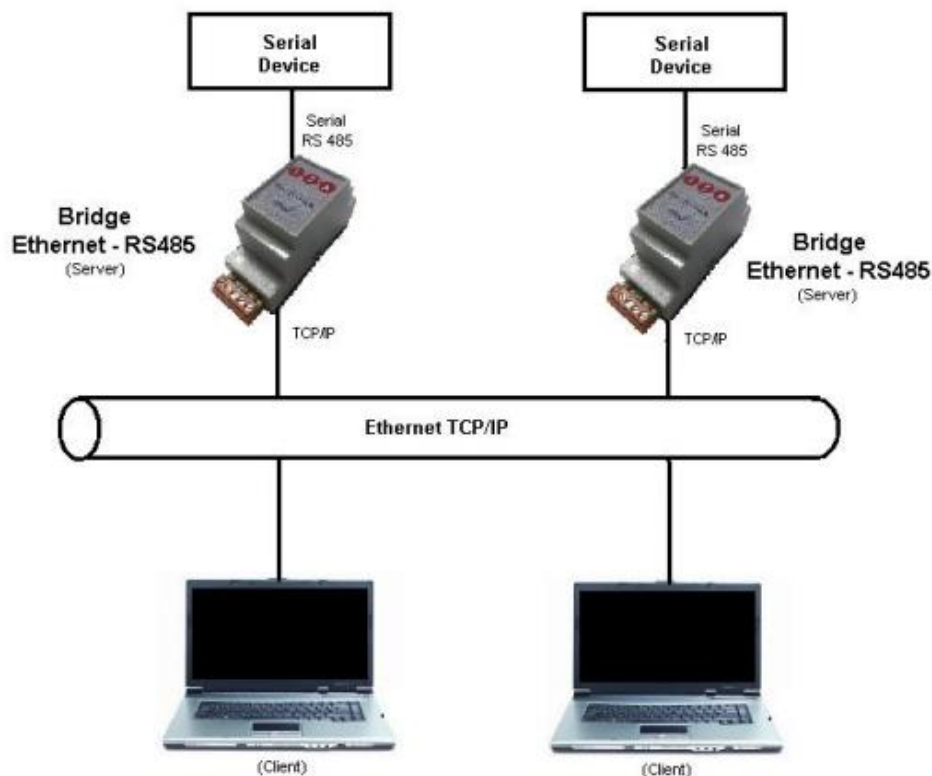


Figura 8 - Ligação Computador <-> Medidor [13].

Para comunicar com um equipamento de medição, tal como indicado na figura anterior, um computador envia para o *IP* do conversor e respectivo porto no qual o conversor está à escuta, lembrando que é uma ligação *TCP*, o pacote *TCP Modbus*. Após a recepção do mesmo, o conversor remove os cabeçalhos *IP* do pacote, ficando apenas com a trama *Modbus*. Estes protocolos são descritos no ponto 2 deste capítulo.

Os conversores são de fácil configuração, através de uma página Web, Figura 3 Figura 9, onde se pode configurar, entre outros parâmetros, a velocidade de ligação aos medidores e o protocolo de comunicação usado.

Panel de configuración

Firmware

Producto: Círculo Industrial Converter TCP2RS
Nombre: XPORT Ethernet Modbus-TCP
Versión de firmware: 2.20
Dirección hardware: 00-20-4A-8D-78-A1

Parámetros de red

Dirección IP: 192.168.3.12
Máscara de subred: 255.255.0.0
Puerta de enlace: 0.0.0.0

Contraseña

Cambiar contraseña
Contraseña:
Repite contraseña:

Parámetros de línea

Velocidad: 19200
Tamaño del byte: 8
Paridad: Ninguna
Bits de stop: 1
Protocolo serie: Full-Duplex

Parámetros Modbus/TCP

Protocolo serie modbus: ModbusRTU
Timeout de carácter: 50
Timeout de mensaje: 500
Dispositivo adjunto: Esclavo

Lista de IP's adjuntas:

IP	Dirección
----	-----------

Añadir Eliminar

Enviar configuración

Figura 9 - Página de configuração do conversor [11].

2.3 Comunicações

O Protocolo utilizado para comunicar com os medidores é o *Modbus*. Este protocolo e a sua variante TCP são descritos nos pontos seguintes.

2.3.1 Protocolo *Modbus*

O protocolo *Modbus* é um protocolo de comunicação de dados habitualmente utilizado em sistemas de automação industrial.

Desenvolvido no final da década de 70, é usado para aquisição e actualização de dados de controladores lógicos programáveis.

Como descrito em [14], este protocolo, faz parte no nível 7, camada Aplicação, do modelo *OSI*, Figura 10, camada esta que faz a ligação entre a aplicação que pediu uma determinada informação, e o protocolo de comunicação que é utilizado para esse mesmo pedido, permitindo a comunicação cliente/servidor entre equipamentos localizados em diferentes redes ou diferentes tipos de ligação.

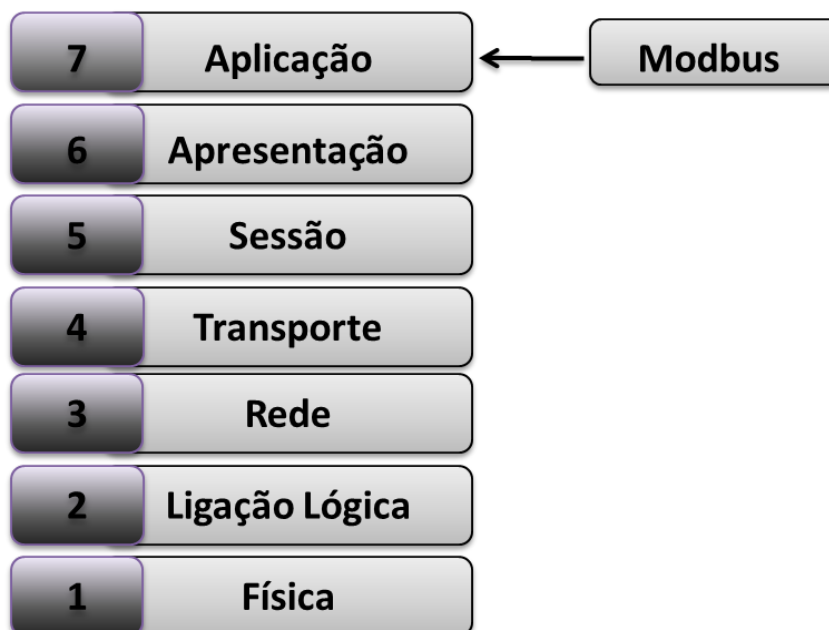


Figura 10 - Modelo OSI.

Este protocolo foi desenvolvido para um sistema pedido/resposta, no qual a aplicação efectua um pedido de um determinado parâmetro a um equipamento, e aguarda pela resposta do mesmo. Existem tipos diferentes de pedidos, desde pedidos de leitura e de escrita, que serão descritos posteriormente.

Actualmente, o protocolo *Modbus* encontra-se implementado para comunicação, entre outros, através de Ethernet/802.3, RS-232, RS485, como ilustrado na Figura 11.

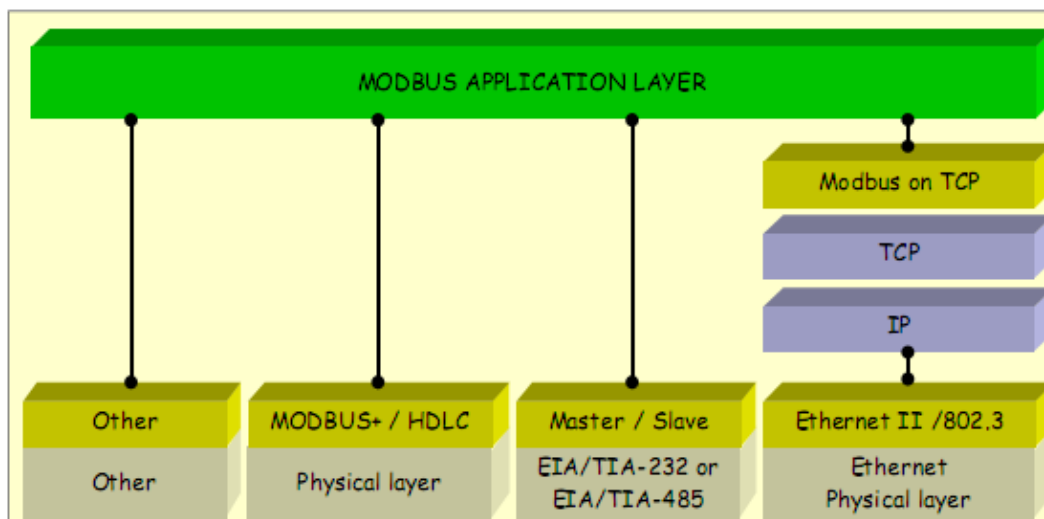


Figura 11 - Camada Aplicação com Modbus, em [14].

Inicialmente, o protocolo *Modbus* cria uma trama, *PDU*, que é completamente independente da camada de rede utilizada para a comunicação. Os restantes campos são posteriormente adicionados de acordo com o tipo de comunicação usado, TCP, etc., ficando assim completa a trama a enviar, *ADU*, como ilustrado na Figura 12.

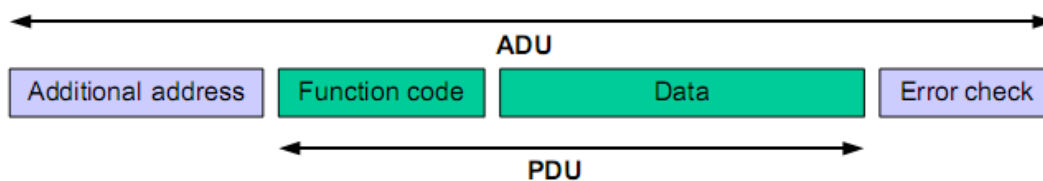


Figura 12 - Trama Modbus [14].

Esta trama é construída pelo cliente que inicia uma transacção, de acordo com a Figura 13, e envia-a para o servidor. A noção de cliente/servidor deve ser pensada da seguinte forma: o cliente é o computador que pretende algo, e o servidor é o equipamento de medição que detém os dados que o cliente pretende.

O código da função que o cliente introduz na trama indica o tipo de pedido que ele pretende, que pode ser de leitura ou de escrita, dos quais os mais usuais encontram-se descritos na Tabela 1 - Códigos das funções Modbus.

Estes códigos permitem o acesso a dados que os aparelhos de medição tenham no seu registo. Estes dados podem ser booleanos, no caso de entradas, quer estejam activas, quer desactivas, ou registos mais complexos, que contêm valores, no caso dos medidores, que poderão ser dados de energia consumida, p.e.

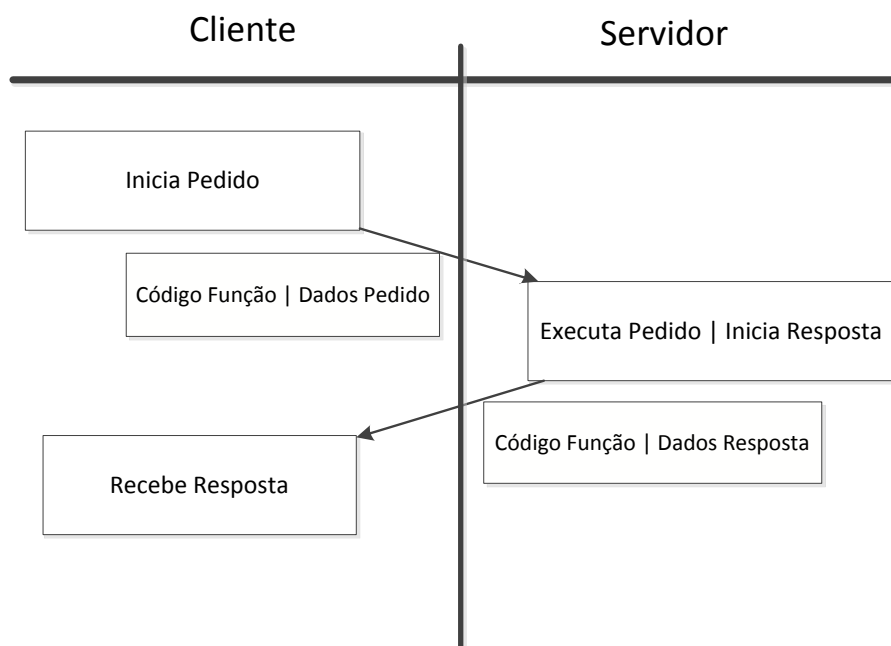


Figura 13 - Ligação Cliente - Servidor.

Tabela 1 - Códigos das funções Modbus.

Código	Acesso	Descrição
01	Leitura	Obter estado de saídas discretas
02	Leitura/Escrita	Obter estado de entradas discretas
03	Leitura	Obter dados de registos de memória ou saídas analógicas
04	Leitura/Escrita	Obter dados de registos das entradas analógicas

Estas funções permitem o acesso aos registos de memória dos equipamentos com que estamos a comunicar, memória essa que pode ser de leitura e/ou escrita, e que pode conter apenas informação *booleana*, para o caso de querermos saber se uma saída está activa ou não, p.e., ou dados concretos, como uma leitura de um valor.

A trama contém diversos parâmetros, que são preenchidos com *bits*, *bytes* (8 bits) ou *words* (16 bits), em sistema hexadecimal. Estes parâmetros são, no caso da trama enviada do cliente para o servidor, 1 *byte* com o endereço do equipamento com o qual estamos a tentar comunicar, outro com a função que pretendemos, Tabela 1, 2 *bytes*, ou uma *word*, com o valor do primeiro registo que pretendemos, outra *word* com a quantidade de registos pretendidos e, finalmente, o campo de CRC, para controlo de erros.

Um exemplo de uma situação pergunta/resposta pode ser verificada na Tabela 2 e Tabela 3, respectivamente.

Tabela 2 - Trama de pedido Modbus.

