



UNIVERSIDADE DA BEIRA INTERIOR
Faculdade de Engenharia
Departamento de Engenharia Eletromecânica

Sistema de Localização dentro de Edifícios com SimpliciTI

João Luís Dâmaso Pereira

Dissertação para obtenção do Grau de Mestre em
Engenharia Eletrotécnica e de Computadores
(2º ciclo de estudos)

Orientador: Prof. Doutor Bruno Jorge Ferreira Ribeiro

Covilhã, junho de 2016

Agradecimentos

Quero aproveitar este modesto espaço para agradecer a todas aquelas pessoas que direta ou indiretamente me acompanharam e ajudaram no decorrer destes dois anos letivos de mestrado. Em particular agradecer à minha família, Cristina Soares e Eduardo Pereira, pelo apoio demonstrado.

O meu agradecimento à Microchip Technology Inc. pelas amostras gratuitas dos sensores de temperatura EMC1001.

Finalmente ao professor orientador Professor Doutor Bruno Ribeiro pela oportunidade da realização deste projeto, pelos conhecimentos transmitidos e pela atenção demonstrada no decorrer deste ano letivo.

Resumo

Verifica-se uma crescente utilização de sistemas de localização dentro de edifícios. No entanto, esse campo de pesquisa ainda não atingiu o mesmo grau de sucesso que o do posicionamento exterior. O objetivo deste trabalho de dissertação é a de implementar um sistema de posicionamento dentro de um edifício utilizando a tecnologia SimpliciTI, desenvolvendo uma interface gráfica para a apresentação das posições estimadas dos utilizadores. Esta foi desenvolvida no laboratório do UBIMedical na Universidade da Beira Interior. O sistema faz o gerenciamento de vários dispositivos, utilizadores, estima uma posição e apresenta a posição estimada dos utilizadores numa aplicação web.

Palavras-chave

Raspberry Pi, SimpliciTI, eZ430 Chronos, CORBA, Lateração

Abstract

There is an increasing use of indoor localization systems. However, this field of research has not yet reached the same degree of success as the outside position systems. The objective of this dissertation is to implement an indoor localization system using SimplicITI technology, developing a GUI for the presentation of estimated positions of users. This was developed in UBIMedical laboratory at the Universidade da Beira Interior. The system makes the management of multiple devices, users, estimates a position and present the position of estimated users in a web application.

Keywords

Raspberry Pi, SimplicITI, eZ430 Chronos, CORBA, Lateration

Índice

Agradecimentos	i
Resumo	iii
Palavras-chave.....	iii
Abstract	iv
Keywords	iv
Índice	v
Lista de Figuras	viii
Lista de Tabelas	x
Lista de Acrónimos	xi
1. Introdução	12
1.1. ENQUADRAMENTO.....	12
1.2. O PROBLEMA EM ESTUDO E A SUA RELEVÂNCIA.....	14
1.3. OBJETIVOS E CONTRIBUIÇÃO DA DISSERTAÇÃO	14
1.4. VISÃO GERAL E ORGANIZAÇÃO DA DISSERTAÇÃO	14
2. Estado da arte	15
2.1. INTRODUÇÃO.....	15
2.2. MÉTODOS DE LOCALIZAÇÃO	15
2.2.1. Trilateração	15
2.2.2. Multilateração	18
2.2.3. Triangulação	19
2.2.4. Detecção por proximidade.....	19
2.2.5. Navegação inercial.....	20
2.2.6. Fingerprinting.....	20
2.3. SOLUÇÕES DISPONÍVEIS.....	21
3. Hardware	22
3.1. PROTOCOLO SIMPLICITI	22
3.1.1. Relógio ez430 chronos.....	23
3.1.2. Ponto de acesso cc1111	23
3.2. RASPBERRY PI.....	24
4. Instalação e configuração de software	27
4.1. INTRODUÇÃO.....	27
4.2. COMANDOS UTILIZADOS	27
4.2.1. Sudo	27
4.2.2. Apt-get.....	27
4.2.3. Nano	28
4.2.4. Wget.....	29
4.2.5. Ifconfig.....	29

4.2.6.	Cloc	29
4.3.	SISTEMA OPERATIVO	30
4.3.1.	Preparação do cartão sd	30
4.3.2.	Alteração da palavra-passe	33
4.3.3.	Configurações iniciais	33
4.3.4.	Configuração da mensagem de boas-vindas	36
4.3.5.	Adicionar utilizador	36
4.3.6.	Configuração do hostname	37
4.4.	HABILITAR I2C	38
4.5.	DONGLE WIRELESS	40
4.6.	APACHE.....	40
4.6.1.	Instalação do apache	41
4.6.2.	Configuração do apache	42
4.6.3.	Permissões do apache	42
4.7.	HYPERTEXT PREPROCESSOR - PHP	43
4.7.1.	Instalação do php	43
4.7.2.	Tags php	43
4.7.3.	Habilitar o debug do php	43
4.8.	MYSQL.....	44
4.8.1.	Instalação do mysql.....	44
4.8.2.	Configuração mysql	44
4.8.3.	Adicionar utilizador mysql.....	45
4.9.	PHPMYADMIN	45
4.10.	WEBMIN.....	46
4.10.1.	Instalação do webmin	46
4.10.2.	Configuração do virtual host	47
4.11.	NMAP	47
4.11.1.	Instalação do nmap	48
4.11.2.	Utilização do nmap	48
4.12.	OMNIORB	48
4.12.1.	Instalação do omniorb.....	49
4.12.2.	Configuração do omniorb	49
4.13.	CONFIGURAÇÃO DO TIMEZONE.....	50
5.	Software desenvolvido no cliente	51
5.1.	PROGRAMA DE CONTROLO DA LIGAÇÃO ENTRE O AP E OS RELÓGIOS	52
5.2.	SENSOR DE TEMPERATURA.....	52
5.3.	SINCRONIZAÇÃO DE DADOS ENTRE O CLIENTE E O SERVIDOR.....	56
5.4.	PÁGINA DE CONFIGURAÇÃO DO CLIENTE	59
5.5.	LATERAÇÃO RSSI	60
6.	Software desenvolvido no servidor.....	64

6.1.	SISTEMA DE GERENCIAMENTO DO SISTEMA	64
6.1.1.	Página principal	66
6.1.2.	Sistema de acesso.....	71
6.1.3.	Controlo de acessos.....	71
6.1.4.	Política de cookies	72
6.1.5.	Visualização da localização de utilizadores	73
6.1.6.	Página de mensagens.....	76
6.1.7.	Gerenciamento de utilizadores	76
6.1.8.	Gerenciamento de dispositivos	77
6.1.9.	Informação de relógios.....	79
6.1.10.	Página de configuração do servidor.....	80
6.1.11.	Página de edição de dados pessoais	82
6.2.	ALTERAÇÃO DO HOSTNAME.....	83
6.3.	DETEÇÃO DE NOVOS DISPOSITIVOS.....	84
6.4.	DETEÇÃO DE REDES SEM FIOS.....	85
7.	Conclusão	86
8.	Referências bibliográficas	88

Lista de Figuras

Figura 1.1 - Diagrama do sistema	12
Figura 2.1 - Exemplo de trilateração	17
Figura 2.2 - Método de detecção por proximidade	20
Figura 3.1 - Rede Simplicidade	22
Figura 3.2 - Relógio eZ430 Chronos	23
Figura 3.3 - Ponto de acesso CC1111	24
Figura 3.4 - GPIO HEADER	25
Figura 4.1 - Win32 Disk Imager	31
Figura 4.2 - Configuração do cliente ssh, Putty	32
Figura 4.3 - Cliente SSH, Putty	32
Figura 4.4 - raspi-config	34
Figura 4.5 - rasp-config, opções internacionais	35
Figura 4.6 - Opções avançadas do raspi-config	35
Figura 4.7 - Comando ifconfig	41
Figura 4.8 - Página por defeito do Apache	42
Figura 4.9 - Servidor virtual	47
Figura 5.1 - Estrutura da base de dados do cliente	51
Figura 5.2 - Pinos do EMC1001	53
Figura 5.3 - Esquema de ligação EMC1001, Raspberry Pi	54
Figura 5.4 - Detecção do sensor temperatura	55
Figura 5.5 - Fluxograma do programa sensor_temp.py	55
Figura 5.6 - Sincronização cliente-servidor	58
Figura 5.7 - Página de configuração cliente	59
Figura 5.8 - Gráfico <i>RSSI</i> (dBm) em função da <i>distância</i> (m)	62
Figura 6.1 - Estrutura da base de dados do servidor	64
Figura 6.2 - Estrutura das diretorias	65
Figura 6.3 - Estrutura da página principal	66
Figura 6.4 - Alerta de segurança	68
Figura 6.5 - Menu de topo para administrador	68
Figura 6.6 - Menu de administrador	68
Figura 6.7 - Menu de topo para utilizadores	69
Figura 6.8 - Menu de utilizador	69
Figura 6.9 - Página principal	70
Figura 6.10 - Sistema autenticação	71
Figura 6.11 - Sistema de controle de acessos	72
Figura 6.12 - Políticas de cookies	73

Figura 6.13 - Estrutura de comunicação AJAX	74
Figura 6.14 - Página de visualização de utilizadores	75
Figura 6.15 - Página para enviar mensagens	76
Figura 6.16 - Página de gerenciamento de utilizadores	77
Figura 6.17 - Página de gerenciamento de clientes	79
Figura 6.18 - Gerenciamento de relógio	80
Figura 6.19 - Página de configuração do servidor	82
Figura 6.20 - Página de edição de dados pessoais	83
Figura 6.21 - Estrutura find_hosts.php	84
Figura 6.22 - Gerenciamento de redes sem fios	85
Figura 7.1 - Número total de linhas de código no servidor	87

Lista de Tabelas

Tabela 1 - Características do Raspberry Pi.....	25
Tabela 2 - Comando apt-get	28
Tabela 3 - Descrição dos pinos do EMC1001.....	53
Tabela 4 - Endereços do EMC1001	53
Tabela 5 - Registos do sensor de temperatura EMC1001.....	54
Tabela 6 - Medições <i>RSSI</i> /Distância ao ar livre	61
Tabela 7 - Medições <i>RSSI</i> /Distância dentro edifício.....	62

Lista de Acrónimos

AJAX	Asynchronous Javascript and XML	SPI	Serial Peripheral Interface
API	Application Programming Interface	SQL	Structured Query Language
BD	Base De Dados	SSH	Secure Hash
CAD	Computer Aided Design	SSH	Secure Shell Hash
CORBA	Common Object Request Broker Architecture	SSID	Service Set Identifier
CPU	Central Processing Unit	SVG	Scalable Vector Graphics
FTP	File Transfer Protocol	TDoA	Time Difference of Arrival
GPIO	General-Purpose Input/Output	TI	Texas Instruments
GPU	Graphics Processing Unit	ToA	Time of Arrival
GUI	Graphical User Interface	TxD	Transmit Data
HTTP	Hypertext Transfer Protocol	UART	Universal Asynchronous Receive Transmit
HTTPS	Secure Hypertext Transfer Protocol	UPS	Uninterruptible Power Supply
I2C	Inter-Integrated Circuit		
IIOIP	Internet Inter-ORB Protocol		
INS	Interoperable Naming Service		
IPS	Indoor Positioning Systems		
ISP	Internet Service Provider		
LBS	Location Based Services		
LCD	Liquid Crystal Display		
MCU	Micro-controller Unit		
MISO	Master In - Slave Out		
MOSI	Master Out - Slave In		
NMAP	Network Mapper		
ORB	Object Request Broker		
P2P	Pear-to-Pear		
PHP	Hypertext Preprocessor		
POA	Portable Object Adapter		
RAM	Random Access Memory		
RCA	Radio Corporation of America		
RF	Radio frequency		
RSSI	Received Signal Strength Indicator		
RTC	Real Time Clock		
RxD	Receive Data		
SCL	Serial Clock		
SDA	Serial Data		
SDRAM	Synchronous dynamic RAM		
SoC	Sistem on Chip		

1. Introdução

1.1. Enquadramento

Este trabalho de dissertação consiste no desenvolvimento de um sistema de localização de pessoas, dentro de um edifício, em tempo real. O sistema é constituído por diferentes equipamentos de hardware cada um com a sua função específica. A Figura 1.1 representa a topologia utilizada no desenvolvimento do sistema.

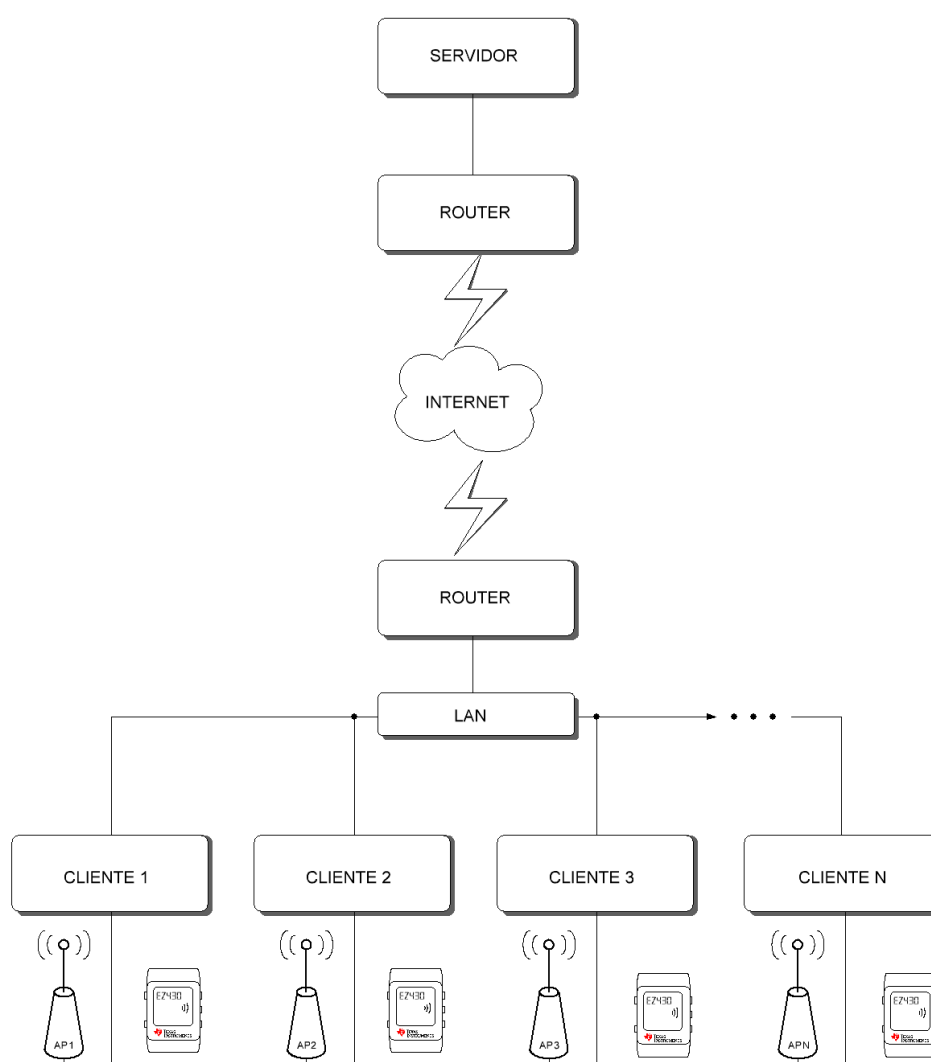


Figura 1.1 - Diagrama do sistema

No topo do sistema encontra-se o dispositivo principal, designado por “servidor”. É constituído por um Raspberry Pi e tem a função de coordenar todo o sistema. Para isso tem de possuir a capacidade de realizar as seguintes tarefas:

- gerenciar utilizadores;
- gerenciar dispositivos;
- servidor de páginas web;
- servidor de base de dados;
- verificar novos dispositivos no sistema.

Em segundo plano na hierarquia do sistema, encontram-se os dispositivos que fazem a interface entre a rede fixa e a rede simplicITI, designados por “cliente”. Estes são constituídos por Raspberry Pi's que têm de realizar as seguintes tarefas:

- receber o sinal dos relógios, através de três AP's simplicITI;
- identificar o(s) relógio(s) presentes no alcance rádio;
- triangular as potências recebidas em cada AP;
- escrever os dados no dispositivo principal, servidor, a cada minuto.

Em terceiro plano na hierarquia do sistema encontra-se o relógio da Texas Instruments, o qual terá de realizar as seguintes tarefas:

- enviar periodicamente um sinal contendo o seu endereço;
- apresentar mensagens, provenientes do servidor.

A sincronização entre o servidor e os clientes pode ser feita de duas formas distintas. Se o servidor estiver fora da rede local, ligado à internet sendo utilizado como se de um serviço web se tratasse. O(s) cliente(s) encontram-se ligados através de uma rede local tcp/ip com acesso à internet. O sistema está idealizado para poder suportar um número infinito de cliente(s) tendo apenas como limitação o poder computacional do servidor bem como a velocidade de transmissão de dados pela rede. Se o servidor estiver dentro da mesma rede que os clientes, poder-se-á estabelecer a sincronização utilizando o protocolo de comunicação CORBA 1.2. A seleção desta opção poder-se-á fazer na página de configuração do servidor, descrita no Capítulo 6.1.10.

A visualização da posição dos diferentes relógios é feita através de uma página web, através de um browser num computador ou em qualquer dispositivo móvel. Nessa página além da visualização das posições é possível:

- Gerir utilizadores, adicionar, apagar e atribuir privilégios de visualização.
- Gerir dispositivos cliente, adicionar e apagar.
- Definir relógios a utilizadores.

1.2. O problema em estudo e a sua relevância

A localização no interior é um tema que ainda não conseguiu obter o mesmo grau de sucesso/precisão que o seu homónimo exterior, o GPS. Por essa razão existem numerosas abordagens para tentar colmatar esta lacuna. Neste trabalho de dissertação será referida algumas dessas soluções e irá ser desenvolvida uma solução própria baseada nestas técnicas.

1.3. Objetivos e contribuição da dissertação

Com este trabalho de dissertação pretendo contribuir para o desenvolvimento de um protótipo funcional de um sistema de localização de pessoas no interior de um edifício, que pelo menos tenha a precisão a nível da sala.

1.4. Visão geral e organização da dissertação

Esta dissertação está organizada por capítulos seguindo o organograma temporal do trabalho desenvolvido.

Os comandos utilizados são apresentados com tipo de letra *Calibri Light* tamanho 10 para serem facilmente distinguidos do restante texto.

No Capítulo 3 descreve-se o hardware utilizado e no Capítulo 4 encontra-se descrito o processo de instalação e configuração de todos os programas necessários para o desenvolvimento dos programas que constituem o sistema de localização desenvolvido.

Nos Capítulos 0 e 6 descrevem-se os programas escritos para o servidor do sistema e para cada um dos clientes.

2. Estado da arte

2.1. Introdução

A localização dentro de edifícios é a tarefa de inferir a posição de um dispositivo móvel dentro de um edifício.

Quando o tema é localização dentro de edifícios, inúmeras questões se levantam, devido principalmente ao facto da estrutura de um edifício ser muito mais complexa do que um sistema rodoviário (Labrador, Pérez, & Wightman, 2011).

A forma e o método que se escolhe aquando do dimensionamento de um sistema LBS irá depender dos diferentes aspetos que o sistema necessite. Aspetos como: o grau de precisão, a tolerância a erros de localização, o custo de implementação do sistema, a privacidade dos utilizadores, entre outros.

De frisar que um sistema que necessite de um grau de tolerância a erros de localização pequeno, poderá fazer perder tempo e dinheiro devido a um erro de localização.

Nestes sistemas, quando é utilizado para localizar pessoas, é necessário ter em conta a privacidade dos utilizadores, pois esta tem de ser respeitada.

2.2. Métodos de localização

Existem uma grande variedade de técnicas de posicionamento e de sistemas de posicionamento. No entanto, existem pouco métodos e algoritmos para inferir a posição a partir de medições.

O problema central na inferência da posição é que utiliza a medida de grandezas físicas, sujeitas a interferência, erros sistemáticos de leitura inerentes aos processos de medição e às condições externas do sistema de leitura.

2.2.1. Trilateração

Trilateração é um método utilizado para determinar a interseção de três superfícies esféricas sabendo a posição dos centros e os raios das esferas. O método de Trilateração é utilizado principalmente para ToA e RSSI (Ahson & Ilyas, 2011).

Esta técnica depende geralmente da utilização da medida da potência do sinal RSSI ou na medição do intervalo de tempo que o sinal demora desde que é enviado até ser recebido ToA, para determinar o raio de cada esfera.

No caso da utilização do método ToA, o raio é determinado com a utilização da equação da velocidade:

$$v = \frac{\Delta x}{\Delta t} \quad (1)$$

onde v é a velocidade de propagação de uma onda eletromagnética e tem o valor de $3 \times 10^8 \text{ m s}^{-1}$. A Equação (1) pode assim ser reescrita da seguinte forma:

$$\Delta x = 3 \times 10^8 \times \Delta t \quad (2)$$

O valor do deslocamento Δx corresponde à distância entre a posição de onde o sinal é emitido e a posição de onde o sinal é recebido. Como as ondas eletromagnéticas se propagam em linha reta podemos dizer que o deslocamento corresponde à distância percorrida pela onda eletromagnética. Como este método mede o tempo entre dois dispositivos, ambos os relógios têm de estar sincronizados, caso contrário a posição determinada terá um erro muito grande, afastando-se da posição real (Ahson & Ilyas, 2011).

Matematicamente, a localização estimada num espaço tridimensional (x, y, z) será a solução do seguinte sistema de equações:

$$r_1^2 = (x - x_{c1})^2 + (y - y_{c1})^2 + (z - z_{c1})^2 \quad (3)$$

$$r_2^2 = (x - x_{c2})^2 + (y - y_{c2})^2 + (z - z_{c2})^2 \quad (4)$$

$$r_3^2 = (x - x_{c3})^2 + (y - y_{c3})^2 + (z - z_{c3})^2 \quad (5)$$

Onde (x_{c1}, y_{c1}, z_{c1}) , (x_{c2}, y_{c2}, z_{c2}) e (x_{c3}, y_{c3}, z_{c3}) representam as localizações das três fontes emissoras de sinal para o qual o dispositivo móvel referencia a sua localização. Estas coordenadas são os centros das esferas cuja intersecção irá representar a localização estimada do dispositivo móvel. Por outro lado, R_1 , R_2 e R_3 correspondem às distâncias calculadas.

Considerando a Figura 2.1, onde AP_1 , AP_2 e AP_3 são as três fontes emissoras de sinal, localizadas nas posições $(0,0,0)$, $(20,0,0)$ e $(20,20,0)$ respetivamente e DM o dispositivo móvel na posição desconhecida (x, y, z) , sabendo que o quadrado dos raios a cada um dos APs é de 200 é possível calcular a posição exata do dispositivo móvel.