Endereço	Função	Registo Inicial		Quantidade Registos		CRC Checksum	
64	03	00	0A	00	03	2C	3C

Neste pedido, tudo em valores hexadecimais, é pretendido que o equipamento que está configurado com o número 64 nos informe dos valores que tem nos registos de memória, função 03, começando no registo 000A. Os registos de memória, definidos por esta função, iniciam no número 40 000, mas como a função vai na trama, a referência 4xxxx é implícita.

A quantidade de registos, tal como o nome indica, informa o equipamento da quantidade de registos de que pretendemos obter informação, neste caso, 3 registos, desde o 00 0A até ao 00 0C. O CRC permite testar se não houve corrupção de dados durante o transporte.

Tabela 3 - Trama de resposta Modbus.

Endereço	Função	Nº Bytes	Registo 1		Registo 2		Registo 3		CRC Checksum	
64	03	06	2E	CE	2E	E8	2F	13	0D	58

A resposta do equipamento apresenta-nos novamente o endereço do próprio, para garantir a origem da trama. A função diz que tipo de dados são enviados, sendo neste caso, registos de armazenamento. O número de *bytes* informa quantos registos foram lidos, sendo 2 *bytes* por registo, o que significa que estamos a receber dados de 3 registos. O conteúdo dos registos é apresentado, por ordem crescente, desde o registo mais baixo, até ao mais elevado.

2.3.2 TCP Modbus

O protocolo *Modbus* foi definido para sistemas mais antigos e, com a actualização das redes de computadores, também se tornou fulcral “actualizar” o protocolo para que se pudesse adequar às mesmas e a sistemas mais rápidos e funcionais.

A actualização efectuada não fez mais que encapsular a actual trama *Modbus* existente, tornando-a um pacote IP, para uso em redes *TCP/IP*, como apresentado na Figura 14 e referido em [15].

Este tipo de encapsulamento é usado nos conversores, como o referido anteriormente, para comunicar com equipamentos que possuam portas de comunicação *RS485*, sendo que os mesmos também farão o contrário, ou seja, remover o encapsulamento IP, e enviar a trama *Modbus* para o equipamento que vem referido nessa mesma trama.

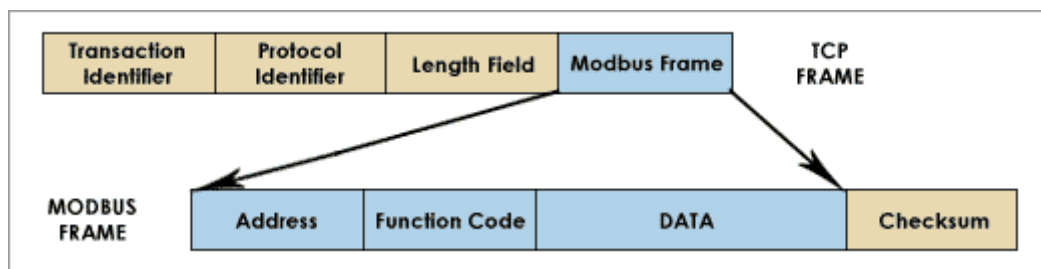


Figura 14 - Pacote Modbus TCP [15].

No pacote enviado, é colocado o endereço IP do conversor, permitindo assim que o pacote circule pela rede até ao ponto de destino, podendo ser entregue sem as restrições de uma ligação *RS485*, limitada pela distância e pela ligação directa entre os dois pontos, o que pede e o que detêm a informação.

2.4 VPN

Para estabelecermos ligações entre dois pontos geograficamente distintos, p.e., uma empresa que tenha sede no Porto e uma sucursal em Lisboa, mas que se pretende que estejam numa rede de computadores comum, recorre-se a ligações *VPN*. Estas ligações permitem que os pontos comuniquem entre si, como se da mesma rede física se tratasse, Figura 15.

Com o uso de *VPNs*, além do referido anteriormente, garante-se também confidencialidade nos dados trocados entre os pontos, pois são usados protocolos criptográficos que fornecem confidencialidade, autenticação e integridade necessários.

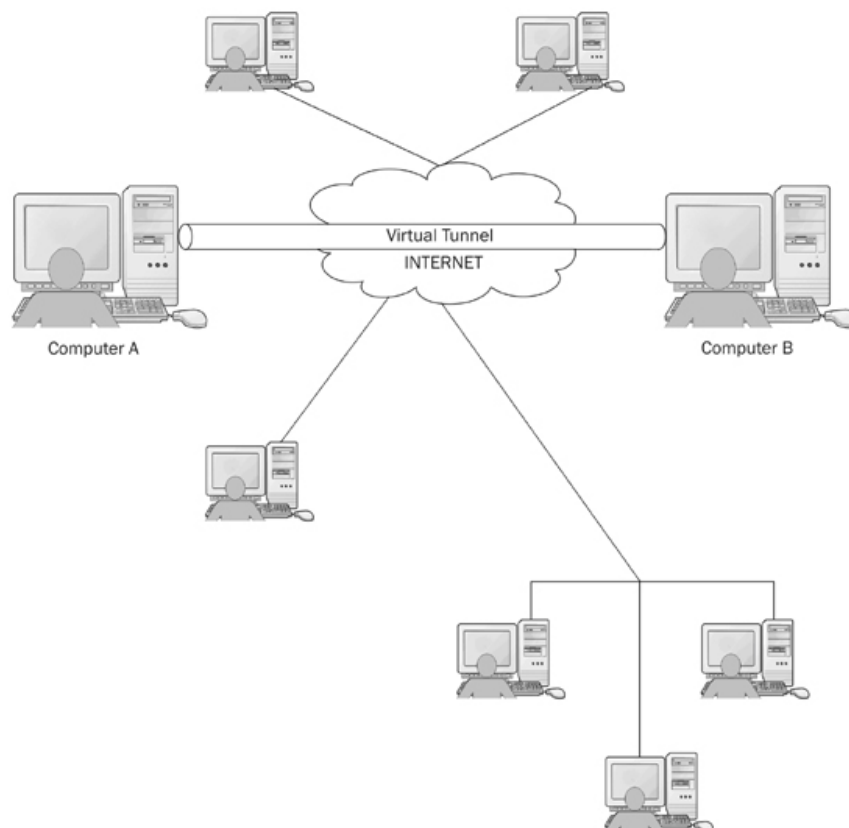


Figura 15 - Túnel VPN [16].

Este tipo de sistema funciona baseado na ideia de túneis, ou seja, é criado um túnel virtual entre os dois pontos da *VPN*, e todos os pacotes que circulam entre os dois pontos têm um *label* que os distingue, permitindo o encaminhamento através de *routers*, garantindo a entrega no sítio correcto.

Para acesso à *VPN*, é necessário, em [17], a autenticação do utilizador que, dependendo do tipo de protocolo escolhido, pode ser desde um conjunto *username/password*, até a um conjunto de chaves criptográficas.

Existem vários tipos de protocolos usados para criar túneis *VPN*, sendo que um dos mais usados é por *SSL*, pois não requer mais do que dois computadores, nem exige a compra de equipamento extra, mas garante, mesmo assim, um elevado grau de segurança e rapidez. Este protocolo é descrito seguidamente.

2.4.1 SSL VPN

Para se estabelecer ligações *VPN* mais económicas, simples de configurar, mas que mantivessem o mesmo grau de segurança, desenvolveu-se um sistema que funcionasse sobre *SSL*.

Utiliza-se o uso de *SSL* nos sites de internet que habitualmente usamos, como *e-banking*, o *e-mail*, entre outros, em que o endereço se inicia com *https*. O princípio é o mesmo, ou seja, troca e dados confidenciais mas que, em vez de ser utilizado em aplicações, como o *browser*, é aplicada a redes. A ligação é estabelecida entre dois computadores, que funcionam como *gateways*, e todo o tráfego é direccionado por esses pontos.

Segundo Charlie Hosner, em [18], «OpenVPN é uma *VPN* que funciona em *user-space*, e que actualmente é conhecida também por *TLS* ou *TLS/SSL*, desde que o IETF reclamou para si o desenvolvimento e gestão do *SSL*». *TLS/SSL*, conforme apresentado na Figura 16 e descrito em [19], pertence à camada transporte, o que permite fornecer segurança indirectamente às aplicações, ou seja, como as aplicações correm sob a camada de transporte, todos os dados que passam para a camada de rede provenientes da camada de aplicação, encontram-se automaticamente encriptados, desde que a aplicação esteja devidamente configurada para utilizar *TLS/SSL*.

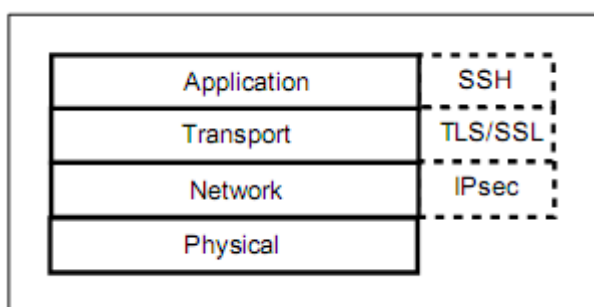


Figura 16 - Modelo TCP [19].

Actualmente, e também segundo Charlie Hosner, «*TLS/SSL* é o protocolo de segurança mais usado em todo o mundo, e, como tal, é constantemente alvo de ataques para testar a sua robustez». Até ao momento não é conhecida forma de, em tempo útil, quebrar este sistema de segurança. Entende-se tempo útil como, p.e., o período de tempo que o certificado *SSL* usado se mantenha com validade, sem ser revogado.

Para a criação do túnel *VPN*, é necessário o uso de encriptação. Esta encriptação poderá ser simétrica ou assimétrica.

Segundo Joseph Steinberg, entre outros em [16], no caso simétrico, os dois pontos terminais do túnel partilham chaves de encriptação e desencriptação comuns, e usam essas mesmas chaves para encriptar todo o tráfego que circula entre os pontos, como se pode observar na Figura 17. Os dados que saem do terminal inicial são encriptados com a chave privada, sendo os mesmos enviados para o destinatário. Na recepção, a mesma chave é usada para desencriptar os dados enviados.

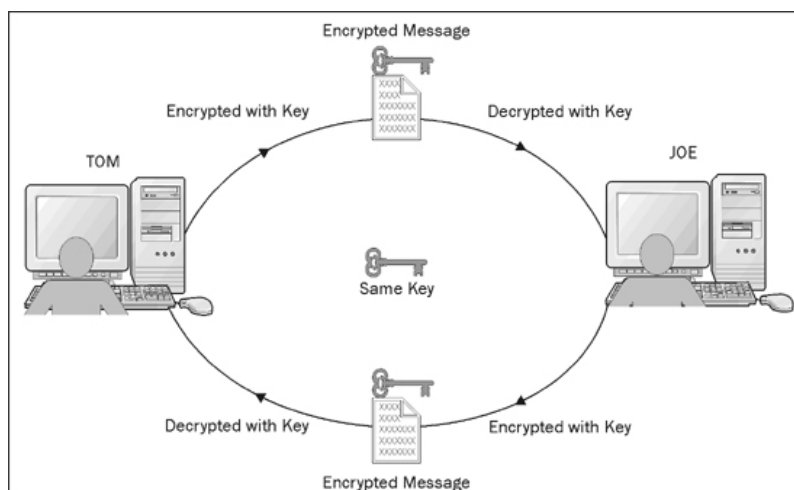


Figura 17 - Encriptação por Chave Simétrica [16].

No caso assimétrico, é criado um conjunto de chaves pública/privada. Cada um dos terminais contém a chave pública do outro, e a sua própria chave privada. Aquando do estabelecimento da ligação, o terminal que vai enviar o pacote encripta-o com a chave pública do outro terminal. Como apenas o receptor tem a chave privada, apenas ele pode descriptar a mensagem, como é apresentado na Figura 18.

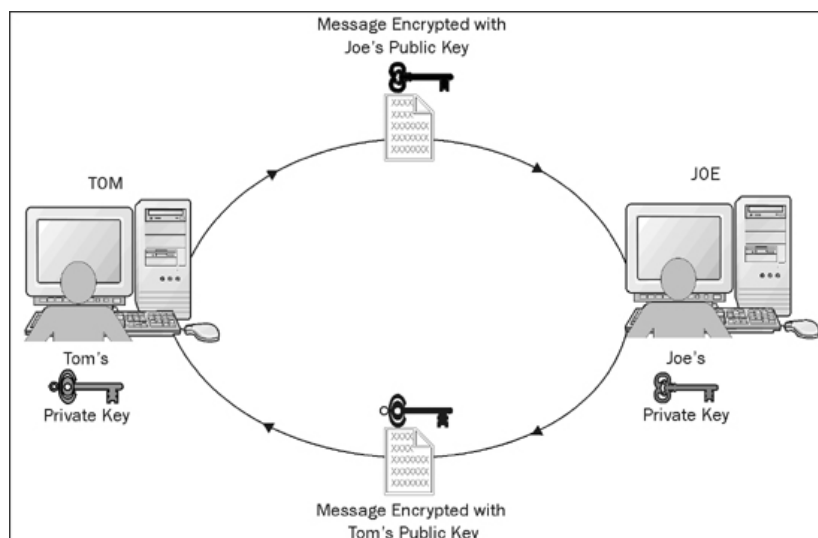


Figura 18 - Encriptação por Chave Assimétrica [16].

Para este sistema funcionar, é necessário a troca das chaves entre servidor/cliente, ou, entre as duas pontas do túnel que se pretende fazer. A atribuição das chaves ao terminal correcto tem que ser garantida, pois não queremos que a chave seja dada a outra “pessoa”. Para o correcto funcionamento deste sistema, este ponto é fulcral, pois uma chave “perdida” permite a outro utilizador entrar na rede que queremos partilhar e manter segura.

Prática usual para a troca da chave é o recurso a uma CA. Uma CA é uma entidade fidedigna que, segundo William Stallings em [20], emite certificados de autenticidade e,

habitualmente, listas de certificados revogados, ou seja, certificados que, por algum motivo, foram considerados inválidos. Este motivo é, normalmente, por pedido do dono do mesmo, que pretende utilizar um novo certificado e, como tal, pretende deixar de utilizar o já existente. Os certificados emitidos pela CA contém a chave pública e a identidade do dono do certificado. Assim, o CA garante ao utilizador, que está a usar a chave pública de outra pessoa, que aquela chave é mesmo dessa pessoa, e não de outra se está a fazer passar por ela, tendo então garantia de autenticidade da chave.