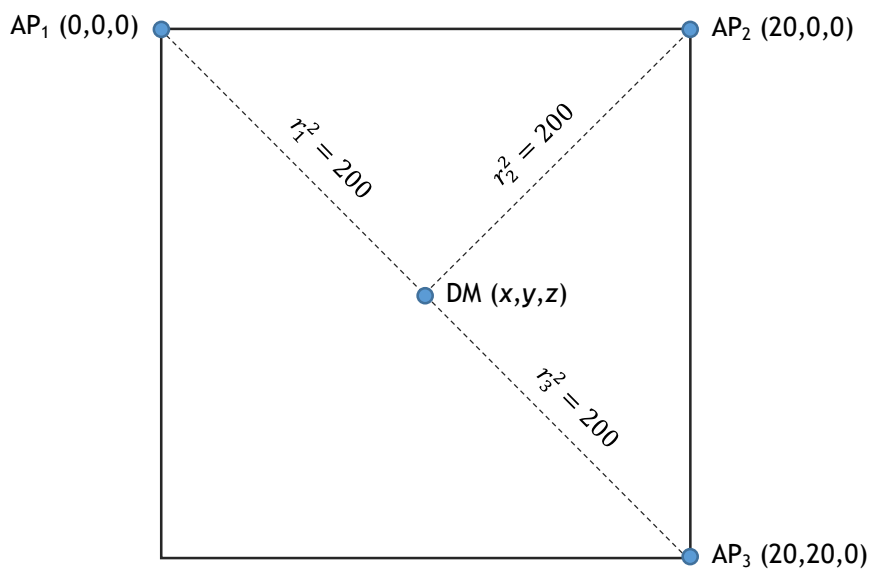


Figura 2.1 - Exemplo de trilateração

Pela resolução do sistema de equações (3), (4) e (5) obtém-se a posição do dispositivo móvel.

$$\begin{cases} (0,0,0) \\ (20,0,0) \\ (20,20,0) \end{cases} \Rightarrow \begin{cases} 200 = x^2 + y^2 + z^2 \\ 200 = (x-20)^2 + y^2 + z^2 \\ 200 = (x-20)^2 + (y-20)^2 + z^2 \end{cases} \Leftrightarrow \begin{cases} z^2 = 200 - x^2 - y^2 \\ - \\ - \end{cases} \Leftrightarrow$$

$$\Leftrightarrow \begin{cases} 0 = (x-20)^2 - x^2 \\ - \\ - \end{cases} \Leftrightarrow \begin{cases} x = 10 \\ - \\ - \end{cases} \Leftrightarrow \begin{cases} 200 = 100 + y^2 + z^2 \\ - \\ 200 = (10-20)^2 + (y-20)^2 + z^2 \end{cases} \Leftrightarrow$$

$$\Leftrightarrow \begin{cases} 100 = y^2 + z^2 \\ 0 = 300 + y^2 - 40y + z^2 \\ - \end{cases} \Leftrightarrow \begin{cases} z^2 = 100 - y^2 \\ -400 = -40y \\ - \end{cases} \Leftrightarrow \begin{cases} 200 = 100 + y^2 + z^2 \\ 0 = 300 + y^2 - 40y + 100 - y^2 \\ - \end{cases} \Leftrightarrow$$

$$\Leftrightarrow \begin{cases} y = 10 \\ - \\ - \end{cases} \Leftrightarrow \begin{cases} - \\ - \\ 200 = 100 + 100 + z^2 \end{cases} \Leftrightarrow \begin{cases} x = 10 \\ y = 10 \\ z = 0 \end{cases}$$

Neste exemplo a posição do dispositivo móvel é (10,10,0) como se pretendia demonstrar, pois o dispositivo móvel foi colocado no centro do quadrado.

Este será o método utilizado para a determinação da posição, descrito no Capítulo 5.5, com a diferença que a fonte emissora de sinal será o dispositivo móvel e os pontos de acesso serão os recetores.

2.2.2. Multilateração

Este método é uma variante do método lateração e difere no sentido em que neste não são utilizadas as medidas estimadas, mas sim a diferença das medidas estimadas (Labrador, Pérez, & Wightman, 2011).

A multilateração estima a posição usando medidas de TDoA de três ou mais fontes emissoras de sinal. Também é conhecido como lateração hiperbólica e refere-se ao processo de localização de um objeto através da interseção de hiperboloides, que resultam quer de computação com precisão a TDoA de um sinal enviado a partir desse objeto e a entrar em três ou mais recetores, ou medindo a TDoA de um sinal transmitido a partir de três ou mais transmissores sincronizados e chegando ao objeto recetor.

Como não há necessidade de medições absolutas de ToA, a sincronização entre os terminais de sinalização não é necessária.

Matematicamente, a solução determina a localização de um objeto num espaço tridimensional, (x, y, z) , através da transmissão de um sinal para um conjunto de quatro recetores com locais conhecidos (x_{c1}, y_{c1}, z_{c1}) , (x_{c2}, y_{c2}, z_{c2}) , (x_{c3}, y_{c3}, z_{c3}) e (x_{c4}, y_{c4}, z_{c4}) . Os tempos de viagem do sinal a partir do objeto móvel para cada um dos quatro recetores, t_1 , t_2 , t_3 , e t_4 , respetivamente, é igual à distância entre o objeto e cada um dos recetores pela velocidade de propagação do sinal. Para simplificar, consideramos que a velocidade de propagação é igual à velocidade da luz no vácuo, c . Ao resolver as equações seguintes, podemos obter a localização aproximada do objeto (x, y, z) :

$$t_1 = \frac{\sqrt{(x - x_{c1})^2 + (y - y_{c1})^2 + (z - z_{c1})^2}}{c} \quad (6)$$

$$t_2 = \frac{\sqrt{(x - x_{c2})^2 + (y - y_{c2})^2 + (z - z_{c2})^2}}{c} \quad (7)$$

$$t_3 = \frac{\sqrt{(x - x_{c3})^2 + (y - y_{c3})^2 + (z - z_{c3})^2}}{c} \quad (8)$$

$$t_4 = \frac{\sqrt{(x - x_{c4})^2 + (y - y_{c4})^2 + (z - z_{c4})^2}}{c} \quad (9)$$

Para simplificar a demonstração, considera-se que o quarto recetor se encontra na origem do referencial:

$$(x_{c4}, y_{c4}, z_{c4}) = (0,0,0) \quad (10)$$

Pode-se obter o TDoA do sinal de chegada, do sinal que chega ao quarto recetor e dos que chegam aos outros recetores através das seguintes equações (Werner, 2014):

$$\Delta t_1 = t_1 - t_4 = \frac{\sqrt{(x - x_{c1})^2 + (y - y_{c1})^2 + (z - z_{c1})^2} - \sqrt{x^2 + y^2 + z^2}}{c} \quad (11)$$

$$\Delta t_2 = t_2 - t_4 = \frac{\sqrt{(x - x_{c2})^2 + (y - y_{c2})^2 + (z - z_{c2})^2} - \sqrt{x^2 + y^2 + z^2}}{c} \quad (12)$$

$$\Delta t_3 = t_3 - t_4 = \frac{\sqrt{(x - x_{c3})^2 + (y - y_{c3})^2 + (z - z_{c3})^2} - \sqrt{x^2 + y^2 + z^2}}{c} \quad (13)$$

2.2.3. Triangulação

Em contraste com a trilateração, que utiliza as distâncias ou medições absolutas de ToA de três ou mais locais, ou com a multilateração, que utiliza medições de TDoA de três ou mais locais, a triangulação é um método para determinar a localização de um dispositivo móvel pela medição do ângulo entre o dispositivo móvel e duas ou mais fontes emissoras de sinal com localizações conhecidas, em vez de medir distâncias ao ponto de localização do dispositivo móvel. O ponto de localização do dispositivo móvel pode então ser fixado como o terceiro ponto de um triângulo com um lado conhecido e dois ângulos conhecidos. O princípio da triangulação é baseado nas leis da trigonometria plana, que afirmam que, se um dos lados e dois ângulos de um triângulo são conhecidos, os outros dois lados e o ângulo podem ser facilmente calculados e a localização de um ponto é geralmente determinada medindo os ângulos de pontos conhecidos (Werner, 2014).

2.2.4. Detecção por proximidade

Este método determina a posição do dispositivo móvel pela proximidade deste com posições previamente conhecidas. A captura de um sinal de um ponto de acesso limita a posição do dispositivo móvel ao raio de ação do ponto de acesso. Consequentemente, este algoritmo não permite determinar as coordenadas da posição do dispositivo móvel, apenas informa que está dentro da sua área de ação. Mas se se utilizar vários pontos de acesso é possível encontrar uma área mais pequena, área que corresponde à interseção dos vários raios de ação dos pontos de acesso, determinando-se assim a posição do dispositivo móvel. Quanto maior for o número de pontos de acesso menor será o erro da determinação da posição do dispositivo móvel.

Uma forma minimalista de representar esta forma de localização está representada na, Figura 2.2 - Método de deteção por proximidade que utiliza apenas três pontos de acesso.

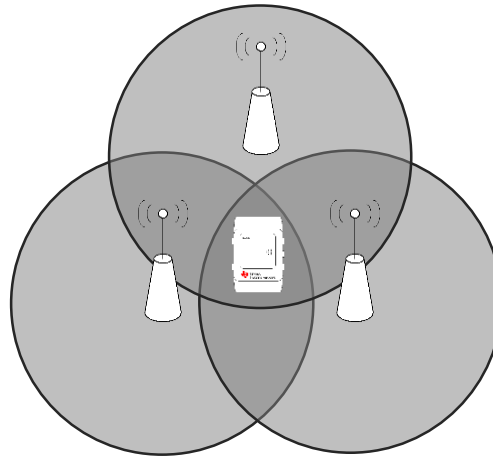


Figura 2.2 - Método de deteção por proximidade

A deteção por proximidade conta com os seguintes métodos:

2.2.5. Navegação Inercial

Os sistemas que utilizam este método estimam a posição utilizando apenas as medições feitas com sensores que estão instalados nos dispositivos móveis.

Sabendo a posição, a velocidade e orientação inicial, atualizam a sua posição medindo a aceleração e a rotação.

A vantagem mais importante deste método reside no facto de o dispositivo móvel poder funcionar de forma completamente autónoma, não dependendo de qualquer infra-estrutura. A maior desvantagem deste método é que a localização de um dispositivo não pode ser observada diretamente de dentro do referencial inercial do dispositivo móvel. Assim, erros de medição irão acumular-se ao longo do tempo tornando o sistema inútil ao longo do tempo.

2.2.6. Fingerprinting

Os métodos descritos anteriormente são baseados numa relação entre uma grandeza física medida e a localização.

Este método não depende de qualquer relação desse tipo, é baseado na reprodutibilidade de padrões.

Desta forma, num ambiente mais complexo onde é mais difícil medir grandezas físicas, para inferir a distância, tem em conta as características específicas do mesmo permitindo que seja distinguido de outros locais com características diferentes. Este método denomina-se de fingerprinting (Werner, 2014).

2.3. Soluções disponíveis

Atualmente existem inúmeros sistemas de localização dentro de edifícios, quer seja para localizar pessoas quer seja para localizar objetos. A grande maioria são soluções comerciais, mas há também algumas soluções gratuitas.

O Redpin apresenta um sistema de posicionamento de interiores de código aberto que foi desenvolvido com o objetivo de fornecer a precisão ao nível de uma sala.

É um sistema baseado no método de fingerprinting, não fornece as coordenadas da posição, mas identifica a localização do dispositivo móvel dentro de uma sala do edifício.

Os dispositivos móveis são telemóveis android ou iphone que apresentam a sua localização numa aplicação. Está disponível no seu sítio oficial (Redpin, 2016).

O Redpin é constituído por dois componentes: um componente sniffer, aplicação que corre num telemóvel android ou iphone, que recolhe informações sobre diferentes dispositivos sem fio para criar uma fingerprint, e um componente de localização, que armazena fingerprints medidos num repositório e contém o algoritmo para localizar um dispositivo móvel. Para permitir fácil uso e acesso para o utilizador o sistema é executado num servidor central, implementado em Java SE 6.0 e SQLite ou MySQL como base de dados (Redpin, 2016).

Andreas Fink propõe no seu artigo o uso do relógio da Texas Instruments eZ430 Chronos para estimar a sua localização dentro de um edifício (Fink, Lange, & Beikirch, 2012).

A empresa Skyhook dedica-se à criação de tecnologia que, utilizando o sinal WiFi existente dentro de um ambiente, deteta a localização de um dispositivo móvel. Segundo a descrição da empresa, a tecnologia usada permite localizar em tempo real um dispositivos móvel, com uma precisão até 2,5 metros, usando os sinais WiFi existente nos edifícios (Skyhook, 2015).

3. Hardware

No desenvolvimento do trabalho prático desta dissertação foi utilizado o hardware que se apresenta a seguir.

3.1. Protocolo SimpliciTI

SimpliciTI é um protocolo de comunicação de rádio frequência, propriedade da Texas Instruments. Este possui as seguintes características:

- baixo consumo: RF network protocolo com suporte a sleep mode;
- baixo custo: utiliza memória FLASH até 8K e 1K RAM dependendo da configuração;
- flexível: comunicação em estrela com extensor e/ou comunicação p2p;
- simples: utiliza um núcleo muito simples;
- versátil: MSP430+CC110x/2500, CC1110/2510, CC1111/CC2511, CC2430, CC2520.

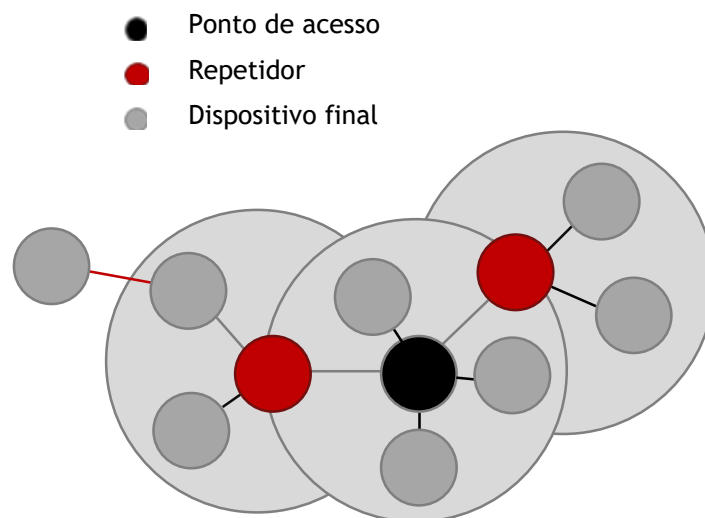


Figura 3.1 - Rede SimpliciTI

3.1.1. Relógio eZ430 Chronos

O eZ430-Chronos é um relógio desportivo de desenvolvimento wireless baseado no integrado CC430F6137. Possui ligação sem fios, um mostrador LCD de 96 segmentos, sensor de pressão, acelerómetro para deteção de movimento em três eixos, medidor de bateria. Este relógio pode comunicar com um PC via placa de desenvolvimento USB CC1111 (Instruments, Chronos: Wireless development tool in a watch, 2015).



Figura 3.2 - Relógio eZ430 Chronos

(Chronos: Wireless development tool in a watch, 2016)

3.1.2. Ponto de acesso CC1111

O ponto de acesso CC1111 USB, Figura 3.3, facilita a ligação entre o Raspberry Pi e a rede de rádios sub-1GHz, tais como os CC1101, CC1120, CC1110 e CC430. O ponto de acesso CC1111 USB pode operar nas bandas de frequência 868 MHz e 915 MHz (Instruments, Sub-1 GHz RF System-on-Chip with Integrated USB, 2015).

Principais características:

- Suporta transferências de dados de 1,2 Kbps a 500 Kbps
- MCU com 8/16/32KB de memória Flash e 1/2/4KB de memória RAM
- Interface USB 2.0
- Co-processador de encriptação AES-128
- Corrente em “Sleep Mode” de apenas 0,3 μ A

O ponto de acesso pode ainda ser programado como dispositivo de captura de pacotes de dados oriundos de dispositivos SimpliciTI atuando como “sniffer”.

O circuito integrado CC1111 da Texas Instruments foi o primeiro sub-1 GHz RF SoC com interface USB, habilitando uma ligação rápida com um computador.

Quando ligado a uma porta USB do Raspberry Pi, este cria uma porta COM virtual, acessível a partir de `/dev/ttyACM0`. Todos os dados são transferidos como se de uma porta série se tratasse.



Figura 3.3 - Ponto de acesso CC1111
(CC1111 USB Evaluation Module Kit 868/915 MHz , 2016)

Foram ainda utilizadas mais dois destes pontos de acesso, configurados como repetidores, para a obtenção de três valores de RSSI.

3.2. Raspberry Pi

O Raspberry Pi é uma pequena placa do tamanho aproximado de um cartão de crédito com as dimensões de 85,60 mm × 53,98 mm × 17 mm e de massa apenas 45 g. O Raspberry Pi é construído na base de um SoC, “system-on-a-chip”, o Broadcom BCM2835. O BCM2835 é um processador multimédia muito utilizado em aplicações móveis e em sistemas embutidos.

Em complemento a este chip foram incluídos vários outros componentes para dar suporte a USB, RCA, e armazenamento de cartão SD.

Embora tenha um tamanho pequeno, possui características impressionantes, as quais se descrevem na Tabela 1 (Dennis, 2013).

Tabela 1 - Características do Raspberry Pi

	Modelo A	Modelo B
SoC	Broadcom BCM2835 (CPU, GPU, DSP e SDRAM)	
CPU	700 Mhz núcleo ARM1176JZF-S (família ARM11)	
GPU	Broadcom VideoCore IV, OpenGL ES 2.0, 1080p30	
Memória (SDRAM)	256 MB	512 MB
Portas USB 2.0	1	2
Saídas de vídeo	Composite RCA, HDMI	
Armazenamento	Slot para cartões SD, MMC, SDIO	
Saídas de som	3,5 mm jack, HDMI	
Rede LAN	Não disponível	Ethernet 10/100 (RJ45)
GPIO	8	
Periféricos (baixo-nível)	UART, I2C bus, SPI bus com selecção de 2 chips	
Corrente (Potência)	500 mA (2,5 W)	700 mA (3,5 W)
Alimentação	5 volt via MicroUSB ou GPIO header	
Dimensões	85,60 mm × 53,98 mm × 17 mm	
Sistemas Operativos	Debian GNU/Linux, Fedora, Arch Linux	

O Raspberry Pi possui um “GPIO Header” com 23 pinos descritos na Figura 3.4.














Descrição	Pino		Pino	Descrição
3,3 VDC	1		2	5,0 VDC
SDA 0 (I2C)	3		4	NC
SCL 0 (I2C)	5		6	0V Massa
GPIO 7	7		8	TxD (UART)
NC	9		10	RxD (UART)
GPIO 0	11		12	GPIO 1
GPIO 2	13		14	NC
GPIO 3	15		16	GPIO 4
NC	17		18	GPIO 5
MOSI	19		20	NC
MISO	21		22	GPIO 6
SCLK	23		24	CE 0
NC	25		26	CE 1

Figura 3.4 - GPIO HEADER

A utilização desta placa em detrimento de outras existentes deveu-se essencialmente à existência de uma grande comunidade on-line, onde rapidamente se esclarece qualquer dúvida, e ao facto de existirem duas unidades disponíveis para trabalhar na universidade. De referir que existem outras placas com hardware semelhante, algumas delas menos dispendiosas, que se poderiam utilizar tornando este equipamento ainda mais económico. Alguns exemplos de outras placas que existem:

- BeagleBone Black;
- ODROID-C1+ da Hardkernel;
- UDOO Dual Basic;
- NanoPC-T1;
- pcDuino4 Set-Top Box;
- Banana Pi;
- Orange pi pc;

4. Instalação e configuração de software

4.1. Introdução

Como o sistema é constituído por dois tipos distintos de dispositivos, o “servidor” e o “cliente” há a necessidade de desenvolver duas distribuições diferentes, pois cada uma tem a sua tarefa no sistema.

A configuração é praticamente igual nas duas distribuições e neste capítulo sempre que for diferente será devidamente identificada.

4.2. Comandos utilizados

O Raspbian, como qualquer sistema operativo Linux traz pré-instalados inúmeros aplicativos úteis para a configuração do sistema. De seguida descrevem-se os mais importantes, no desenvolvimento prático desta dissertação.

4.2.1. sudo

O comando `sudo` permite a utilizadores comuns obterem privilégios de outro utilizador que em geral é o super utilizador, para executar tarefas específicas dentro do sistema de maneira segura e controlável pelo administrador. O nome é uma forma abreviada de se referir a “substitute user do”.

4.2.2. apt-get

O `apt-get` é um recurso desenvolvido originalmente para a distribuição Debian que permite a instalação e a atualização de programas e bibliotecas no Linux de uma forma fácil.

Em qualquer sistema operacional há a necessidade de se instalar programas para que o uso do computador faça sentido. O problema é que, muitas vezes, o processo de instalação é difícil, obrigando o utilizador a procurar as dependências ou editar scripts.

Para lidar com esse tipo de problemas, os programadores da distribuição Debian Linux criaram o APT (Advanced Packaging Tool), uma ferramenta para instalar e atualizar pacotes e as suas dependências de maneira rápida e prática.

Quando o `apt-get` é utilizado no Linux, o comando consulta um ficheiro denominado `sources.list`, geralmente disponível no diretório `/etc/apt/`. Esse arquivo informa onde estão os locais (repositórios) nos quais ele obterá os pacotes requisitados pelo utilizador.

Pelo arquivo `/etc/apt/sources.list` o APT sabe onde procurar os pacotes solicitados. Para ele conhecer o que está instalado no sistema operacional precisa saber exatamente o que procurar. Se não tiver essas informações, o `apt-get` não saberá que um determinado pacote essencial a um programa não está instalado e não providenciará a sua instalação, por exemplo. Para lidar com essa situação, o `apt-get` utiliza um banco de dados próprio e compara-o aos repositórios registados no `sources.list` (Wester, 2016).

Na Tabela 2 apresentam-se os comandos e as respetivas funções do comando `apt-get`.

Tabela 2 - Comando `apt-get`

Comando	Descrição
<code>sudo apt-get update</code>	Atualiza o ficheiro <code>souces-list</code> .
<code>sudo apt-get install nome_do_programa</code>	Instala o programa.
<code>sudo apt-get remove nome_do_programa</code>	Remove o programa.
<code>sudo apt-get --purge remove nome_do_programa</code>	Remove o programa incluindo os ficheiros de configuração.
<code>sudo apt-get upgrade</code>	Atualiza o sistema operativo.

4.2.3. nano

A configuração de um sistema operativo Linux pode ser feita pela edição dos ficheiros de configuração dos respetivos programas que se querem configurar. Para tal é necessário utilizar um editor de texto. O mais utilizado é o `vi`, mas por gosto pessoal o editor de texto utilizado na configuração do sistema operativo foi o `nano`.

```
sudo nano nome_do_ficheiro
```

4.2.4. wget

Wget é um programa gratuito que é utilizado para fazer downloads de ficheiro da Web a partir da linha de comandos. Suporta os protocolos HTTP, HTTPS e FTP.

Wget tem a capacidade de seguir os atalhos que se encontram nas páginas HTML e criar uma cópia local dos sítios web, recreando as próprias diretorias do sítio original. O programa wget respeita o “Robot Exclusion Standard” robots.txt (William E. Shotts, 2009).

Exemplo de utilização deste programa:

```
wget url_do_ficheiro
```

4.2.5. ifconfig

O comando ifconfig apresenta diferentes tipos de informações sobre a ou as placas de rede instaladas no sistema. Informações como:

- Tipo de interface.
- Endereço MAC.
- Endereço IPV4.
- Estatísticas de envio e receção de dados.

Além da informação sobre as placas de rede físicas que estão presentes, também são apresentadas as informações sobre o dispositivo de Loopback (lo). Esta é a interface de rede usada para fins internos do servidor (Vugt, 2006).

4.2.6. cloc

O comando cloc, acrónimo de “count lines of code”, é um programa que conta o número de linhas de código especificando a linguagem utilizada, o número de linhas de código, o número de linhas em branco e o número de linhas de comentários.

Para instalar este programa utiliza-se o comando:

```
sudo apt-get install cloc
```

4.3. Sistema Operativo

O Raspberry Pi conta com uma variadíssima quantidade de sistemas operativos, que podem ser obtidos gratuitamente na internet. No seu sítio oficial, à data da escrita desta dissertação, podia-se descarregar os sistemas:

- NOOBS - Sistema de arranque que permite a instalação de diferentes sistemas operativos no mesmo cartão;
- RASPBIAN Debian Wheezy;
- UBUNTU MATE;
- SNAPPY UBUNTU CORE;
- WINDOWS 10 IOT CORE;
- OSMC - Midia Player;
- OPENELEC - Midia Player;
- PINET;
- RISC OS.