Chaves simétricas ou assimétricas são usadas com bastante rapidez e são criadas usando algoritmos sólidos, como *AES*, *DES* ou *3DES*.

Conforme referido anteriormente, *TLS/SSL VPNs* operam em *user-space*. Desta forma, não necessitam de ligações complexas com o kernel do *SO* sobre o qual estão a operar. Normalmente, para usar encriptação na camada de rede, é necessário recurso ao *kernel*, mas neste caso esta questão foi resolvida de forma mais simples, através do uso de “interfaces virtuais”. Estas interfaces funcionam de forma transparente para o utilizador, ou seja, como se fosse uma interface de rede física. Assim, todo o tráfego é redireccionado por essa porta virtual, e encriptado e só depois segue para a camada física do kernel.

Com esta solução, basta configurar um servidor de *VPN* num determinado local, e instalar os clientes nas máquinas que se querem ligar à rede que esse servidor se encontra a partilhar. Esses clientes podem ser fixos ou móveis, também conhecidos por *road warriors*.

2.5 Scada

Os sistemas *SCADA* foram desenvolvidos para permitir informar o estado do processo industrial, sendo este estado actualizado periodicamente. Inicialmente, era apenas possível monitorizar sinais representativos de medidas e estados de dispositivos, através de indicadores e lâmpadas, sem recurso a uma aplicação que permitisse a interoperabilidade com o utilizador, como refere [21]

Os computadores assumiram um papel preponderante, com a evolução tecnológica, na recolha e posterior tratamento de dados, permitindo criar sistemas complexos de controlo e visualização dos mesmos e dos dados por eles gerados ou obtidos.

Os sistemas *SCADA* foram actualizados e, actualmente, utilizam sistemas de comunicação e computação que permitem automatizar controlo e monitorização dos processos industriais, efectuando recolha de informação em ambientes complexos e apresentando essa mesma informação de forma *user-friendly*, Figura 19, ao utilizador, com recurso a interfaces Homem-Máquina.

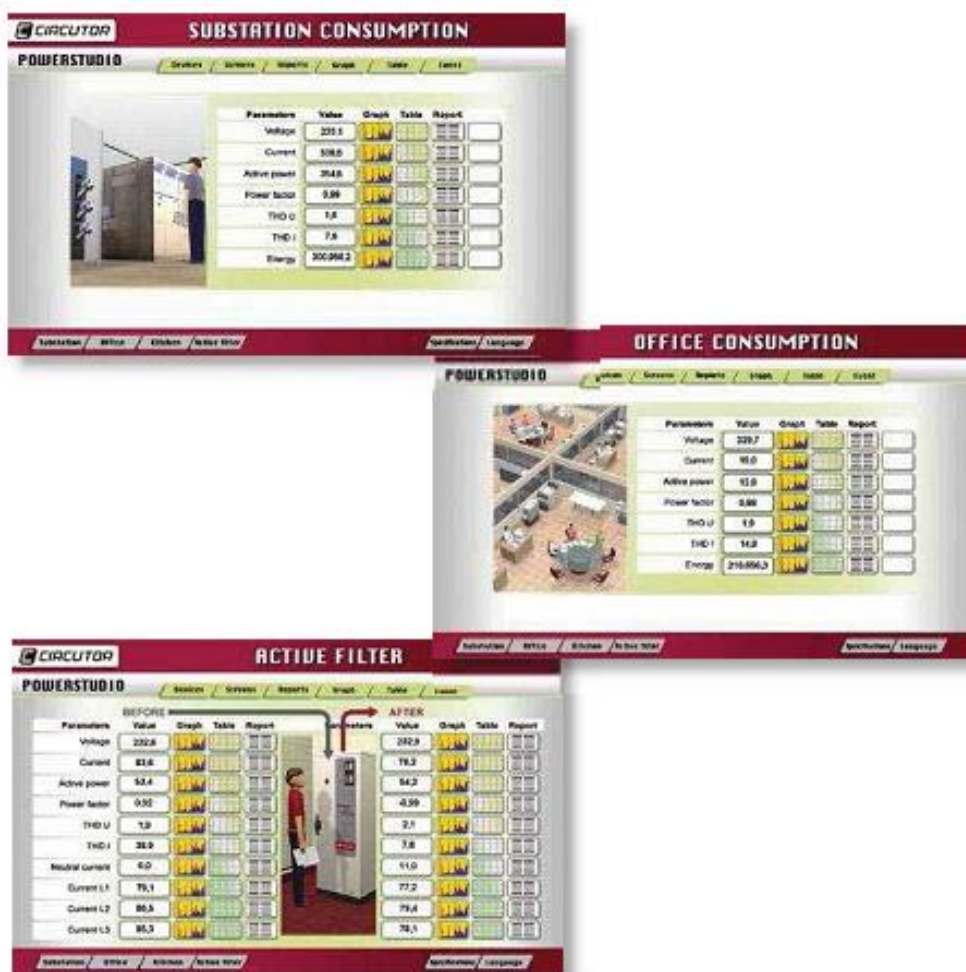


Figura 19 - Imagem de aplicação SCADA Circutor [22].

Estes sistemas abrangem aplicações cada vez mais diversificadas, podendo estar presentes em diversas áreas, tais como a indústria de celulose, a indústria petrolífera, a indústria têxtil, a indústria metalúrgica, a indústria automóvel, a indústria electrónica, controlo de AVAC em centros comerciais, entre outras.

Estes sistemas revelam-se de crucial importância na estrutura de gestão das empresas, facto pelo qual deixaram de ser vistos como meras ferramentas operacionais, ou de engenharia, e passaram a ser considerados como uma importante fonte de informação.

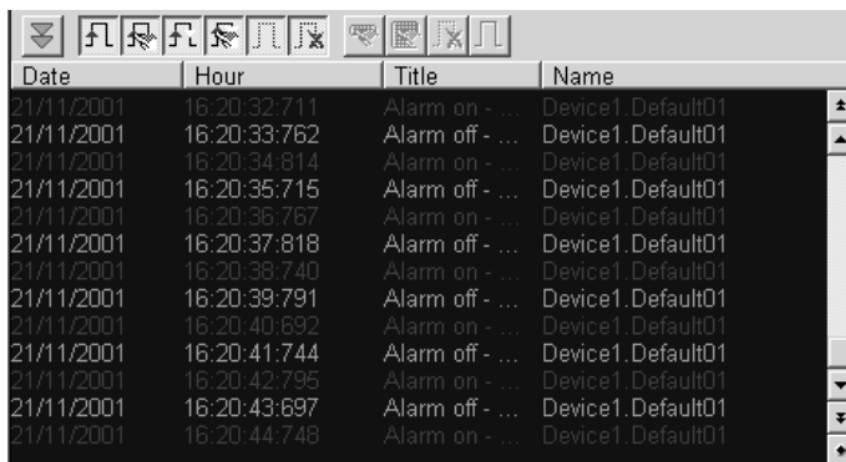
Num ambiente industrial cada vez mais complexo e competitivo, os factores relacionados com a disponibilidade e segurança da informação assumem elevada relevância, tornando-se necessário garantir que a informação está disponível e segura quando necessária, independentemente da localização geográfica. Torna-se portanto necessário implementar mecanismos de acessibilidade, mecanismos de segurança e mecanismos de tolerância a falhas.

Os sistemas SCADA melhoram a eficiência do processo de monitorização e controlo, disponibilizando em tempo útil o estado actual do sistema, através de um conjunto de

previsões, gráficos e relatórios, o que permite a tomada de decisões operacionais apropriadas, quer automaticamente, quer por iniciativa do operador.

De entre as funcionalidades de um sistema *SCADA*, os principais são:

- Gestão de alarmes, Figura 20;
- Curvas de tendência, em tempo real e histórico, Figura 21;
- Registo de dados, Figura 21;
- Interface Gráfica, Figura 23.



Date	Hour	Title	Name
21/11/2001	16:20:32:711	Alarm on - ...	Device1.Default01
21/11/2001	16:20:33:762	Alarm off - ...	Device1.Default01
21/11/2001	16:20:34:814	Alarm on - ...	Device1.Default01
21/11/2001	16:20:35:715	Alarm off - ...	Device1.Default01
21/11/2001	16:20:36:767	Alarm on - ...	Device1.Default01
21/11/2001	16:20:37:818	Alarm off - ...	Device1.Default01
21/11/2001	16:20:38:740	Alarm on - ...	Device1.Default01
21/11/2001	16:20:39:791	Alarm off - ...	Device1.Default01
21/11/2001	16:20:40:692	Alarm on - ...	Device1.Default01
21/11/2001	16:20:41:744	Alarm off - ...	Device1.Default01
21/11/2001	16:20:42:795	Alarm on - ...	Device1.Default01
21/11/2001	16:20:43:697	Alarm off - ...	Device1.Default01
21/11/2001	16:20:44:748	Alarm on - ...	Device1.Default01

Figura 20 - Scada - Gestão de alarmes [21].

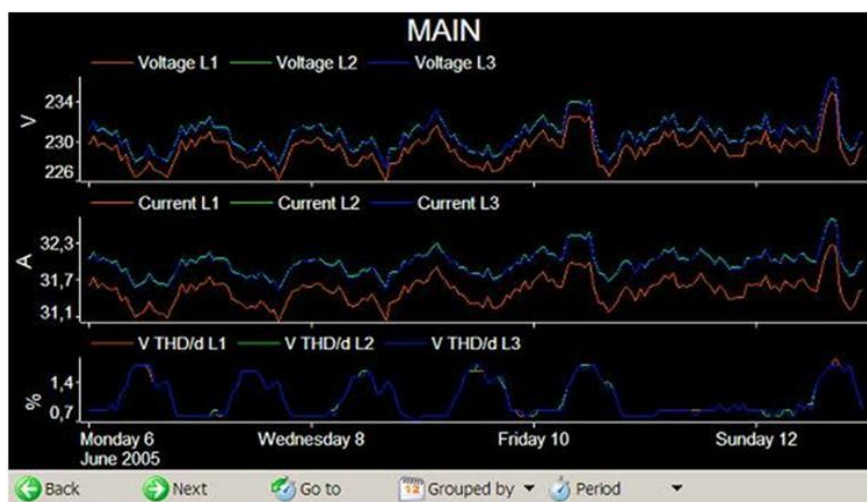


Figura 21 - Scada - Curvas de tendência.

	SB-Mech1	SB-Mech2	SB-Mech3	SB-Mech4	SB-Lab2	Total
Energy Meter Reading	0	0	0	0	0	0
Present Demand KW	0	0	0	0	0	0
Month-to-Date Energy	0	0	0	0	0	0
Month-to-Date Peak KW	0	0	0	0	0	0
Prev Month Energy	0	0	0	0	0	0
Prev Month Demand	0	0	0	0	0	0
Dem Invt 1 KWH	0	0	0	0	0	0
Dem Invt 2 KWH	0	0	0	0	0	0
Dem Invt 3 KWH	0	0	0	0	0	0
Dem Invt 4 KWH	0	0	0	0	0	0
Dem Invt 5 KWH	0	0	0	0	0	0
Month Begin KWH	0	0	0	0	0	0
Dem Invt Beg KWH	0	0	0	0	0	0
Dem Invt End KWH	0	0	0	0	0	0
Alarm KWH Roll Mult	0	0	0	0	0	0
Alarm KWH Roll Status	0	0	0	0	0	0
Demand Calc Roll Mult	0	0	0	0	0	0
Demand Calc Roll Status	0	0	0	0	0	0

	SB-C4BA	SB-C4BB	SB-C4SBA1	SB-L2SB1	SB-MEOD4B1	SB-R2BC	SB-E2B	SB-LEQD4B
Energy Meter Reading	0	0	0	0	0	0	0	0
Present Demand KW	0	0	0	0	0	0	0	0
Month-to-Date Energy	0	0	0	0	0	0	0	0
Month-to-Date Peak KW	0	0	0	0	0	0	0	0
Prev Month Energy	0	0	0	0	0	0	0	0
Prev Month Demand	0	0	0	0	0	0	0	0
Dem Invt 1 KWH	0	0	0	0	0	0	0	0
Dem Invt 2 KWH	0	0	0	0	0	0	0	0
Dem Invt 3 KWH	0	0	0	0	0	0	0	0
Dem Invt 4 KWH	0	0	0	0	0	0	0	0
Dem Invt 5 KWH	0	0	0	0	0	0	0	0
Month Begin KWH	0	0	0	0	0	0	0	0
Dem Invt Beg KWH	0	0	0	0	0	0	0	0
Dem Invt End KWH	0	0	0	0	0	0	0	0
Alarm KWH Roll Mult	0	0	0	0	0	0	0	0
Alarm KWH Roll Status	0	0	0	0	0	0	0	0
Demand Calc Roll Mult	0	0	0	0	0	0	0	0
Demand Calc Roll Status	0	0	0	0	0	0	0	0

Figura 22 - Scada - Registo de dados.

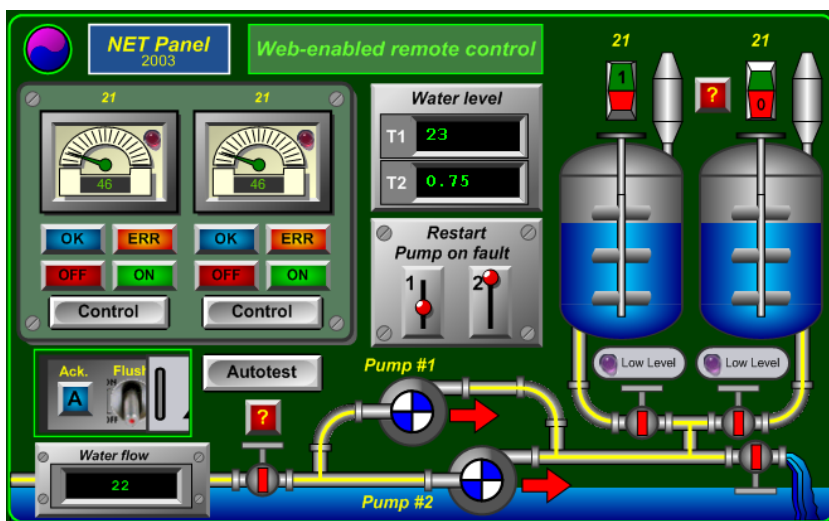


Figura 23 - Scada - Interface Gráfica.

2.6 Conclusão

O recurso a medidores, conversores e sistemas *SCADA* é já amplamente utilizado em empresas e indústria, mas apenas para uso interno. A aplicação destes para serviços de melhoria da eficiência energética tem que sofrer mudanças para tentar colmatar os problemas referidos no ponto 1.2, como, p.e., ultrapassar firewalls, sem a necessidade de abrir portos do lado do cliente, permitir o envio de dados seguros do ponto de medição para o centro de recolha de dados, permitir o acesso aos dados por outras aplicações, editar directamente os dados na base de dados, e copiar de uma base de dados para outra apenas os valores pretendidos, sem ter que fazer uma cópia integral da mesma.

Capítulo 3

Arquitectura do Sistema Implementado

Neste capítulo é apresentada a solução adoptada para o problema apresentado no ponto 1.2 desta dissertação, para a realização de medições dos dados energéticos.

Foi criada uma rede virtual para interligar dois pontos distintos, um que alberga os equipamentos de medição colocados nos pontos onde se pretende efectuar a medição dos dados energéticos, e o lugar onde os mesmos são armazenados. Foi também desenvolvido um sistema que permite recolher os dados dos equipamentos de medição, e armazená-los numa base de dados, para que sejam apresentados através de uma plataforma Web.

3.1 Criação da Rede Virtual Privada

Para a criação de do túnel VPN, Figura 24, entre o ponto onde é colocado o medidor, e a rede onde se encontra um servidor para armazenamento de dados, recorreu-se à aplicação OpenVPN apresentada em [23].

Foi instalada a versão servidor da aplicação no servidor de VPN, a correr *Ubuntu*, e configurado, como descrito *a posteriori*, para permitir o acesso ao servidor que irá armazenar os dados provenientes dos medidores.

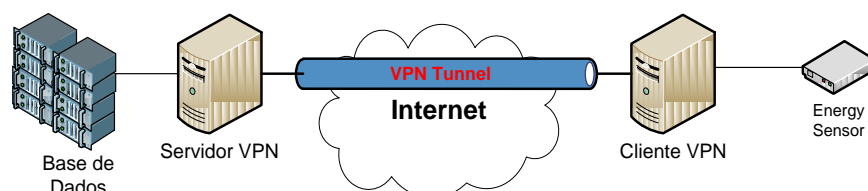


Figura 24 - Ligação Servidor-Medidor.

Para facilitar o processo de criação das chaves criptográficas, foi instalada aplicação Webmin, Figura 25 e apresentada em [24], que também nos permite configurar a *OpenVPN*.

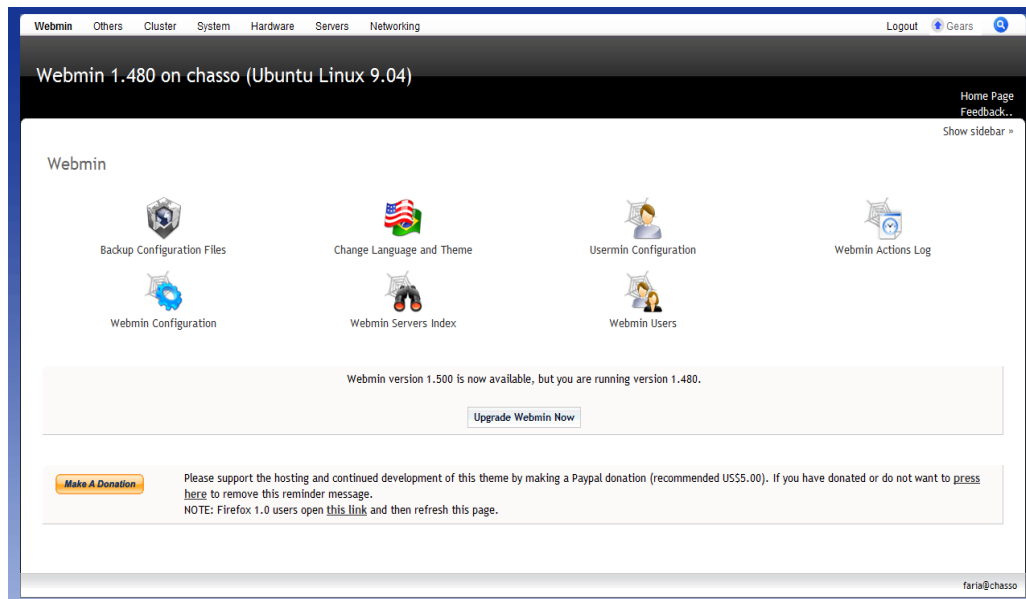


Figura 25 - Aplicação Webmin.

Depois de instalado o *Webmin* e a *OpenVPN*, começa-se por configurar um *CA*, para posteriormente gerar as respectivas chaves, quer para o servidor, quer para os clientes que irão permitir comunicar com os medidores.

Para configurar um *CA*, preenche-se os dados indicados na Figura 26. Escolhe-se o tipo de encriptação pretendida, tendo em atenção que quanto maior, no caso apresentado está 2048 bits, mais lenta é a geração da chave, mas também mais difícil de decifrar.

New Certification Authority	
Name of Certification Authority	<input type="text" value="changeme"/>
Complete path to openssl.cnf	<input type="text" value="/etc/openvpn/openvpn-ssl.cnf"/>
Keys directory	<input type="text" value="/etc/openvpn/keys"/>
Key size (bit)	<input type="text" value="2048"/>
Expiration time of Certification Authority key (days)	<input type="text" value="3650"/>
State	<input type="text" value="US"/>
Province	<input type="text" value="NY"/>
City	<input type="text" value="New York"/>
Organization	<input type="text" value="My Org"/>
Email	<input type="text" value="me@my.org"/>

Figura 26 - Interface para criar um CA.

Após gerar o CA, é necessário criar uma chave para o servidor que ficará do lado da empresa que presta o serviço. Para tal, basta escolher no *key server* a opção servidor e preencher os dados de acordo com a Figura 27. Esta chave é única, independentemente da quantidade de clientes que possam estar na *VPN*.

New key to Certification Authority: smartwall_gp

Key name: changeme

Key password (min 4 chars):

Key Server: server

Generate exportable PKCS#12 key: no

Server key doesn't need password!

Password for exporting PKCS#12 (min 4 chars):

Key expiration time (days): 3650

State: PT

Province: Porto

City: Porto

Organization: Smartwall

Organization Unit: Office

Email: info@smartwall

Save

Figura 27 - Interface para gerar Chave Servidor.

Finda a criação da chave que irá ser usada no servidor, é necessário configurar o servidor para a chave criada, o porto usado para a ligação vinda da internet (tipicamente usa-se o 1194), o tipo de encriptação usada no túnel, apresentado na Figura 28, a rede que será atribuída aos equipamentos, quer computadores, quer os conversores, que estarão na *VPN*, como demonstra a Figura 29. Aqui é, também, a altura de encaminhar rotas da rede interna e opções de *DNS*, caso tal seja pretendido.

Encrypt packets with cipher algorithm (option cipher) AES-128-CBC 128 bit default key (fixed)

Figura 28 - Selecção do tipo de encriptação usada no túnel *VPN*.

Desta forma, permitimos que os equipamentos que estão ligados na *VPN*, possam usar um servidor interno de *DNS* para resolver os pedidos, principalmente quando são usados nomes em vez de *IPs* na rede interna.

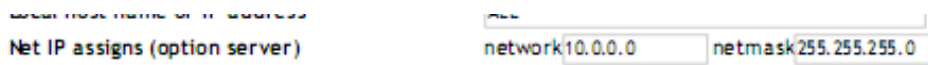


Figura 29 - Rede túnel VPN.

Após configurar o servidor, pode-se gerar a chave para o cliente, tal como gerado para o servidor, Figura 27, apenas mudando o parâmetro *key server* para *client*. Caso seja pretendido, pode ser adicionado uma *password*, garantindo assim ainda mais segurança. De notar que para o caso em questão, isto não foi usado pois pretende-se que a ligação seja automática, e que seja estabelecida aquando do arranque da máquina cliente, sem qualquer tipo de interacção humana. Caso se colocasse uma *password*, tal não seria possível.

Aqui, tal como sucedeu com o servidor, é necessário então passar à parte de configuração do cliente. Neste ponto, além de escolher a chave criada no ponto anterior, define-se o *IP* público do servidor e outras configurações adicionais, como por exemplo, partilhar a rede interna do lado do cliente, tal como foi feito com o servidor. De notar que neste ponto não é possível escolher o tipo de encriptação usado no túnel, pois este parâmetro é definido do lado do servidor. Neste caso torna-se imperativo colocar *IP* fixo no cliente, pois é ele que comunica com o medidor, e sem *IP* fixo, sistemas automatizados de recolha de dados não funcionariam. Para tal, preenche-se o campo “*ccd file configurations ccd file content*”, tendo em atenção que isto se trata de uma rede *P2P*, e como tal tem que se reservar 4 *IPs*, um para a rede, um para cada ponto e um para *broadcast*.

Exporta-se então os ficheiros criados para o cliente e instala-se o cliente OpenVPN no computador que estará ligado ao medidor, directamente ou através de um conversor referido anteriormente.

Terminando estes passos, apenas nos falta permitir o encaminhamento da rede interna, descrito no ponto 3.3, para o túnel, abrir o porto que foi configurado aquando da configuração do servidor *VPN* para permitir as ligações provenientes da internet ou em alternativa, colocar o servidor numa *DMZ* e exportar a chave criada para o cliente.

No capítulo 4 será referido o caso prático, onde serão apresentados exemplos dos ficheiros criados, quer para o servidor, quer para o cliente.

3.2 Sistema de recolha de dados

Para a comunicação com os aparelhos de medição, através do protocolo *Modbus*, foi utilizada a biblioteca gratuita e *opensource Jamod*, descrita em [25], desenvolvida em e para *Java*.

Esta biblioteca permite o envio e recepção de comandos *Modbus* e *TCP Modbus* através de diversas funções, para modelos servidor/cliente.

Foram desenvolvidas diversas funções que utilizam esta biblioteca, permitindo assim obter os dados dos registos pretendidos dos medidores, que se encontram no outro extremo do túnel *VPN*, e, através de funções criadas para armazenamento na base de dados, descritas posteriormente, armazenar os mesmos.

Um diagrama com a estrutura de funções criadas é apresentado na Figura 30, sendo que as funções dividem-se em dois grupos, as que funcionam com a base de dados, apresentadas a verde, e as que comunicam com o equipamento de medição, a azul.

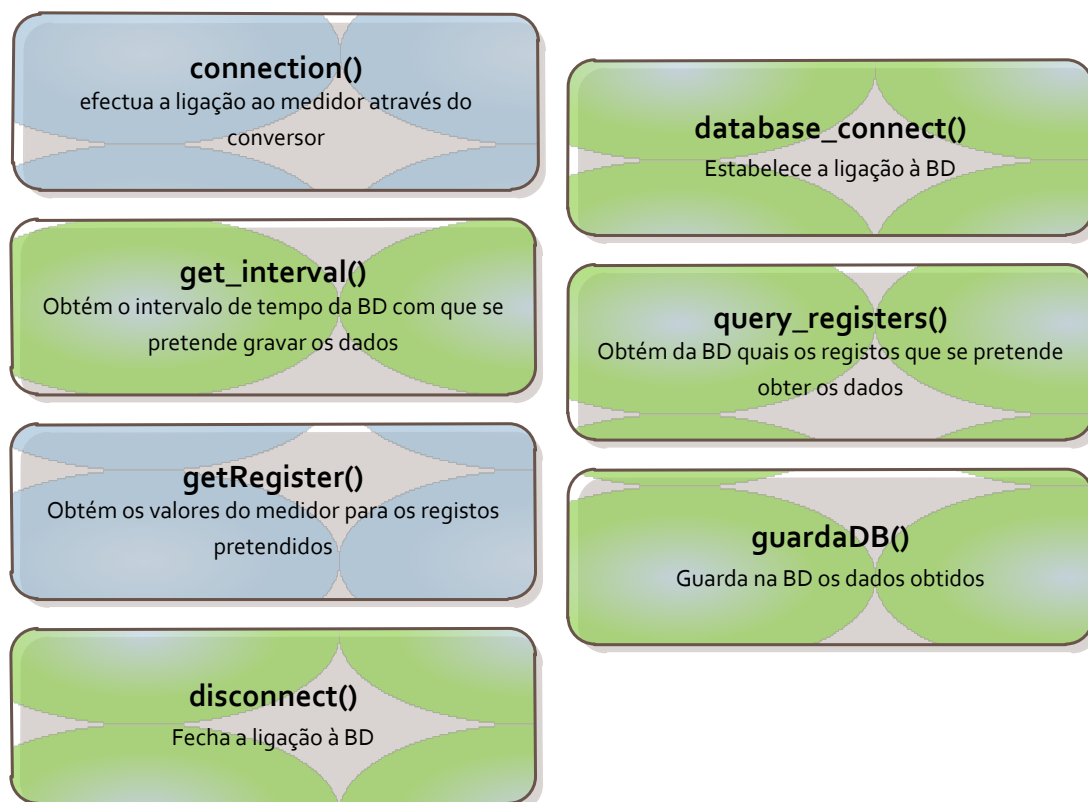


Figura 30 - Diagrama de funções desenvolvidas em Java.

O código desenvolvido para as funções criadas é apresentado seguidamente, com uma breve descrição do seu funcionamento.