Estes sistemas são na sua grande maioria baseados no sistema operativo Linux com a exceção do WINDOWS 10 IOT CORE que é um sistema operativo da Microsoft. Existem ainda muitas outras distribuições não oficiais com os mais diversos projetos. Para a realização do projeto desta dissertação foi utilizado o RASPIAN Debian Wheezy.

Ao contrário dos sistemas de instalação de sistemas operativos para computadores pessoais, a instalação do sistema operativo é facilitada, pois todo o sistema é distribuído numa imagem do cartão SD, consistindo a sua instalação, na simples transferência dessa imagem para o cartão.

4.3.1. Preparação do cartão SD

Depois de se obter o ficheiro da imagem do sistema operativo Raspbian é necessário recorrer a um computador que possua um leitor/gravador de cartões SD para se poder copiar para o cartão SD. Se o computador que se utilize correr o sistema operativo Windows, pode ser utilizado o programa Win32 Disk Imager para copiar a imagem do sistema para o cartão. Este programa é gratuito e de código aberto, podendo ser obtido no seu sítio oficial.

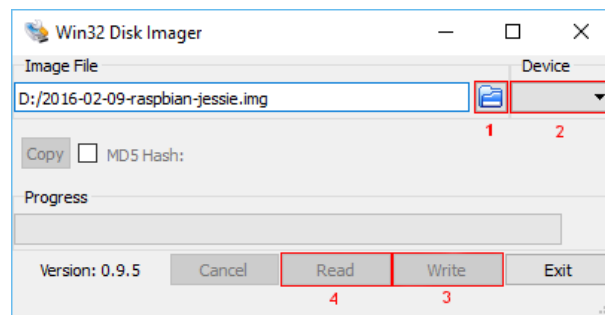


Figura 4.1 - Win32 Disk Imager

Para copiar a imagem, do sistema para o cartão, deve-se em 1 selecionar o ficheiro da imagem que se pretende gravar, em 2 selecionar o cartão SD onde se vai gravar e em 3 iniciar a gravação.

Com este programa é possível realizar o processo inverso, guardar uma cópia de segurança do cartão num ficheiro, bastando para isso clicar em 4, com o cartão a copiar inserido no leitor de cartões.

No sistema operativo Linux utiliza-se o programa `dd` na linha de comandos, para copiar a imagem para o cartão:

```
sudo dd bs=4M if=~/.wheezy-raspbian.img of=/dev/sdd
```

O comando para realizar uma cópia de segurança é:

```
sudo dd bs=4M if=~/.wheezy-raspbian.img of=/dev/sdd
```

Com o sistema operativo copiado, coloca-se o cartão no Raspberry Pi e liga-se a alimentação bem como o cabo de rede ao router. Como o sistema operativo vem com ssh, habilitado por definição, é possível configurá-lo utilizando um cliente ssh não havendo a necessidade de ligar um teclado nem de um monitor. O cliente ssh utilizado é o programa `putty`.

PuTTY é um cliente SSH e Telnet gratuito, de código aberto disponível para sistemas Windows e plataformas Unix.

Ao iniciar este programa, surge uma janela de configuração, onde se tem de especificar o endereço IP do Raspberry Pi que queremos aceder bem como o protocolo de comunicação a utilizar, ssh.

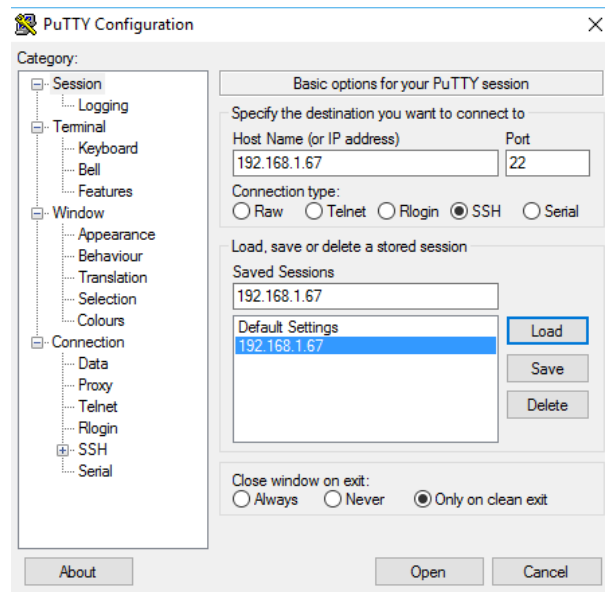


Figura 4.2 - Configuração do cliente ssh, Putty

Depois de se clicar em “Open” surge uma janela apresentada na Figura 4.3. Como não se utiliza um monitor para se saber qual é o IP, pode-se abrir as definições do router e verificar os dispositivos a ele ligados ou utilizar o programa nmap noutra maquina Linux ligada à mesma rede.

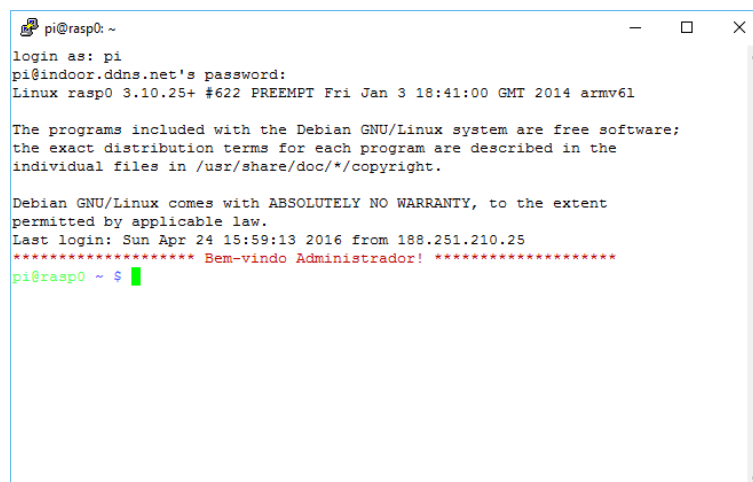


Figura 4.3 - Cliente SSH, Putty

4.3.2. Alteração da palavra-passe

No final da instalação é aconselhável, por motivos de segurança, a alteração da palavra-passe que vem por defeito “raspberry” para o utilizador “pi”. Para isso basta executar o comando `passwd`.

```
passwd

Changing password for pi.
(current) UNIX password:
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
```

Assim como, também se deve alterar a palavra-passe de “root” com o comando `sudo passwd root`.

```
sudo passwd root

Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
```

4.3.3. Configurações iniciais

Para se realizar configurações iniciais do sistema, utiliza-se uma ferramenta escrita por Alex Bradbury especialmente para este sistema, o “raspi-config” que se executa a partir da linha de comandos, `sudo raspi-config`. É necessário o comando “sudo” porque necessita de privilégios de root (administrador). Aparecerá um ecrã azul, com as opções, conforme a Figura 4.4.

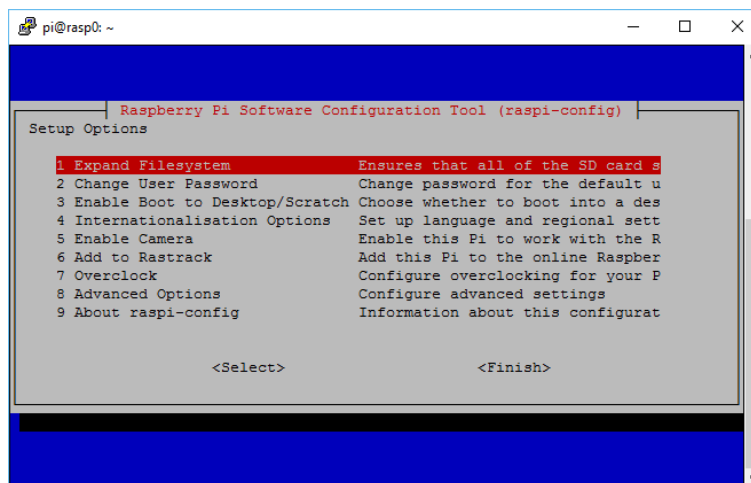


Figura 4.4 - raspi-config

No menu principal, aparece em primeiro lugar a opção “Expand filesystem”, esta serve para expandir a partição do cartão microSD da capacidade da imagem original de aproximadamente 3GB, para a capacidade total do cartão de 8GB. Esta operação necessita que o sistema operativo reinicie para que as alterações sejam aplicadas.

Em segundo lugar aparece a opção “Change user password”, esta serve para mudar a palavra-passe que vem por defeito no Raspbian que é “raspberry” para outra diferente.

Em terceiro lugar aparece a opção “Enable boot to desktop or Scratch” esta serve para definir como arranca o sistema, para a linha de comandos ou diretamente para desktop.

Neste trabalho prático de dissertação, foi selecionada a opção de arrancar para a linha de comandos, pois não irá ser utilizado o ambiente gráfico.

Em quarto lugar aparece a opção “Internationalisation options”, ao selecionar esta opção irá aparecer outro sub-menu como mostra a Figura 4.5. Nestas opções é importante escolher corretamente a localização, pois é a partir daqui que a hora sistema do Raspberry Pi é atualizada para a hora do país.

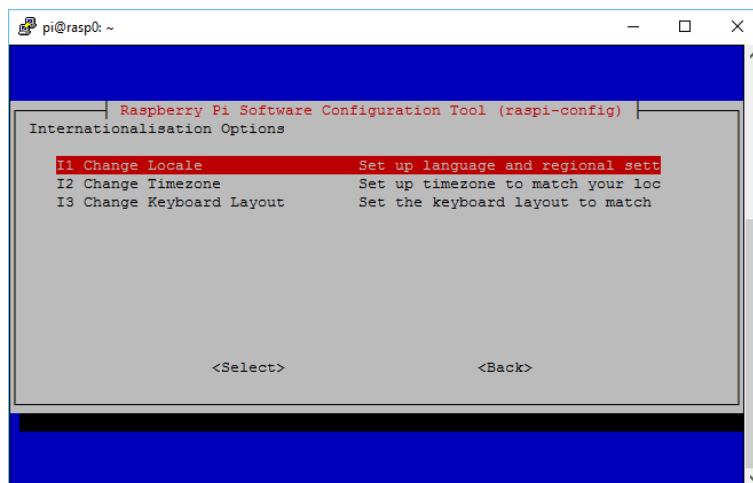


Figura 4.5 - raspi-config, opções internacionais

Change locale - Seleciona a localidade, por exemplo pt_PT.UTF-8 UTF-8.

Change timezone - Seleciona a hora do local.

Change keyboard layout - Seleciona o layout do teclado utilizado.

Voltando ao menu principal aparece em oitavo lugar a opção “Advanced options”, ao selecionar esta opção irá aparecer outro submenu, Figura 4.6.

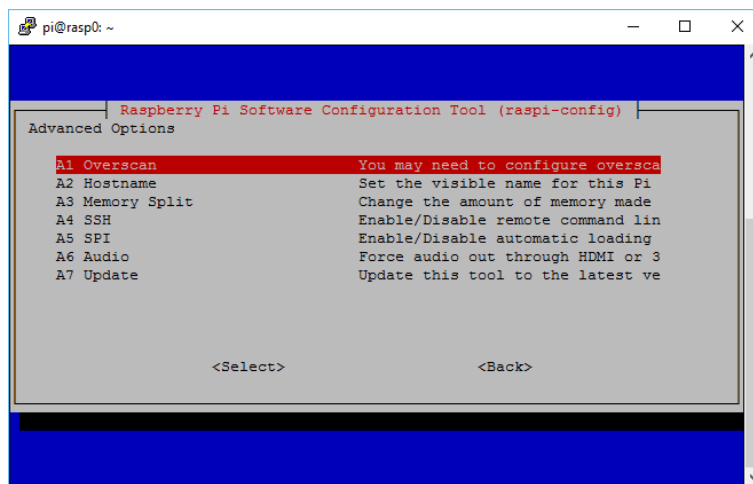


Figura 4.6 - Opções avançadas do raspi-config

Neste submenu aparecem as seguintes opções:

Overscan - Opção que não é utilizada, pois não é utilizada a saída HDMI existente na placa.