- *Connection()* - Efectua a ligação ao conversor *Ethernet<->RS485* , a partir do seu *IP* da *VPN*, e do porto onde este se encontra à escuta de pedidos.

```
public void connection (String host, int port) throws UnknownHostException{
    InetAddress addr = InetAddress.getByByName(host);
    con = new TCPMasterConnection(addr);
    con.setPort(port);
    try {
        con.connect();
        System.out.println("ligado");
    } catch (Exception e) {
        e.printStackTrace();
        System.out.println("nao ligado");
    }
}
```

- `database_connect()` - estabelece a ligação à base de dados *MySQL*, para posterior consulta da mesma.
- `disconnect()` - Fecha a ligação à base de dados.
-

```
public static Connection database_connect() {
    Connection conn = null;
    String userName = "user";
    String password = "password";
    String url = "jdbc:mysql://192.168.10.10/energia";
    try{
        Class.forName ("com.mysql.jdbc.Driver").newInstance ();
        conn = DriverManager.getConnection (url, userName, password);
    }
    catch (Exception e){
        System.out.println(e);
        System.err.println ("Cannot connect to database server");
    }
    return conn;
}

public static void disconnect(Connection conn){
    try {
        conn.close ();
    }
    catch (Exception e) { /* ignore close errors */ }
}
```

- `get_Interval()` - Obtem da base de dados o parâmetro que indica o intervalo, em milisegundos, de tempo com que o sistema irá efectuar o pedido de leitura dos valores medidos no equipamento de medição.

```
public int get_interval(){
    int intervalo = 100000; //caso a base de dados esteja vazia, mede de 100 em 100
segundos
    Connection connect=database_connect();
    String sql = "SELECT * FROM configuracao";
    try {
        Statement data = connect.createStatement();
        data.executeQuery(sql);
        ResultSet answer = data.getResultSet();
        int count = 0;
        while ( answer.next() ) {
            intervalo=answer.getInt("intervalo_medicao");
            count++;
        }
        answer.close();
    } catch (SQLException e) {
        e.printStackTrace();
    }
    disconnect(connect);
    return intervalo;
}
```

- query_registers() - Obtém da base de dados os registos, ou seja, as variáveis de memória interna que contem os para metro de medição pretendidos, como por exemplo, tensão.

```
public static String[][] query_registers(){
    String[][] dados = null;
    ArrayList<String> tipo = new ArrayList<String>();
    ArrayList<String> name = new ArrayList<String>();
    ArrayList<String> value = new ArrayList<String>();
    ArrayList<String> multiplicativo = new ArrayList<String>();
    ArrayList<String> units = new ArrayList<String>();
    ArrayList<List> lista = new ArrayList<List>();
    Connection connect=database_connect();
    String sql = "SELECT * FROM registers";
    try {
        Statement data = connect.createStatement();
        data.executeQuery(sql);
        ResultSet answer = data.getResultSet();
        int count = 0;
        while ( answer.next() ) {
            tipo.add(answer.getString("tipo"));
            name.add(answer.getString("name"));
            value.add(answer.getString("value"));
            multiplicativo.add(String.valueOf(answer.getInt("multiplicativo")));
            units.add(answer.getString("units"));
            count++;
        }
        dados=new String [count][5];
        for (int i=0;i<count;i++){
            dados[i][0]=tipo.get(i);
            dados[i][1]=name.get(i);
            dados[i][2]=value.get(i);
            dados[i][3]=multiplicativo.get(i);
            dados[i][4]=units.get(i);
        }
        answer.close();
    } catch (SQLException e) {
        e.printStackTrace();
    }
    disconnect(connect);
    return dados;
}
```

- `getRegister()` - Obtêm do equipamento que se encontra a fazer medições o valor guardado num determinado registo de memória.

```
public float getRegister(String register,int unitid) {
    reg =Integer.parseInt(register, 16);
    ReadInputRegistersRequest req = new
ReadInputRegistersRequest(reg,Integer.decode("4").intValue());
    //atribuir o numero da unidade
    req.setUnitID(unitid);
    ModbusTCPTransaction trans = new ModbusTCPTransaction(con);
    //enviar o pedido
    trans.setRequest(req);
    try {
        trans.execute();
    } catch (ModbusIOException e) {
        e.printStackTrace();
    } catch (ModbusSlaveException e) {
        e.printStackTrace();
    } catch (ModbusException e) {
        e.printStackTrace();
    }
    try {
        //obter a resposta
        ReadInputRegistersResponse res = (ReadInputRegistersResponse) trans.getResponse();
        if (res.getRegisterValue(0)<0){
            valor_reg=Math.abs(res.getRegisterValue(1));
        }
        else{
            valor_reg=res.getRegisterValue(0)*65536+res.getRegisterValue(1);
        }
        return valor_reg;
    } catch (ClassCastException e) {
        trans.getResponse();
        return -1;
    }
}
```

- guardaDB() - Guarda na base de dados os valores medidos para os registos pretendidos obtidos a partir da função query_registers().

```
public static void guardaDB(String tipo,String name,Float valor,String units){
    String data_hora_local = data_hora_actual();
    Connection connect=database_connect();
    String sql = "INSERT INTO "+tipo+"(data,name,valor,units)
VALUES("+data_hora_local+", "+name+", "+valor+", "+units+)";
    try {
        Statement data = connect.createStatement();
        int answer = data.executeUpdate(sql);
        data.close();
        connect.close();
    } catch (SQLException e) {
        e.printStackTrace();
    }
    disconnect(connect);
}
```

3.3 Encaminhamentos

Para que o servidor permita o acesso à sua rede interna a partir da VPN, é necessário adicionar regras de encaminhamento do tráfego IPv4 no SO. Basta para isso fazer na consola, ou terminal, o seguinte:

```
>>nano /etc/sysctl.conf
net.ipv4.ip_forward = 1
>>sysctl -p /etc/sysctl.conf
```

Nos clientes Windows onde queremos encaminhar a rede interna, ou seja, queremos que seja possível chegar à rede interna do cliente a partir do túnel VPN, é modificado uma chave de registo, *HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters*, e acrescentar o valor "IPEnableRouter"=1. Com isto, permitimos que o tráfego TCP/IP seja encaminhado entre a placa de rede física, que nos fornece a rede local, e a placa de rede virtual, que nos fornece a rede proveniente da VPN.

Com isto, já conseguimos atingir o conversor do lado do servidor para recolher dados.

3.4 Base de Dados

A base de dados usada é uma base de dados *MySQL*, pois é *opensource* e muito fiável, permitindo desta forma que os dados armazenados possam também ser usados por aplicações de terceiros ou futuras aplicações, além de também permitir efectuar alguma análise de dados directamente da base de dados.

Utilizou-se então no servidor Linux a base de dados *MySQL*, e para administrar a base de dados, instalou-se também o *phpmyadmin*, que é uma ferramenta Web para gestão de base de dados.

Para a ligação entre a base de dados e a aplicação desenvolvida em *java*, foi utilizado o *mysql-connector-java*, descrito em [26]. Esta ferramenta é o *driver* oficial para ligações entre aplicações *java* e base de dados *MySQL*.

Um exemplo de uma base de dados criada, tendo as tabelas principais, é apresentado na Figura 31, onde além da tabela que contem as configurações relevantes, acresce a tabela *registers*, que contem todos os registos de memória dos equipamentos de medição utilizados.

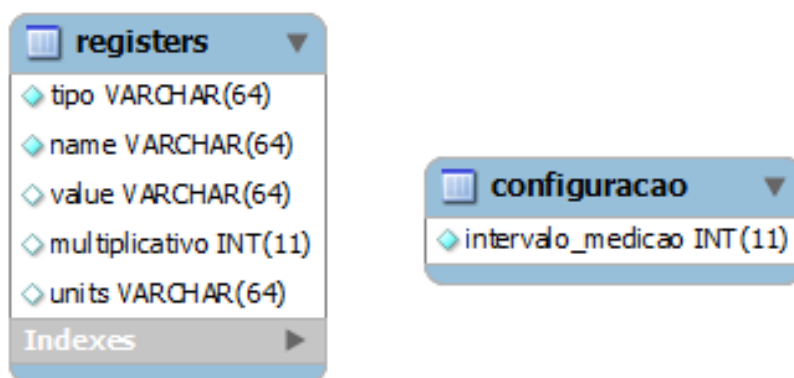


Figura 31 - Base de Dados MySQL inicial.

3.5 Interface Gráfica

Para a interface gráfica, foi instalado no servidor a correr *Ubuntu*, um servidor de html, *Apache2* e de php, o *php5*, além da já referida base de dados *MySQL*, onde estão armazenados os dados medidos, os registos usados, parâmetros das configurações e as contas de utilizador para efectuar o login na página Web.

Além do uso de *html* para desenvolver a interface visual da página *Web*, elementos estáticos, e do *php* para troca de dados entre o cliente e o servidor, elementos dinâmicos, foram usados parte de código em *javascript*, para as animações dos menus, mas principalmente para a criação dos gráficos usados.

A ferramenta que permitiu o desenho gráfico dos valores obtidos dos equipamentos de medição, é designada de *Flot*, e pode ser encontrada em [27]. Este código permite, além de criar gráficos. Interação com o utilizador, permitindo-lhe, ao passar com o rato por cima do mesmo, visualizar pontos específicos do gráfico, tal como ilustrado na Figura 32, fazer zoom no mesmo, entre outras.

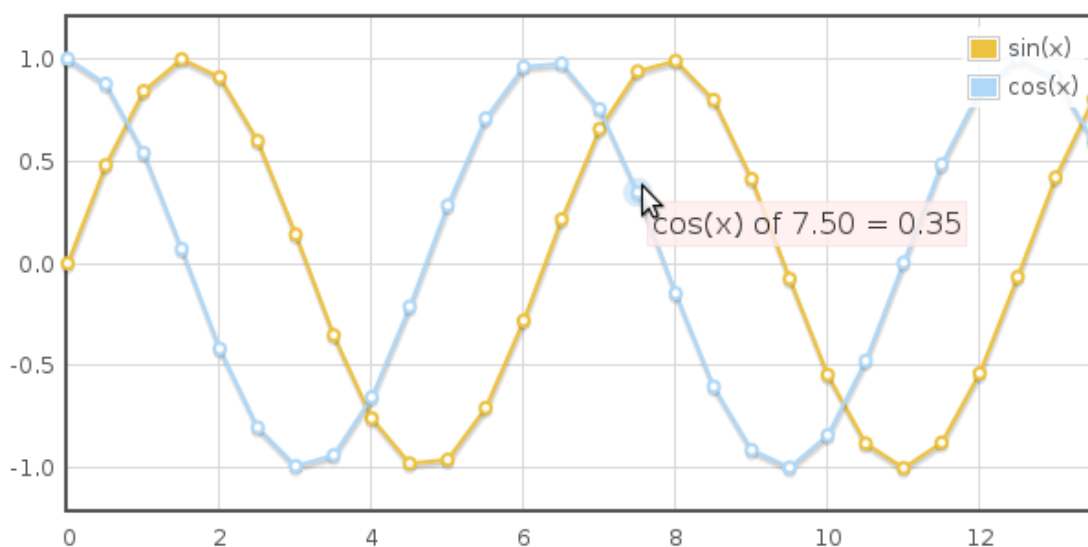


Figura 32 - Gráfico criado usando Flot [27].

O script que cria o gráfico com os valores guardados na base de dados em *javascript* usando *Flot*, é apresentado seguidamente, bastando depois no *html* colocar o gráfico, recorrendo a uma *div* com o tamanho que queremos o mesmo,

```
<div id="placeholder" style="width:800px;height:300px"></div>
```

```

<script id="source" language="javascript" type="text/javascript">
jQuery.noConflict();
var plot;
jQuery(function () {
    var V1=[], V2=[], V3=[];
    V1=valoresV1;
    V2=valoresV2;
    V3=valoresV3;

    plot = jQuery.plot(jQuery("#placeholder"),
        [ { data: V1, label: "V1 = 0.00"},
          { data: V2, label: "V2 = 0.00" },
          { data: V3, label: "V3 = 0.00" } ], {
        series: {
            lines: { show: true },
            points: { show: true }
        },
        crosshair: { mode: "x" },
        grid: { hoverable: true, autoHighlight: false , clickable: true }
    });

    var legends = jQuery("#placeholder .legendLabel");
    legends.each(function () {
        // fix the widths so they don't jump around
        jQuery(this).css('width', jQuery(this).width());
    });

    var updateLegendTimeout = null;
    var latestPosition = null;
    function updateLegend() {
        updateLegendTimeout = null;
        var pos = latestPosition;
        var axes = plot.getAxes();
        if (pos.x < axes.xaxis.min || pos.x > axes.xaxis.max ||
            pos.y < axes.yaxis.min || pos.y > axes.yaxis.max)
            return;
        var i, j, dataset = plot.getData();
        for (i = 0; i < dataset.length; ++i) {
            var series = dataset[i];
            // find the nearest points, x-wise
            for (j = 0; j < series.data.length; ++j)
                if (series.data[j][0] > pos.x)
                    break;
            // now interpolate
            var y, p1 = series.data[j - 1], p2 = series.data[j];
            if (p1 == null)
                y = p2[1];
            else if (p2 == null)
                y = p1[1];
            else
                y = p1[1] + (p2[1] - p1[1]) * (pos.x - p1[0]) / (p2[0] - p1[0]);

            legends.eq(i).text(series.label.replace(/=.*/, "=" + y.toFixed(2)));
        }
    }
}

```

```
jQuery("#placeholder").bind("plothover", function (event, pos, item) {
    latestPosition = pos;
    if (!updateLegendTimeout)
        updateLegendTimeout = setTimeout(updateLegend, 50);
});
function showTooltip(x, y, contents) {
    jQuery('<div id="tooltip">' + contents + '</div>').css( {
        position: 'absolute',
        display: 'none',
        top: y + 5,
        left: x + 5,
        border: '1px solid #fdd',
        padding: '2px',
        'background-color': '#fee',
        opacity: 0.80
    }).appendTo("body").fadeIn(200);
}
var previousPoint = null;
jQuery("#placeholder").bind("plothover", function (event, pos, item) {
    jQuery("#x").text(pos.x.toFixed(2));
    jQuery("#y").text(pos.y.toFixed(2));
    if (jQuery("#enableTooltip:checked").length > 0) {
        if (item) {
            if (previousPoint != item.datapoint) {
                previousPoint = item.datapoint;
                jQuery("#tooltip").remove();
                var x = item.datapoint[0].toFixed(2),
                    y = item.datapoint[1].toFixed(2);
                showTooltip(item.pageX, item.pageY,
                    (item.series.label).split("=")[0] + " of " + x + " = " + y);
            }
        }
        else {
            jQuery("#tooltip").remove();
            previousPoint = null;
        }
    }
});
jQuery("#placeholder").bind("plotclick", function (event, pos, item) {
    if (item) {
        jQuery("#clickdata").text("You clicked point " + item.dataIndex + " in " + item.series.label + ".");
        plot.highlight(item.series, item.datapoint);
    }
});
});(jQuery);

</script>
```

3.6 Alarmes

Em situações anormais, como falta de conectividade com o conversor *Ethernet*->*RS485*, dados anormais, ou outra situação que se pretenda monitorizar, foram criados dois *scripts* para aviso, sendo um deles por email, podendo ser usado para situações anormais, mas não críticas, p.e., e outro para envio de sms, para casos críticos, pois pode acarretar custos, dependendo do serviço com o operador móvel.