Hostname - Define o nome deste dispositivo que aparecerá na rede interna, parâmetro importante, pois cada Raspberry Pi deverá ter um hostname diferente para que não haja conflitos de nomes dentro da rede. Para o servidor foi alterado para “rasp0”.

Memory split - Modifica a quantidade de memória disponível para a GPU, a qual foi colocada no valor mínimo, para rentabilizar recursos do sistema.

SSH - Habilita a ligação de acesso à linha de comandos usando o protocolo ssh. Ssh permite o acesso à linha de comandos remotamente a partir de outro computador. Esta opção vem habilitada por defeito, e é muito importante visto que é a partir deste acesso que se configura e programa todas as aplicações para o funcionamento do trabalho prático.

SPI - Habilita a interface SPI.

I2C - Habilita a interface I2C, que é utilizado para comunicar com o sensor de temperatura EMC1001.

Áudio - Força a saída de áudio pela saída HDMI, opção não utilizada.

Update - Atualiza o sistema para a última versão que está disponível no repositório.

4.3.4. Configuração da mensagem de boas-vindas

A mensagem de boas vindas surge sempre que se efetua acesso a partir de uma ligação SSH. Para personalizar esta mensagem deve-se alterar o ficheiro `.profile` que se encontra na diretoria do respetivo utilizador `/home/pi` com o comando:

```
nano ~/.profile
```

Adicionando a seguinte linha:

```
echo "$(tput setaf 2)*****Bem-vindo Administrador*****"
```

4.3.5. Adicionar utilizador

Uma forma de aumentar a segurança do sistema, bem como rentabilizar o hardware, é a criação de utilizadores. Neste trabalho de dissertação utiliza-se o utilizador “indoor” o qual é responsável por:

- alojar a página de gerenciamento web em `/home/indoor/www`;
- aceder à base de dados MySQL com o nome de utilizador `indoor`;

- configurar o domínio “indoor.ddns.net”;

Para criar o utilizador “indoor” utiliza-se o comando `sudo adduser indoor`. Para adicionar uma palavra-passe ao utilizador utiliza-se o comando `sudo passwd indoor`.

4.3.6. Configuração do hostname

O hostname é o nome de cada dispositivo que se encontra ligado à rede local, por defeito o nome é “raspberrypi”. Quando se liga mais do que um Raspberry Pi à mesma rede local, deve-se alterar o hostname de cada para que não haja conflitos de nomes.

É possível alterar o hostname de várias formas: utilizando o “raspi-config”, a forma mais fácil, ou então editando os ficheiros `/etc/hosts` e `/etc/hostname` na linha de comandos.

No primeiro ficheiro deve-se alterar apenas o nome à frente do ip 127.0.1.1 por aquele que se deseja como hostname.

```
sudo nano /etc/hosts
```

No segundo ficheiro deve-se apenas alterar o nome que contém pelo nome do hostname que se pretende.

```
sudo nano /etc/hostname
```

Finalmente para aplicar as alterações deve-se executar o script `hostname.sh` e reiniciar o sistema com os comandos:

```
sudo /etc/init.d/hostname.sh
```

```
sudo reboot
```

No trabalho prático desta dissertação colocou-se ainda a opção de se poder alterar o hostname de qualquer dispositivo ligado ao servidor a partir da página de gerenciamento de dispositivos. O desenvolvimento deste script encontra-se descrito no Capítulo 6.2..

4.4. Habilitar I2C

Como já foi referido na Tabela 1, o Raspberry Pi suporta o protocolo de comunicação I2C. Este proporciona um caminho de comunicação de velocidade média utilizando apenas dois fios (SCL e SDA). Esta interface irá ser utilizada para conectar um sensor de temperatura. Como esta interface não vem habilitada por definição nas imagens do Raspberry Pi, é necessário habilitá-la e instalar algumas ferramentas de debug.

Nesse sentido, deve-se editar o ficheiro `/etc/modprobe.d/raspi-blacklist.conf` com o comando:

```
sudo nano /etc/modprobe.d/raspi-blacklist.conf
```

Colocando um `#` no início da linha, como mostra nas seguintes linhas, estas são interpretadas como comentários.

```
enabling the SPI and I2C bus:
```

```
#blacklist spi-bcm2708
```

```
#blacklist i2c-bcm2708
```

É necessário definir os módulos I2C no ficheiro `/etc/modules` para que o sistema os carregue aquando do arranque do sistema. Para editar o ficheiro utiliza-se o comando:

```
sudo nano /etc/modules
```

Deve-se adicionar os seguintes módulos em linhas separadas:

```
i2c-dev
```

```
i2c-bcm2708
```

É ainda necessário instalar o pacote `i2c-tools`, que é um conjunto de programas que facilitam o desenvolvimento e debug da aplicação que utiliza este protocolo. Para o instalar utilizam-se os comandos:

```
sudo apt-get update
```

```
sudo apt-get install i2c-tools
```

Este conjunto de ferramentas é constituído pelas seguintes aplicações:

- `i2cdetect` - Deteta um dispositivo I2C e apresenta o endereço desse dispositivo.
- `i2cget` - Adquire dados de um dispositivo I2C.
- `i2cset` - Envia dados para um dispositivo I2C.

Finalmente é altura para ligar o sensor de temperatura. Para o executar de uma forma que não danifique nem o sensor nem a porta I2C do Raspberry Pi, deve-se desligar o Raspberry Pi da alimentação. Como qualquer sistema operativo, este deve ser devidamente encerrado para evitar que algum ficheiro fique corrompido. Para encerrar o sistema operativo utiliza-se o seguinte comando:

```
sudo shutdown -h now
```

Para o desenvolvimento do programa escrito em Python é ainda necessário instalar o módulo `python-smbus`, módulo responsável pela comunicação I2C e o módulo `python-mysqldb`, módulo responsável pela ligação com a base de dados. Para os instalar utilizam-se os seguintes comandos:

```
sudo apt-get install python-smbus
sudo apt-get install python-mysqldb
```

Para o programa conseguir utilizar a interface I2C necessita que o utilizador “pi” tenha privilégios do grupo I2C. Para lhos atribuir executou-se o comando:

```
sudo adduser pi i2c
```

Para o sistema operativo ficar pronto para desenvolver o software, basta agora reiniciá-lo com o comando `sudo reboot`. (Cox, 2014) O programa que comunica com o sensor de temperatura está descrito no Capítulo 5.2..

4.5. Dongle Wireless

Para a utilização de um cliente onde não há acesso à rede local via cabo, previu-se a sua ligação via wireless. Para esse efeito utilizou-se um dongle usb já suportado pelo sistema operativo. Para verificar o seu reconhecimento por parte do sistema operativo utiliza-se o comando `lsusb` o qual mostra todos os dispositivos usb ligados.

```
Bus 001 Device 002: ID 0424:9512 Standard Microsystems Corp.
```

```
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
```

```
Bus 001 Device 003: ID 0424:ec00 Standard Microsystems Corp.
```

```
Bus 001 Device 004: ID 0bda:0179 Realtek Semiconductor Corp.
```

```
Bus 001 Device 005: ID 0451:16a6 Texas Instruments, Inc. BM-USBD1 BlueRobin RF heart rate sensor receiver
```

O dongle wireless é o dispositivo 004 ligado ao Bus 001 que utiliza um chip da Realtek Semiconductor Corp. Para configurar a sua ligação à rede é necessário editar o ficheiro `/etc/wpa_supplicant/wpa_supplicant.conf` adicionando o ssid e a palavra-passe da rede onde se pretende ligar, da seguinte forma:

```
network={
    ssid="Nome_da_Rede"
    psk="Palavra-Passe"
}
```

No final da edição deste ficheiro o wpa-supPLICANT atualizará as alterações em poucos segundos e tentará ligar-se à rede. Caso não aconteça poderá haver a necessidade de reiniciar a interface com `sudo ifdown wlan0` para desligá-lo e `sudo ifup wlan0` para voltar a ligá-lo ou então reiniciar o sistema operativo com `sudo reboot` (Membrey & David, 2012).

Para facilitar o gerenciamento da rede wireless foi escrito um script que pode ser acedido a partir da página de configuração e o qual se descreverá no Capítulo 0. em detalhe.

4.6. Apache

Apache é um programa que tem como tarefa servir páginas web. A sua primeira versão foi lançada em 1995 e ficou reconhecido por conseguir efetuar muito bem a sua tarefa,

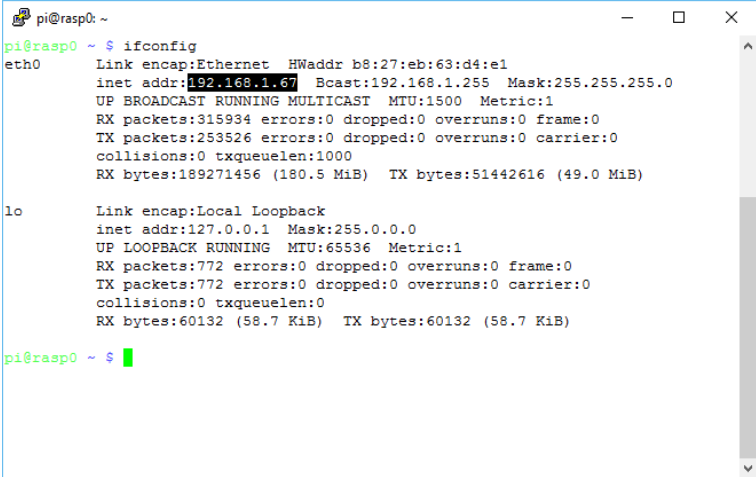
tornando-se um simples, mas poderoso servidor de páginas web gratuito. Estima-se que este esteja a ser utilizado entre 54 e 58 por cento de todas as páginas mundiais (Membrey & David, 2012).

4.6.1. Instalação do Apache

Para a instalação do Apache utiliza-se o comando:

```
sudo apt-get install apache2
```

Após finalizar a instalação e para verificar o correto funcionamento deve-se abrir num browser o ip do Raspberry Pi. Para verificar qual é o ip do Raspberry Pi pode-se utilizar o comando `ifconfig` o qual apresenta todos os dispositivos de rede ligados ao Raspberry Pi bem como o ip atribuído a cada um deles na rede. A Figura 4.7 apresenta o resultado deste comando sendo o ip, 192.168.1.67, o ip da interface `eth0` é o que se deve utilizar no browser.



```
pi@rasp0: ~  
pi@rasp0 ~ $ ifconfig  
eth0      Link encap:Ethernet  HWaddr b8:27:eb:63:d4:e1  
          inet addr:192.168.1.67  Bcast:192.168.1.255  Mask:255.255.255.0  
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1  
          RX packets:315934  errors:0  dropped:0  overruns:0  frame:0  
          TX packets:253526  errors:0  dropped:0  overruns:0  carrier:0  
          collisions:0  txqueuelen:1000  
          RX bytes:189271456 (180.5 MiB)  TX bytes:51442616 (49.0 MiB)  
  
lo        Link encap:Local Loopback  
          inet addr:127.0.0.1  Mask:255.0.0.0  
          UP LOOPBACK RUNNING  MTU:65536  Metric:1  
          RX packets:772  errors:0  dropped:0  overruns:0  frame:0  
          TX packets:772  errors:0  dropped:0  overruns:0  carrier:0  
          collisions:0  txqueuelen:0  
          RX bytes:60132 (58.7 KiB)  TX bytes:60132 (58.7 KiB)  
  
pi@rasp0 ~ $
```

Figura 4.7 - Comando `ifconfig`

A interface loopback, `lo`, é uma interface de rede virtual que o Raspberry Pi utiliza para comunicar com ele próprio. É normalmente utilizado para diagnósticos e reparações do sistema e para se ligar a programas servidores que nele correm. Esta interface assume sempre, no caso do Raspberry Pi, o ip 127.0.0.1 e o hostname de `localhost`. No desenvolvimento de software dos capítulos 0 e 6 este hostname será muito utilizado, pois é com ele que o software faz chamadas a ele próprio (Kirch & Dawson, 2000).