Os *scripts* foram escritos em *python*, linguagem que não necessita de tantos recursos como *java*, e que, além de mais leve, também é mais rápida a executar, o que para esta situação se justifica. Tal como o uso de *java* para garantir que a aplicação pudesse ser utilizada multi-plataforma, ou seja, independente do SO, estes *scripts* também foram criados tendo isso em mente.

O envio de sms é efectuado usando um modem *GPRS*, sendo possível usar os comuns modems 3G *UMTS* que encontramos no mercado, como o da Figura 33, estando o suporte para os mesmos nos diversos SO largamente desenvolvido. Para trocar entre os diversos SO, apenas é necessário actualizar a porta de ligação com o modem no script. Como se recorre a comandos AT para comunicar com o modem *GPRS*, pode ser usado qualquer tipo de modem, não sendo necessário um modem específico.



Figura 33 - Modem 3G UMTS.

- Envio de *e-mail*:

```
fromaddr = 'ricardo.faria@fe.up.pt'
toaddrs = ["Ricardo Faria<ricardo.faria@fe.up.pt>"]

subject = 'ERRO gerar Previsoes'
####ir ao argumento buscar o erro e adicionar a mensagem
msg = 'Erro(s) encontrado(s): \r\n'+sys.argv[1] +erro_programa
# Credentials (if needed)
username = '###'
password = '###'
#
BODY = string.join((
    "From: %s" % fromaddr,
    "Subject: %s" % subject,
    "",
    msg
), "\r\n")
# The actual mail send
server = smtplib.SMTP('smtp.fe.up.pt:25')
#server.connect()
server.login(username,password)
server.sendmail(fromaddr, toaddrs, BODY)
server.quit()
```

- Envio de SMS:

```
#variaveis
ntlm = sys.argv[1] #obtem do arg1 o n telefone
msg = sys.argv[1]#obtem do arg2 a mensagem a enviar
enter = '\r\n'
ctrlz = '\x1a\n'
#conf pserie
ser = serial.Serial()
ser.port='/dev/ttyUSB1' #unix
ser.port='com10' #windows
ser.baudrate=115200
ser.rtscts=1
ser.timeout=3
ser.open()
#configuracoes do modem
ser.write('AT'+ enter) # AT
ser.readline()
time.sleep(2)
#Registo e nivel de sinal
ser.write('AT+CREG?' + enter)
ser.readline()
reply = ser.readline()
if reply == '+CREG: 0,1'+ enter:
    print "Modem registado na rede..."
else:
    print "Falha de registo..."
    sys.exit
ser.write('AT+CMEE=1'+ enter)
time.sleep(2)
#enviar sms
print "Enviar SMS's"
ser.write('AT+CMGF=1'+ enter)
print "CMGF:",ser.read(0)
time.sleep(2)
ser.write('AT+CMGS="'+ntlm+'"+ enter)
print "CMGS:",ser.read(0)
time.sleep(2)
ser.write(''+ msg + '' + ctrlz)
reply = ser.read(2)
print "final:",reply
```

Capítulo 4

Caso Prático

O sistema de contabilidade energética municipal tem como principal objectivo fazer uma gestão e controlo de energia nas instalações municipais. Neste sistema é armazenada informação referente a todas as instalações com consumo de energia eléctrica facturado ao município. É efectuado o tratamento de dados que resultam em informações sobre o consumo e sugestões ao utilizador para reduzir a facturação energética. Estas sugestões resultam da comparação de valores de cada instalação com os valores parametrizados de consumo por área e por potência contratada, permitindo identificar consumos energéticos e potências contratadas acima dos valores normais para cada tipo de instalação.

Principais funções do sistema de contabilidade energética:

- Cruzar informação das características das instalações com a facturação para permitir identificar indícios de ineficiências;
- Visualizar e analisar históricos de consumo para detectar consumos anómalos, alertando o gestor para o facto;
- Indicar qual o melhor tarifário e fornecer sugestões de actuação comercial e comportamental;
- Apresenta uma síntese de consumos e relatórios, com tabelas e gráficos que permitem aos autarcas estar actualizados quanto às contas energéticas.

O caso para o qual o sistema foi inicialmente desenvolvido foi para a da Câmara Municipal de Ponte da Barca, CMPB, mas estruturado para se aplicar em qualquer situação.

O pretendido era interligar os edifícios pertencentes à CMPB, nos quais se pretendia efectuar medições energéticas, e o integrador, da *Smartwatt - Certificação Energética e Microgeração*, empresa que iria fornecer o serviço e efectuar a análise dos dados recolhidos.

A CMPB encontra-se dividida em 3 partes. Os Paços do Concelho, que se encontra interligado na mesma infraestrutura informática que as Piscinas Municipais e o Espaço Internet, apesar de em edifícios e localizações diferentes, a Escola Básica e, finalmente, a Escola Secundária. Estas duas têm a sua própria rede independente, e não se encontram na mesma rede que os Paços do Concelho, tal como demonstrado na Figura 34.

Para tal, foi instalado um servidor a correr *Ubuntu*, no qual foram inseridas as aplicações desenvolvidas, descritas no capítulo 3, para criar a ligação VPN entre a rede da CMPB, e o integrador

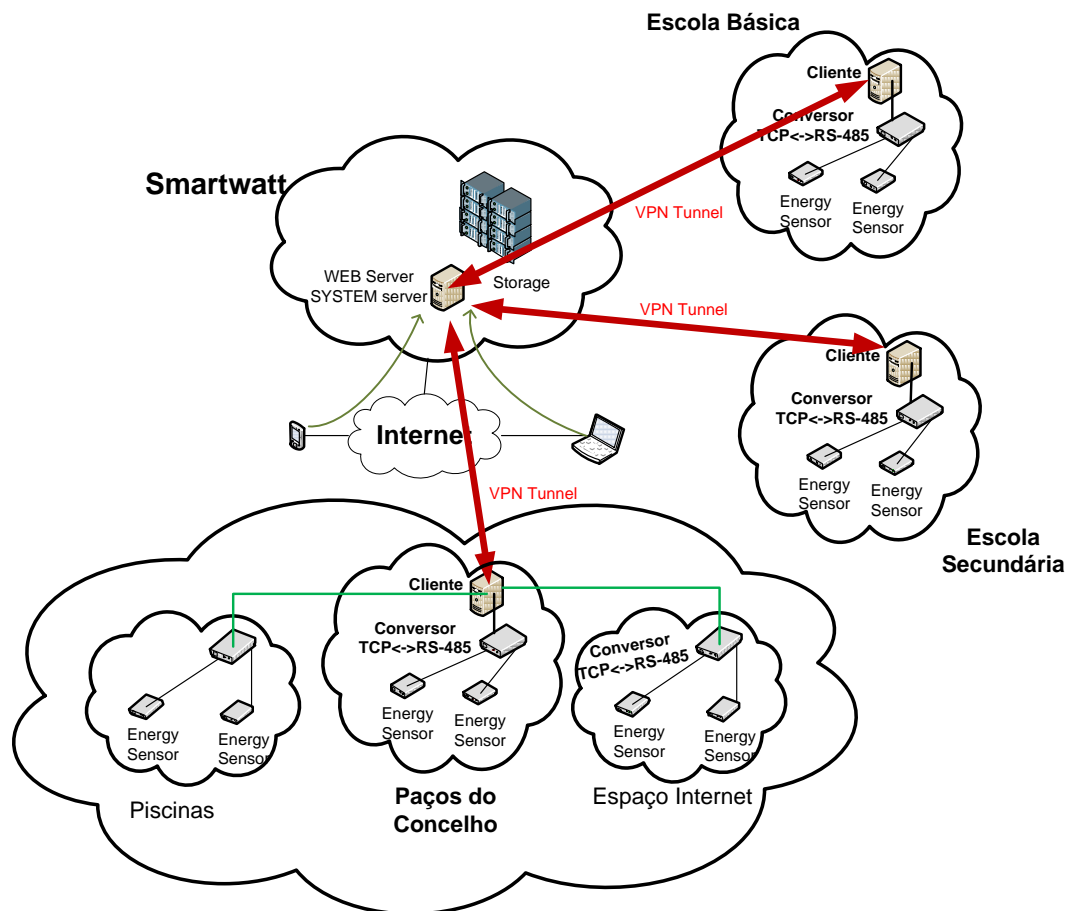


Figura 34 - Esquema do sistema instalado.

Do lado do cliente, CMPB, está instalado um computador, em cada rede diferente, Paços do Concelho, Escola Básica e Secundária, com sistema *Windows*, onde foi configurado a versão cliente de *OpenVPN*, para estabelecer o túnel *VPN* com o integrador, computador este que está fisicamente ligado na rede da CMPB. Nessa mesma rede, está um, no caso de cada escola, e vários, no caso dos Paços do Concelho, Espaço Internet e Piscinas, conversores *Ethernet<->RS485* ao qual os aparelhos de medição estão ligados. Cada conversor tem um endereço IP específico, que o distingue dos outros, e cada equipamento de medição também tem um identificador único dentro de cada conversor, ou seja, no mesmo conversor não

podem existir dos medidores com o mesmo identificador, mas entre conversores pode. Isto garante que se sabe exactamente de onde são provenientes os valores medidos. Basta encaminhar os pedidos pelo conversor ao qual está ligado o medidor.

Começando pelas configurações do servidor, para a *OpenVPN*, foi usado o método descrito no ponto 3.1, com o recurso à aplicação *Webmin*. Os ficheiros criados para o servidor e para o cliente apresentam-se seguidamente.

- Server.ovpn

```
port Porto
proto udp
dev tun
ca keys/smartwatt_ge/ca.crt
cert keys/smartwatt_ge/server_ge.crt
key keys/smartwatt_ge/server_ge.key
dh keys/smartwatt_ge/dh1024.pem
server 10.0.1.0 255.255.255.0
crl-verify keys/smartwatt_ge/crl.pem
cipher AES-128-CBC
user nobody
group nogroup
status servers/server_ge/logs/openvpn-status.log
log-append servers/server_ge/logs/openvpn.log
verb 2
mute 20
max-clients 100
management 127.0.0.1 8500
keepalive 10 120
client-config-dir /etc/openvpn/servers/server_ge/ccd
client-to-client
comp-lzo
persist-key
persist-tun
ccd-exclusive
push "route 10.0.1.0 255.255.255.0"
push "route 192.168.1.30 255.255.255.255"
push "route 192.168.1.31 255.255.255.255"
push "route 192.168.1.215 255.255.255.255"
push "route 192.168.1.216 255.255.255.255"
push "route 192.168.101.20 255.255.255.255"
route 192.168.1.30 255.255.255.255
route 192.168.1.31 255.255.255.255
route 192.168.1.215 255.255.255.255
route 192.168.1.216 255.255.255.255
route 192.168.101.20 255.255.255.255
```

Os *push "route"* e os *route* servem para o servidor encaminhar e avisar os clientes das redes que se estão a partilhar na *VPN*.

Também do lado do servidor, existe um ficheiro por cada cliente autorizado a ligar-se à *VPN*, designado *ccd* que, como exemplo, contém as seguintes informações:

```
ifconfig-push 10.0.1.10 10.0.1.9
iroute 192.168.101.20 255.255.255.255
iroute 192.168.101.21 255.255.255.255
iroute 192.168.104.200 255.255.255.255
iroute 192.168.105.200 255.255.255.255
iroute 192.168.106.200 255.255.255.255
```

A primeira linha indica ao cliente qual o seu *IP* na *VPN*, e qual a *gateway* que deve usar para comunicar com a *VPN*. As restantes linhas indicam quais as redes que este cliente partilha com a *VPN*, com os restantes clientes e com o servidor.

- *client.ovpn*

```
client
proto udp
dev tun
ca ca.crt
dh dh1024.pem
cert ge_pc.crt
key ge_pc.key
remote IP Port
cipher AES-128-CBC
verb 2
mute 20
keepalive 10 120
comp-lzo
persist-key
persist-tun
float
resolv-retry infinite
nobind
route-method exe
route-delay 2
```

Junto com este ficheiro, são colocados os ficheiros dos certificados criados, que, por questões de segurança, não são apresentados neste ponto.

Para criação no cliente do túnel VPN, como referido anteriormente, foi instalado a versão cliente de *OpenVPN*, *OpenVPN GUI for Windows*, na qual o a ligação ao servidor VPN pode ser visto em[28]. Por questões de compatibilidade com a versão *Vista* do *Windows*, foi utilizada uma versão beta do cliente, apesar de ser uma versão bastante estável. A ligação do cliente à VPN pode ser vista na Figura 35.



Figura 35 - Cliente OpenVPN Windows.

Para a base de dados *MySQL*, foram criadas tabelas segundo o esquema da Figura 36.

Este esquema é o esquema genérico, sendo que, quando se pretende aumentar o número de equipamentos de medição, basta replicar as tabelas genéricas.