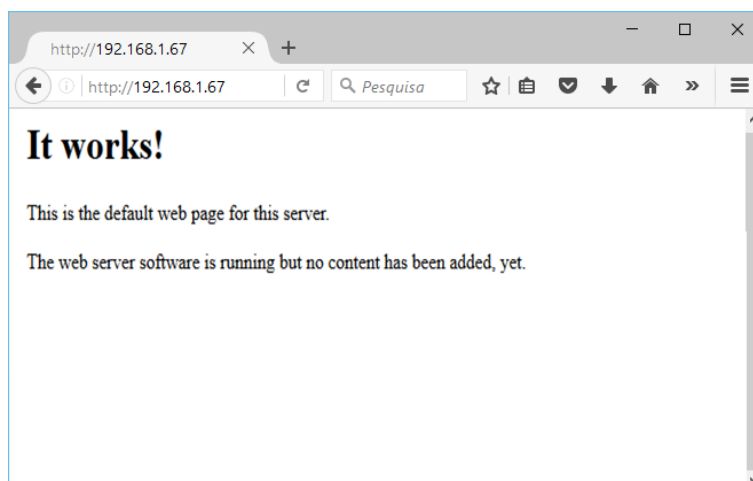


Figura 4.8 - Página por defeito do Apache

A Figura 4.8 apresenta a página que o Apache instala por defeito e que se encontra em `/var/www`. Para que este servidor seja acedido fora da rede local é necessário abrir no router a porta 80 e reencaminhá-la para o ip deste Raspberry Pi.

4.6.2. Configuração do Apache

As configurações dependem da distribuição que se está a desenvolver, servidor ou cliente. Para o cliente não são necessárias configurações adicionais, ficando os ficheiros que se pretendem servir, na pasta que está definida por defeito `/var/www`.

Para o servidor é necessário configurar um servidor virtual pois este vai ser acedido fora da rede local. A configuração de um servidor virtual pode ser efetuada pela edição de ficheiros de configuração ou pela utilização do Webmin, descrita no Capítulo Configuração do virtual host4.10.2.

4.6.3. Permissões do Apache

Com o Apache a funcionar corretamente é necessário dar-lhe autorização para que consiga executar comando que apenas os sudoers têm. Para isso é necessário editar o ficheiro `/etc/sudoers` e adicionar no final do ficheiro as seguintes linhas:

```
apache ALL=(ALL) NOPASSWD: ALL
www-data ALL=(ALL) NOPASSWD: ALL
```

4.7. Hypertext Preprocessor - PHP

PHP é uma linguagem de código aberto “server-side scripting” projetada especificamente para páginas web. O código PHP é inserido dentro do código HTML sendo executado cada vez que a página é visitada. O código PHP é interpretado pelo servidor web que gera HTML como saída. Foi concebido em 1994 por Rasmus Lerdorf. Segundo <http://www.php.net/usage.php>, em janeiro de 2013 encontrava-se instalado em 244 milhões de domínios em todo o mundo.

PHP originalmente significava “Personal Home Page”, mas foi alterado para “PHP Hypertext Preprocessor” (Welling & Thomson, 2009). A versão do PHP instalada é a 5.4.45-0+deb7u1.

4.7.1. Instalação do PHP

A instalação do PHP requer além da instalação do programa, a instalação de um módulo adicional que permite que o php consiga comunicar com a base de dados mysql. Para os instalar utiliza-se o comando:

```
sudo apt-get install php5 php5-mysql
```

4.7.2. Tags PHP

Quando o PHP interpreta um ficheiro procura as tags de início e de fim, `<?php` e `?>`, que informam o PHP quando deve iniciar ou terminar a interpretação do código existente entre essas tags. A interpretação realizada dessa forma, permite que o PHP seja incluído em vários tipos de documentos, pois tudo que está fora dessas tags é ignorado pelo interpretador do PHP.

O PHP também permite a tag curta `<?>`, essa opção está disponível quando habilitada na diretiva `short_open_tag = On` no arquivo de configuração `/etc/php5/apache2/php.ini` (PHP, 2016).

4.7.3. Habilitar o debug do PHP

Quando se desenvolve uma aplicação é sempre útil ter um feedback de erros de programação, para que isso aconteça em php é necessário habilitar essa função, pois vem desabilitada por definição. Para a habilitar é necessário editar o ficheiro `/etc/php5/apache2/php.ini`, procurando as linhas que se seguem abaixo e apagar o ponto e vírgula no início de cada linha.

```
// Enable debugging
```

```
;error_reporting(E_ALL);  
;ini_set('display_errors', true);
```

Para que as novas definições sejam carregadas é necessário reiniciar o apache2, reiniciando o sistema `sudo reboot` ou reiniciar apenas o serviço `apache2`, `sudo /etc/init.d/apache2 restart`. No final do trabalho de desenvolvimento terminado, deve-se desabilitar outra vez esta função, por motivos de segurança (Tahaghoghi & Williams, 2007).

4.8. MySQL

MySQL é o sistema de base de dados de código aberto mais utilizado. MySQL é rápido, estável e de fácil utilização. Pode ser executado nos sistemas operativos Windows, Linux, Mac OS X e Unix. As aplicações que necessitem MySQL podem ser escritas numa grande variedade de linguagens de programação, tais como C, C ++, C #, Java, Perl, PHP, Python, VB e VB.NET. O MySQL está muito bem documentado na Internet, e há muitos livros sobre o tema. MySQL está à beira de repetir, no mercado de banco de dados, o sucesso alcançado pelo Linux no sector sistema operativo. Em combinação com PHP ou Perl, MySQL é cada vez mais usado como sistema de base de dados para páginas web (Kofler, 2005).

4.8.1. Instalação do MySQL

Para instalar o MySQL executa-se o comando:

```
sudo apt-get install mysql-server
```

Durante o processo de instalação é pedido que se introduza a palavra-passe para acesso à base de dados.

4.8.2. Configuração MySQL

Para se conseguir aceder à base de dados a partir de um outro computador é necessário configurar o sistema operativo para dar acesso à base de dados.

Em primeiro lugar deve-se editar o ficheiro `/etc/mysql/my.cnf` com o comando:

```
sudo nano /etc/mysql/my.cnf
```

A linha `bind-address`, está definida por defeito para a interface loopback “127.0.0.1”, dando apenas acesso a si próprio. Para dar acesso a todas as interfaces, deve-se substituir este valor por “0.0.0.0” e reiniciar este serviço (Adelstein & Lubanovic, 2007).

```
sudo service mysql restart
```

Conceder acesso à base de dados fora do servidor não deve ser permitida por razões de segurança e por essa razão, no final do trabalho de desenvolvimento, esta configuração foi revertida.

4.8.3. Adicionar utilizador MySQL

Para adicionar um utilizador novo acede-se ao programa `mysql` com o comando:

```
mysql -u root -p
```

Este comando pergunta pela palavra-passe de utilizador da base de dados que depois de inserida, surge o prompt do `mysql`, `mysql>` (Sverdlov, 2015).

```
mysql> CREATE USER 'indoor'@'localhost' IDENTIFIED BY 'palavra-passe';
mysql> GRANT ALL PRIVILEGES ON indoor . * TO 'novo_utilizador'@'localhost';
mysql> FLUSH PRIVILEGES;
```

4.9. Phpmyadmin

O `phpmyadmin` é uma aplicação web de gerenciamento de base de dados, permite criar, visualizar, editar, apagar e muitas outras funcionalidades, a partir de uma página web. A utilização desta ferramenta permite o gerenciamento de todas as bases de dados de uma forma mais rápida e fácil em detrimento à utilização do terminal.

Para a instalar é necessário executar o comando:

```
sudo apt-get install phpmyadmin
```

A versão instalada é a 3.4.11.1deb2+deb7u1.

4.10. Webmin

O Webmin é uma aplicação web que permite a configuração de um sistema Linux ou UNIX a partir de uma página web.

Normalmente, é necessário efetuar manualmente a configuração de ficheiros e correr comandos no terminal para criar contas de utilizadores, criar servidores de páginas virtuais. O Webmin permite efetuar essas tarefas com uma interface web simplificada, atualizando todas as configurações automaticamente. Esta aplicação torna a administração do sistema bastante facilitada.

As principais características do Webmin:

- criar, editar e apagar contas de utilizadores;
- exportar ficheiros e diretorias para outros sistemas utilizando o protocolo NFS;
- atribuir quotas de disco para cada um dos utilizadores;
- instalar, visualizar e remover pacotes de software;
- alterar o endereço IP, configurações de DNS e configurações de routing;
- configurar uma firewall para proteger o sistema;
- criar e configurar páginas web virtuais para o servidor de páginas Apache;
- criar, editar e apagar tabelas e campos em bases de dados MySQL ou PostgreSQL;
- partilhar ficheiros em sistemas Windows configurando o servidor Samba.

4.10.1. Instalação do Webmin

A instalação desta aplicação não está simplificada pois não se encontra no repositório do sistema operativo não podendo, portanto, utilizar o comando `apt-get`. Em vez disso, deve-se instalar, caso ainda não estejam instaladas, as dependências desta aplicação:

```
sudo apt-get install libnet-ssleay-perl
sudo apt-get install libauthen-pam-perl
sudo apt-get install libio-pty-perl
sudo apt-get install apt-show-versions
```

Realizar o download da página oficial, com o comando `wget`:

```
sudo wget http://downloads.sourceforge.net/project/webadmin/webmin/1.610/webmin_1.610
_all.deb
```

Com o download efetuado procede-se à instalação com o comando dpkg:

```
sudo dpkg --install webmin_1.610_all.deb
```

Para aceder à aplicação basta abrir um browser e apontar para o endereço `https://<ip_da_máquina>:10000`. Para se poder aceder fora da rede local é necessário abrir a porta 10000 no router da rede local.

4.10.2. Configuração do virtual host

O apache tem a capacidade de poder alojar diferentes endereços de internet através da adição de “virtual server”. Para configurar um novo servidor, para o utilizador “indoor”, utiliza-se o webmin e no menu clicar em Servidores, de seguida em Apache e depois surge a página que se apresenta na Figura 4.9.

Figura 4.9 - Servidor virtual

4.11. Nmap

Nmap é acrónimo de “Network Mapper” e é uma aplicação de código aberto, de análise e segurança de uma rede TCP/IP.

O Nmap utiliza pacotes de dados TCP/IP para determinar que dispositivos estão disponíveis na rede anfitriã, que serviços (nome da aplicação e versão) os anfitriões estão a correr, quais sistemas operativos (e versões de SO) eles estão a executar, que tipo de firewalls estão em uso, e dezenas de outras características. Foi projetado para verificar rapidamente grandes redes. Nmap pode ser executado em todos os principais sistemas operativos de computador, e existem versões de terminal ou de interface gráfico (Lyon, 2008).

4.11.1. Instalação do Nmap

Para instalar esta aplicação no Raspbian basta executar, no terminal, o comando:

```
sudo apt-get install nmap
```

4.11.2. Utilização do Nmap

O Nmap é muito útil aquando da verificação dos IPs e dos hostnames dos diferentes dispositivos existentes na rede local, para isso basta executar no terminal nmap seguido do IP que se quer verificar.

De uma forma mais prática coloca-se um asterisco no último valor do IP para que o nmap verifique todos os 255 IPs existentes na rede.

```
nmap 192.168.1.*
```

```
Nmap scan report for rasp0.lan (192.168.1.67)
```

```
Host is up (0.011s latency).
```

```
Not shown: 996 closed ports
```

```
PORT      STATE SERVICE
```

```
22/tcp    open  ssh
```

```
80/tcp    open  http
```

```
2809/tcp  open  corbaloc
```

```
10000/tcp open  snet-sensor-mgmt
```

Neste exemplo, o resultado devolvido pelo terminal consegue ver o hostname “rasp0”, o IP, “192.168.1.67” e os quatro serviços disponíveis, bem como as respetivas portas.

4.12. OmniORB

OmniORB é um Object Request Broker (ORB), em português “agente de requisição de objetos”, que implementa as especificações da versão 2.6 da arquitetura CORBA.

OmniORB é bastante rico em recursos, mas não implementa na sua totalidade as últimas especificações do CORBA. O objetivo é fornecer as partes mais importantes da especificação de uma forma eficiente (omniORB Documentation, 2016).