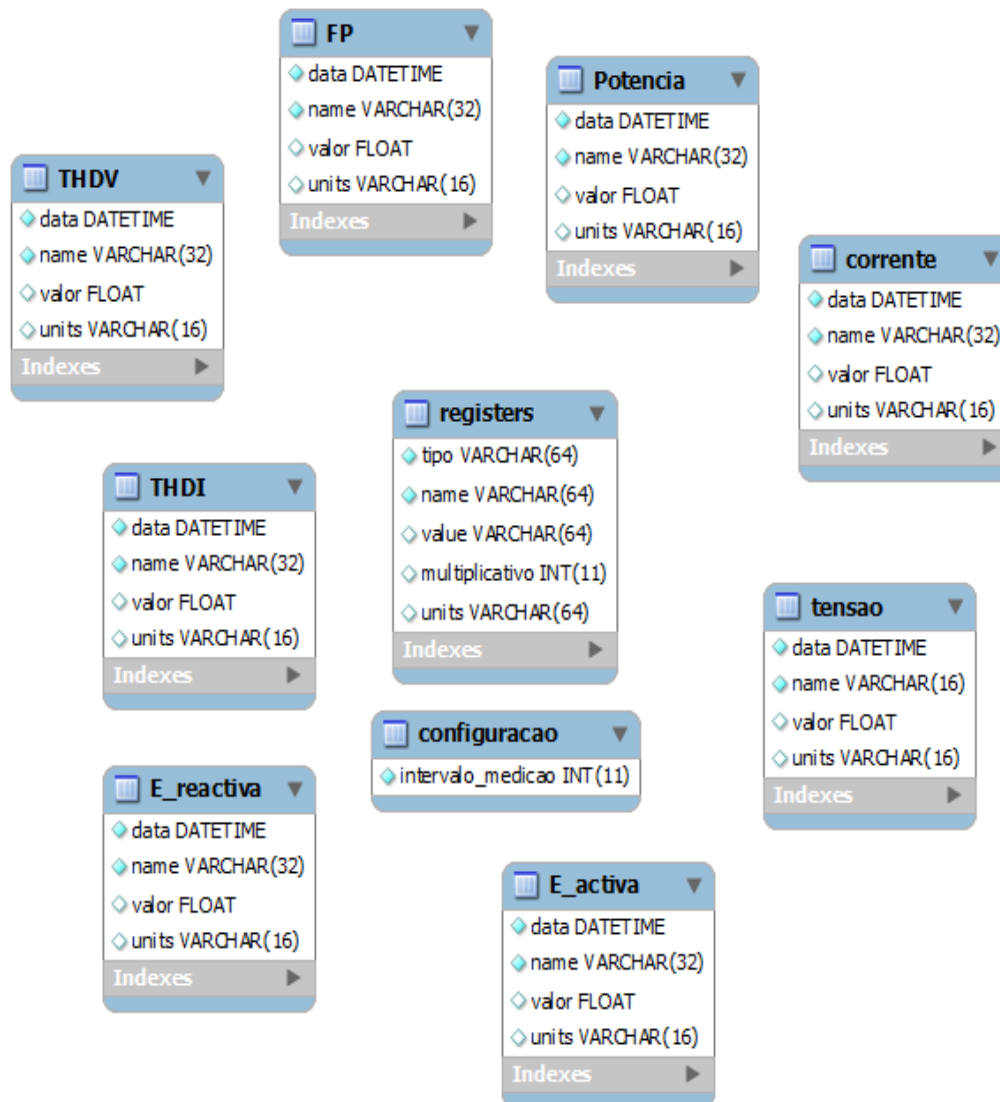


Figura 36 - Esquema da base de dados utilizada.

O script utilizado para a criação das tabelas é apresentado seguidamente:

```
SET SQL_MODE="NO_AUTO_VALUE_ON_ZERO";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS
*/;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8 */;

--
-- Base de Dados: `energia`
-----

-- Estrutura da tabela `configuracao`
DROP TABLE IF EXISTS `configuracao`;
CREATE TABLE IF NOT EXISTS `configuracao` (
  `intervalo_medicao` int(11) NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
-----

-- Estrutura da tabela `corrente`
DROP TABLE IF EXISTS `corrente`;
CREATE TABLE IF NOT EXISTS `corrente` (
  `data` datetime NOT NULL default '0000-00-00 00:00:00',
  `name` varchar(32) NOT NULL default "",
  `valor` float default NULL,
  `units` varchar(16) default NULL,
  UNIQUE KEY `data` (`data`,`name`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
-----

-- Estrutura da tabela `E_activa`
DROP TABLE IF EXISTS `E_activa`;
CREATE TABLE IF NOT EXISTS `E_activa` (
  `data` datetime NOT NULL default '0000-00-00 00:00:00',
  `name` varchar(32) NOT NULL,
  `valor` float default NULL,
  `units` varchar(16) default NULL,
  UNIQUE KEY `data` (`data`,`name`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
-----
```

```
-- Estrutura da tabela `E_reactiva`  
DROP TABLE IF EXISTS `E_reactiva`;  
CREATE TABLE IF NOT EXISTS `E_reactiva` (  
  `data` datetime NOT NULL default '0000-00-00 00:00:00',  
  `name` varchar(32) NOT NULL default "",  
  `valor` float default NULL,  
  `units` varchar(16) default NULL,  
  UNIQUE KEY `data` (`data`,`name`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

```
-----  
-- Estrutura da tabela `FP`  
DROP TABLE IF EXISTS `FP`;  
CREATE TABLE IF NOT EXISTS `FP` (  
  `data` datetime NOT NULL default '0000-00-00 00:00:00',  
  `name` varchar(32) NOT NULL default "",  
  `valor` float default NULL,  
  `units` varchar(16) default NULL,  
  UNIQUE KEY `data` (`data`,`name`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

```
-----  
-- Estrutura da tabela `Potencia`  
DROP TABLE IF EXISTS `Potencia`;  
CREATE TABLE IF NOT EXISTS `Potencia` (  
  `data` datetime NOT NULL default '0000-00-00 00:00:00',  
  `name` varchar(32) NOT NULL default "",  
  `valor` float default NULL,  
  `units` varchar(16) default NULL,  
  UNIQUE KEY `data` (`data`,`name`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

```
-----  
-- Estrutura da tabela `registers`  
DROP TABLE IF EXISTS `registers`;  
CREATE TABLE IF NOT EXISTS `registers` (  
  `tipo` varchar(64) NOT NULL,  
  `name` varchar(64) NOT NULL default "",  
  `value` varchar(64) default NULL,  
  `multiplicativo` int(11) default '1',  
  `units` varchar(64) default NULL,  
  KEY `tipo` (`tipo`,`name`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

```
-----  
-- Estrutura da tabela `tensao`  
  
DROP TABLE IF EXISTS `tensao`;  
CREATE TABLE IF NOT EXISTS `tensao` (  
  `data` datetime NOT NULL default '0000-00-00 00:00:00',  
  `name` varchar(16) NOT NULL default "",  
  `valor` float default NULL,  
  `units` varchar(16) default NULL,  
  UNIQUE KEY `data` (`data`,`name`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

```
-- Estrutura da tabela `tensao`
DROP TABLE IF EXISTS `tensao`;
CREATE TABLE IF NOT EXISTS `tensao` (
  `data` datetime NOT NULL default '0000-00-00 00:00:00',
  `name` varchar(16) NOT NULL default "",
  `valor` float default NULL,
  `units` varchar(16) default NULL,
  UNIQUE KEY `data` (`data`,`name`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
-----

-- Estrutura da tabela `THDI`
DROP TABLE IF EXISTS `THDI`;
CREATE TABLE IF NOT EXISTS `THDI` (
  `data` datetime NOT NULL default '0000-00-00 00:00:00',
  `name` varchar(32) NOT NULL default "",
  `valor` float default NULL,
  `units` varchar(16) default NULL,
  UNIQUE KEY `data` (`data`,`name`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
-----

-- Estrutura da tabela `THDV`
DROP TABLE IF EXISTS `THDV`;
CREATE TABLE IF NOT EXISTS `THDV` (
  `data` datetime NOT NULL default '0000-00-00 00:00:00',
  `name` varchar(32) NOT NULL default "",
  `valor` float default NULL,
  `units` varchar(16) default NULL,
  UNIQUE KEY `data` (`data`,`name`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

A aplicação Web desenvolvida serve para uma breve análise dos dados recolhidos, e configurações básicas, desde criar, apagar e alterar registos de memória, dos quais pretendemos obter dados. Seguem-se imagens da plataforma Web de gestão e análise desenvolvida.

Inicialmente, é pedido ao utilizador que efectue o *login* na página, como apresentado na Figura 37.



Figura 37 - Página Web, Autenticação.

Após a autenticação, o utilizador tem três opções de escolha, *Gestão*, *Registos* e *Dados* no menu.

Na parte de *Gestão*, é apresentada informação sobre o período entre cada recolha de dados dos equipamentos de medição, como apresentado na Figura 38.



Figura 38 - Página Web, Período de amostragem.

Na página *Registos*, o utilizador pode consultar os registos de memória dos diferentes parâmetros de medição, como ilustra a Figura 39.

GESTÃO REGISTOS DADOS

Tipo	Nome	Valor	Multiplicativo	Unidades
tensao	v1	00	10	V
tensao	v2	0A	10	V
tensao	v3	14	10	V
THDV	THD_V1	30	10	%
THDV	THD_V2	32	10	%
THDV	THD_V3	34	10	%
FP	FP	24	100	
Potencia	Kw3	18	1	W
Potencia	Kw2	0E	1	W
Potencia	KwIII	1E	1	W
Potencia	KvarLIII	20	1	W
Potencia	KvarCIII	22	1	W
FP	FPIII	26	100	
THDI	THD_I1	36	10	%
THDI	THD_I2	38	10	%
THDI	THD_I3	3A	10	%
E_activa	KwhIII	3C	1	Wh
E_reactiva	KvarhLIII	3E	1	Wh
E_reactiva	KvarhCIII	40	1	Wh
corrente	A1	02	1	mA
corrente	A2	0C	1	mA
corrente	A3	16	1	mA
Potencia	Kw1	04	1	W

Editar Registos

Figura 39 - Página Web, Registos.

Estes dados podem ser consultados no manual do fabricante do aparelho de medição, e foram utilizados estes parâmetros para todos os equipamentos de medição instalados.

Nesta página, o utilizador pode optar por actualizar os registos, podendo acrescentar, modificar e apagar os registos existentes na base de dados, como se pode verificar na Figura 40.

GESTÃO REGISTOS DADOS

<input type="checkbox"/>	Tipo	Nome	Valor	Multiplicativo	Unidades
<input type="checkbox"/>	tensao	v1	00	10	V
<input type="checkbox"/>	tensao	v2	0A	10	V
<input type="checkbox"/>	tensao	v3	14	10	V
<input type="checkbox"/>	THDV	THD_V1	30	10	%
<input type="checkbox"/>	THDV	THD_V2	32	10	%
<input type="checkbox"/>	THDV	THD_V3	34	10	%
<input type="checkbox"/>	FP	FP	24	100	
<input type="checkbox"/>	Potencia	Kw3	18	1	W
<input type="checkbox"/>	Potencia	Kw2	0E	1	W
<input type="checkbox"/>	Potencia	KwIII	1E	1	W
<input type="checkbox"/>	Potencia	KvarLIII	20	1	W
<input type="checkbox"/>	Potencia	KvarCIII	22	1	W
<input type="checkbox"/>	FP	FPIII	26	100	
<input type="checkbox"/>	THDI	THD_I1	36	10	%
<input type="checkbox"/>	THDI	THD_I2	38	10	%
<input type="checkbox"/>	THDI	THD_I3	3A	10	%
<input type="checkbox"/>	E_activa	KwhIII	3C	1	Wh
<input type="checkbox"/>	E_reactiva	KvarhLIII	3E	1	Wh
<input type="checkbox"/>	E_reactiva	KvarhCIII	40	1	Wh
<input type="checkbox"/>	corrente	A1	02	1	mA
<input type="checkbox"/>	corrente	A2	0C	1	mA
<input type="checkbox"/>	corrente	A3	16	1	mA
<input type="checkbox"/>	Potencia	Kw1	04	1	W

Figura 40 - Página Web, Editar Registos.

Para visualização dos dados obtidos das medições, existem duas hipóteses no menu *Dados*, *Figura 41, Gráficos e Tabelas*.

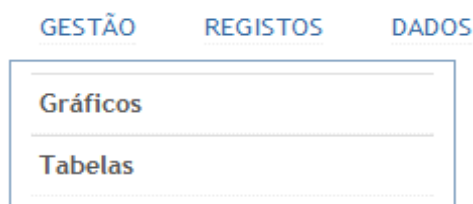


Figura 41 -Página Web, menu Dados.

Quer em *Gráficos*, quer em *Tabelas*, pode-se optar pela escolha do período que pretendemos observar os dados guardados. Existe um conjunto de *dropdowns*, que nos permitem efectuar essa escolha, como mostra a Figura 42 e Figura 43. Existe a opção de escolha de um determinado dia, um mês inteiro, ou um ano completo.

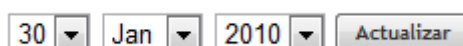


Figura 42 - Página Web, escolha do periodo de amostragem.

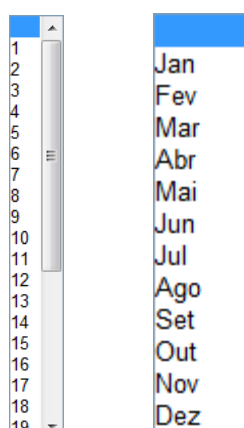


Figura 43 - Página Web, escolha de dia e mês, respectivamente.

Para a escolha de um mês completo, basta colocar o campo *Dia* vazio, e para o ano completo, coloca-se o campo *Dia* e *Mês* vazio, como é apresentado na figura seguinte.

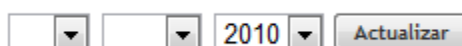


Figura 44 - Página Web, escolha dos dados de um ano completo.

Além da escolha do período de amostragem, é também possível escolher entre três tipos de dados recolhidos, *Tensão*, *Potência Activa* e *Energia Activa*.



Figura 45 - Página Web, escolha dos dados.

Como o nome indica, em *Tabelas*, é possível consultar os dados adquiridos sob a forma de tabela, Figura 46.

30 ▾ Jan ▾ 2010 ▾ Actualizar

Tensão ▾ Actualizar tipo

Data	Hora	Nome	Valor Médio
2010-01-30	16	v1	234.75
2010-01-30	15	v1	232.63333129883
2010-01-30	14	v1	232.30000050863
2010-01-30	13	v1	232.38333638509
2010-01-30	12	v1	233.08333587646
2010-01-30	11	v1	231.96000061035
2010-01-30	10	v1	233.96666971842
2010-01-30	9	v1	234.61666615804
2010-01-30	8	v1	233.98333485921
2010-01-30	7	v1	233.88333384196
2010-01-30	6	v1	234.73333231608
2010-01-30	5	v1	232.58333587646
2010-01-30	4	v1	234.36666870117
2010-01-30	3	v1	234.95000203451
2010-01-30	2	v1	234.01666514079
2010-01-30	1	v1	233.39999643962
2010-01-30	0	v1	234.10000101725
2010-01-30	16	v2	234.45000457764
2010-01-30	15	v2	232.29999796549
2010-01-30	14	v2	232.03333282471
2010-01-30	13	v2	232.46666463216
2010-01-30	12	v2	233.29999796549
2010-01-30	11	v2	231.6799987793
2010-01-30	10	v2	233.93333435059
2010-01-30	9	v2	234.55000050863
2010-01-30	8	v2	233.76666768392
2010-01-30	7	v2	233.83333333333
2010-01-30	6	v2	234.46666717529
2010-01-30	5	v2	232.28333536784
2010-01-30	4	v2	233.49999745687
2010-01-30	3	v2	233.89999898275
2010-01-30	2	v2	233.25
2010-01-30	1	v2	232.76666514079
2010-01-30	0	v2	233.03333282471
2010-01-30	16	v3	233.5
2010-01-30	15	v3	231.88333638509
2010-01-30	14	v3	231.71666717529
2010-01-30	13	v3	231.30000050863
2010-01-30	12	v3	232.39999898275

Figura 46 - Página Web, Tabela

Por fim, no menu *Gráficos*, os dados são apresentados sob forma de gráficos interactivos descritos no ponto 3.5, como elucidam as seguintes figuras.

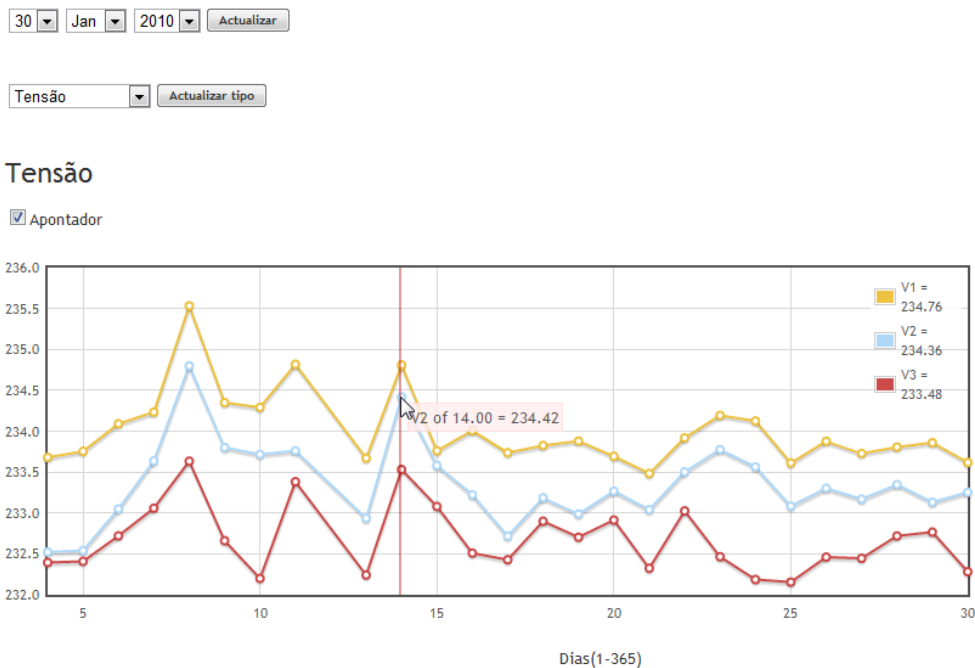


Figura 47 - Página Web, Gráfico de Tensão.

Com a caixa *Apontador* seleccionada, é possível ao passar o rato visualizar os dados específicos do ponto em questão. No canto superior direito aparecem os valores dos diferentes parâmetros, onde o rato se encontra.

Além da *Tensão*, também foram criados *scripts* para a *Potência Activa e Energia Activa*, Figura 48 e Figura 49, respectivamente.

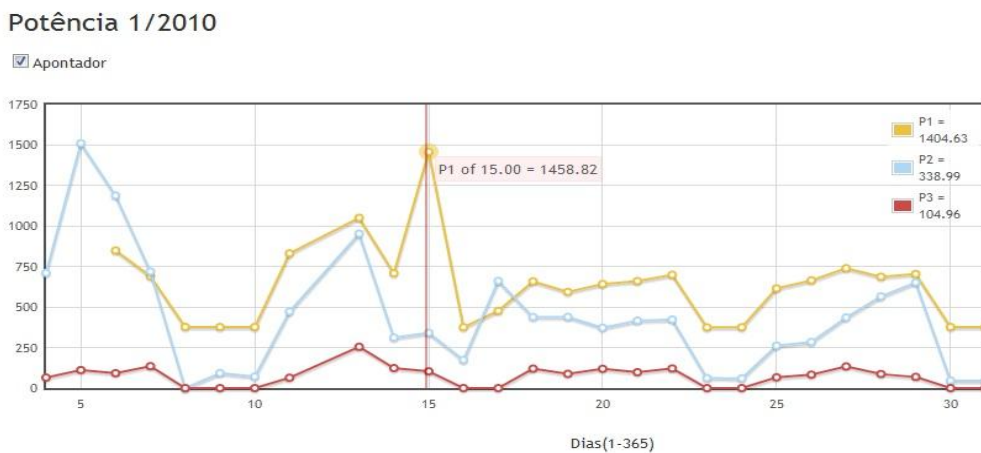


Figura 48 - Página Web, gráfico Potência Activa.

Energia Activa

Apontador

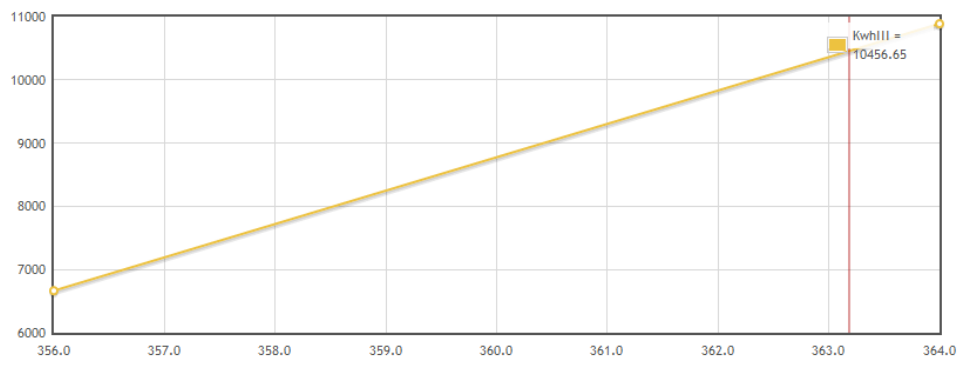


Figura 49 - Página Web, gráfico Energia Activa.

Capítulo 5

Conclusões

Este último capítulo apresenta uma síntese do trabalho reportado neste documento, referindo as conclusões retidas. São também apresentadas as perspectivas de desenvolvimento futuro.

5.1 Resultados obtidos

O trabalho desenvolvido levou à realização de um sistema de recolha de dados e à sua visualização para posterior análise, sendo criada uma aplicação que permite recolher dados de equipamentos de medição, situados numa localização diferente do sítio onde se realiza o seu armazenamento, estando em redes de computadores diferentes, interligadas através da internet.

Este sistema foi criado para ser o mais abrangente possível, permitindo o uso de diferentes tipos de equipamentos de medição e a sua aplicação em sistemas independentemente do sistema operativo utilizado.

Foi também pensado e desenvolvido o código, de forma intuitiva e simples, para que possa ser alterado de acordo com cada situação onde possa ser aplicado, caso a forma genérica não seja suficiente, sendo que o mínimo conhecimento em programação é suficiente.

Assim, os objectivos desta dissertação foram cumpridos.

Além dos objectivos iniciais, a criação de um sistema de alarme, em caso de falhas, como o envio de *e-mail* ou *sms*, é um add-on que pode ser encarado com grande utilidade, pois a

falha do sistema pode levar à falta de dados, o que pode implicar uma análise dos dados medidos não tão completa.

5.2 Desenvolvimento futuro

Como trabalho futuro, o actual sistema pode ser enriquecido com:

- Melhoria da plataforma Web, com a inclusão de novos elementos;
- Criação de relatórios dos dados obtidos;
- Aplicação dos sistemas de alarme a mais situações, como valores fora do habitual;
- ...

Referências

1. *Advanced Metering Scoping Study*. Associates, Levy. 2001.
2. **Thompson, Olin**. *Enterprise Energy Management Software - The Key to Effective Energy Utilization*. 2002.
3. **Claridge, D. E.** *A Perspective on Methods for Analysis of Measured Energy Data from Commercial Buildings*. 1998.
4. *Bulls-Eye Commissioning: Using Interval Data as Diagnostic Tool*. **Price, Will e Reid, Hart**. 2002.
5. **Seem, J. E.** *Using Intelligent Data Analysis to Detect Abnormal Energy Consumption in Buildings*. 2002.
6. **Garrido, João**. *Sistemas Energéticos para o sector Edifícios em Portugal: Sustentabilidade e Potencial de Inovação*. 2008.
7. **Circutor**. *"Analisadores de Energia"*. s.l. : Circutor.
8. **ISA - Intelligent Sensing Anywhere**. iMeterRail. *ISAlabs*. [Online] [Citação: 10 de 10 de 2009.] <http://www.isalabs.com/products/products.htm>.
9. **Lammertbies**. RS-485. *RS-485*. [Online] [Citação: 10 de 01 de 2010.] <http://www.lammertbies.nl/comm/info/RS-485.html>.
10. **ISA - Intelligent Sensing Anywhere**. iMeterPlug. [Online] [Citação: 10 de 10 de 2009.] <http://www.isalabs.com/products/products.htm>.
11. **Circutor**. Conversor RS-485 MODBUS/TCP. [Online] [Citação: 10 de 11 de 2009.] <http://www.circutor.com/m-measuring/m5-electrical-power-analyzers-cvm->

family/accessories-for-analyzers/tcp2rs-modbus-tcp-modbus-tcp-converter-m54032_R_134_5295.aspx.

12. **ISA.** iMeterLogger. *Isalabs*. [Online] [Citação: 10 de 11 de 2009.] <http://www.isalabs.com/products/imeterlogger.htm>.

13. **ISA - Intelligent Sensing Anywhere.** Bridge Ethernet RS485. *ISALabs*. [Online] [Citação: 10 de 11 de 2009.] <http://www.isalabs.com/products/docs/Bridge%20TCP%20IP%20RS485%20PB%20r.pdf>.

14. **Modbus.org.** Modbus Specs. *Modbus.org*. [Online] [Citação: 20 de 01 de 2010.] http://www.modbus.org/docs/Modbus_Application_Protocol_V1_1b.pdf.

15. **IntelliCom.** Modbus TCP overview. *IntelliCom*. [Online] [Citação: 20 de 01 de 2010.] http://www.intellicom.se/solutions_ModbusTCP_overview.cfm.

16. **Joseph Steinberg, Tim Speed, et.al.** *Understanding SSL VPN*. s.l. : Packt, 2005.

17. **Rede Nacional de Ensino e Pesquisa.** Rede Nacional de Ensino e Pesquisa. *Rede Nacional de Ensino e Pesquisa*. [Online] [Citação: 22 de 01 de 2010.] <http://www.rnp.br/newsgen/9811/vpn.html>.

18. **Hosner, Charlie.** *"OpenVPN and the SSL VPN Revolution"*. 2004.

19. **Dahl, Olr Mstyin.** *Limitations and Differences if using IPsec, TLS/SSL or SSH as VPN-solution*. 29/10/2004.

20. **Stallings, William.** *Cryptography and Network Security*. 2006.

21. **Schneider Electric.** *Diálogo Homem-Máquina, Documento técnico nº4*. s.l. : Schneider Electric, 2009.

22. **Circutor.** *Energy management Software*.

23. **OpenVPN.** OpenVPN. *OpenVPN*. [Online] [Citação: 22 de 01 de 2010.] <http://openvpn.net/>.

24. **Webmin.** Webmin. *Webmin*. [Online] [Citação: 22 de 01 de 2010.] <http://www.webmin.com/>.

25. **team, Jamod development.** Jamod. *Jamod*. [Online] [Citação: 10 de 11 de 2009.] <http://jamod.sourceforge.net>.

26. **MySQL.** MySQL Connector/J. *MySQL*. [Online] [Citação: 10 de 11 de 2009.] <http://dev.mysql.com/downloads/connector/j/3.0.html>.
27. Flot. *Flot*. [Online] <http://code.google.com/p/flot/>.
28. **GUI, OpenVPN.** OpenVPN GUI for Windows. *OpenVPN GUI for Windows*. [Online] <http://openvpn.se/development.html>.
29. **Cisco.** What Is a Virtual Private Network? [Online] [Citação: 20 de 01 de 2010.] <http://ptgmedia.pearsoncmg.com/images/1587051796/samplechapter/1587051796content.pdf>.