Os destaques são:

- exemplos para as linguagens de programação C ++ e Python;
- o suporte para a especificação POA;
- o suporte para o INS;
- suporte nativo para o protocolo IIOIP 1.2;
- suporte de thread;
- suporte para interfaces de invocação dinâmica;
- maior controle sobre o gerenciamento de conexão;
- serviço COS e omniNames;
- muitas plataformas são suportadas, incluindo a maioria das plataformas Unix e Windows.

4.12.1.Instalação do OmniORB

Antes de instalar os pacotes e para que o repositório seja atualizado é necessário executar o comando:

```
sudo apt-get update
```

Depois instala-se o libboostalldev com o comando:

```
sudo apt-get install libboost-all-dev
```

Para terminar instalam-se os restantes pacotes:

```
sudo apt-get install vim build-essential openmpi-bin libopenmpi-dev python2.7 python2.7-dev uuid  
uuid-dev openjdk-7-jdk libtool autotools-dev autoconf automake python-omniorb omnievents  
omniidl omniidl-python omniorb omniorb-idl omniorb-nameserver libcos4-dev libomnievents-dev  
libomniorb4-dev python-numpy liblog4cxx10-dev xsdcxx
```

4.12.2.Configuração do OmniORB

Ao terminar a instalação é necessário configurar o omniORB editando dois ficheiros: o `/etc/hosts` e o `/etc/omniORB.cfg`.

No ficheiro `/etc/hosts` deve-se alterar a primeira linha com:

127.0.0.1 localhost localhost.localdomain

No ficheiro `/etc/omniORB.cfg` procura-se a seção `InitRef` e:

- Remover “`DefaultInitRef = corbaloc::`”
- Adicionar “`InitRef = NameService=corbaname::localhost`”
- Adicionar “`InitRef = EventService=corbaloc::localhost:11169/omniEvents`”

Para que as novas configurações fiquem ativas é necessário reiniciar os serviços `omniNames` e `omniEvents`, com os comandos:

```
sudo /etc/init.d/omniorb4-nameserver stop
sudo /etc/init.d/omniorbeventservice stop
sudo /etc/init.d/omniorb4-nameserver start
sudo /etc/init.d/omniorbeventservice start
```

Para finalizar a configuração deve-se criar um atalho simbólico para o `omniidl` com o comando:

```
ln -s /usr/lib/omniidl/omniidl /usr/lib/python2.7/distpackages/omniidl
```

Para as restantes máquinas o procedimento é o mesmo, mas na configuração do ficheiro deve conter o nome da máquina principal, o servidor.

4.13. Configuração do timezone

Para que todo o sistema funcione corretamente é necessário que todos os dispositivos tenham a mesma hora configurada. Para configurar a data e a hora executa-se o comando:

```
sudo dpkg-reconfigure tzdata
```

Com este comando surge um menu onde se pode seleccionar a área geográfica e o fuso horário. Depois basta reiniciar o serviço com o comando:

```
sudo /etc/init.d/ntp restart
```

5. Software desenvolvido no cliente

No lado do(s) cliente(s) foi desenvolvido/atualizado os seguintes programas:

- indoor.c - controla os access point, os relógios e guarda a informação na base de dados.
- Sensor_temp.py - lê a temperatura ambiente e guarda-a na base de dados.
- send_to_server.php - sincroniza os dados, guardados na base de dados, com o servidor.
- página de configuração - possibilidade de configuração dos parâmetros do cliente, Figura 5.7.

Todos estes programas utilizam a base de dados MySQL para guardar e trocarem os dados dos relógios e do servidor. Na Figura 5.1 está apresentada a estrutura da base de dados do cliente. A imagem foi produzida com o programa de demonstração “dbForge Studio for MySQL”, que é um cliente MySQL para o Sistema Operativo Windows.

Table Name	Column Name	Data Type	
sistema_parametros	index	INT	
	index_dispositivo	VARCHAR	
	empresa_id	VARCHAR	
	estado	INT	
	dispositivo_id	VARCHAR	
	hostname	VARCHAR	
	token	VARCHAR	
	sala	VARCHAR	
	ativo	INT	
	url_servidor	VARCHAR	
	date_time	DATETIME	
	temperatura	VARCHAR	
	sala_x	VARCHAR	
	sala_y	VARCHAR	
	ap_id	VARCHAR	
	ap_x	VARCHAR	
	ap_y	VARCHAR	
	re1_id	VARCHAR	
	re1_x	VARCHAR	
	re1_y	VARCHAR	
re2_id	VARCHAR		
re2_x	VARCHAR		
re2_y	VARCHAR		
Constraints			
	PRIMARY		
sistema_relogios	ID_relogio	INT	
	ID_pessoa	VARCHAR	
	nome_pessoa	VARCHAR	
	AP_Registo	VARCHAR	
	ultima_mudanca_batt	TIMESTAMP	
	intervalo	VARCHAR	
	bateria	VARCHAR	
	sala	VARCHAR	
	ativo	INT	
	Constraints		
	PRIMARY		
sistema_historico	ID_relogio	VARCHAR	
	repetidor1	VARCHAR	
	repetidor2	VARCHAR	
	repetidor3	VARCHAR	
	Timestamp	DATETIME	
	Potencia1	VARCHAR	
	Salto1	VARCHAR	
	TiD	VARCHAR	
	ativo	INT	
	Indexes		
	ID_relogio		
sistema_mensagens	index	VARCHAR	
	ID_destino	INT	
	origem_id	VARCHAR	
	Timestamp_registo	TIMESTAMP	
	Mensagem	VARCHAR	
	enviado	INT	
	Indexes		
		ID_destino	
emergencia	ID_relogio	VARCHAR	
	Timestamp	DATE	
	TiD	VARCHAR	

Figura 5.1 - Estrutura da base de dados do cliente

5.1. Programa de controlo da ligação entre o AP e os relógios

O programa que controla os diversos pontos de acesso e relógios foi escrito na linguagem de programação C, originalmente, pelo Professor Doutor Bruno Ribeiro.

Foram construídas novas bases de dados, alteradas as funções de escrita na base de dados e adicionadas novas funções por forma a otimizar o programa com a interface aqui desenvolvida.

Este programa consegue gerenciar quinze relógios. Sempre que um relógio se liga ao ponto de acesso o programa executa as seguintes tarefas:

- guarda na base de dados o estado do relógio como ativo;
- envia para o relógio o tempo das novas sincronizações;
- envia para o relógio a localização do ponto de acesso;
- envia para o relógio uma mensagem, caso a haja na base de dados;
- guarda o valor de RSSI detetado no ponto de acesso e nos repetidores.

Sempre que um relógio sai do raio de ação do ponto de acesso o programa guarda na base de dados que o relógio se encontra no estado de inativo.

O relógio envia para o ponto de acesso a sua identificação e quando é pressionado o botão de emergência envia um pedido de emergência guardando o programa o estado de emergência na base de dados.

O relógio apresenta no seu LCD a localização de onde se encontra, o nome da pessoa que se encontra na base de dados e a mensagem que lhe é enviada.

5.2. Sensor de temperatura

A temperatura é uma das propriedades que influenciam as comunicações via rádio alterando a potência recebida, não podendo por essa razão ser descurada. Além disso, a adição de um sensor de temperatura permite saber a temperatura do local onde cada cliente se encontra, sendo esta apresentada na página de localização (planta do edifício).

O sensor de temperatura utilizado foi o EMC1001-AFZQ-TR da Microchip Technology Inc, que possui uma resolução de 0,25 °C e uma exatidão de 1,5 °C. Utiliza o protocolo de comunicação I2C e o seu endereço é selecionado por meio de uma resistência ligada ao pino 1. Os valores da resistência bem como o valor dos respetivos endereços estão apresentados na Tabela 4. A sua alimentação é de 3,3 V, podendo ser alimentado diretamente do barramento do Raspberry Pi e consome apenas 44 µA quando configurado para uma

conversão por segundo. O esquema de ligação bem como a descrição de cada pino estão apresentados na Figura 5.2 e na Tabela 3 respetivamente.

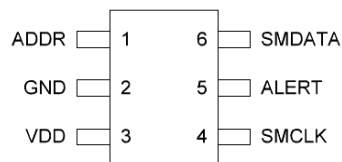


Figura 5.2 - Pinos do EMC1001

Tabela 3 - Descrição dos pinos do EMC1001

N.º	Pino	Descrição
1	ADDR	Seleção de endereço por meio da resistência (Tabela abaixo)
2	GND	Massa
3	V _{DD}	Tensão de alimentação de 3,0 V a 3,6 V
4	SMCLK	Entrada de relógio SMBus
5	ALERT	Saída lógica utilizada para gerar uma interrupção
6	SMDATA	Entrada/Saída dados SMBus

Este sensor tem a particularidade de se poder escolher o endereço que se pretende utilizar, consoante o valor da resistência ligada ao pino ADDR, o que permite utilizar apenas 1 pino, ao contrário dos habituais 3 (Microchip Technology Inc, 2016).

Tabela 4 - Endereços do EMC1001

ADDR/THERM PULL-UP RESISTOR	ENDEREÇO SMBUS EM BINÁRIO	ENDEREÇO SMBUS EM HEXADECIMAL
7,5 kΩ ±5%	1001 000b	0x09
12 kΩ ±5%	1001 001b	0x49
20 kΩ ±5%	0111 000b	0x0E
33 kΩ ±5%	0111 001b	0x4E

Este sensor possui um conjunto de registos para se poder ler o valor da temperatura bem como para configurar as saídas de alarmes. Na Tabela 5, encontram-se os registos usados no desenvolvimento do programa.

Tabela 5 - Registos do sensor de temperatura EMC1001

REGISTO EM HEXADECIMAL	DESCRIÇÃO
0x00	Valor da temperatura - High Byte
0x01	Estado
0x02	Valor da temperatura - Low Byte
0x03	Configuração
0x04	Permite configurar o número de conversões por unidade de tempo

A construção do hardware está representada pelo esquema de ligação da figura seguinte:

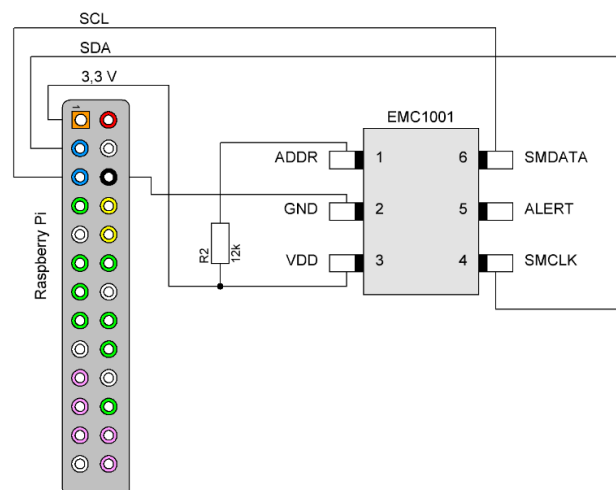


Figura 5.3 - Esquema de ligação EMC1001, Raspberry Pi

Para verificar que tudo ficou corretamente instalado, ligou-se o circuito ao Raspberry Pi, com ele desligado da alimentação (`sudo shutdown -h now`) e executou-se o seguinte comando:

```
i2cdetect -y 1
```

Com este comando apareceu o endereço do dispositivo ligado à rede I2C, neste caso é o endereço "0x49" que foi escolhido através da resistência de 12kΩ, como é apresentado na Figura 5.4.

```
pi@rasp1: ~  
pi@rasp1 ~ $ i2cdetect -y 1  
0 1 2 3 4 5 6 7 8 9 a b c d e f  
00: ---  
10: ---  
20: ---  
30: ---  
40: --- 49 ---  
50: ---  
60: ---  
70: ---  
pi@rasp1 ~ $
```

Figura 5.4 - Detecção do sensor temperatura

Com o hardware concluído e a funcionar, escreveu-se o programa em Python que seguiu a estrutura apresentada na Figura 5.5 - Fluxograma do programa sensor_temp.py

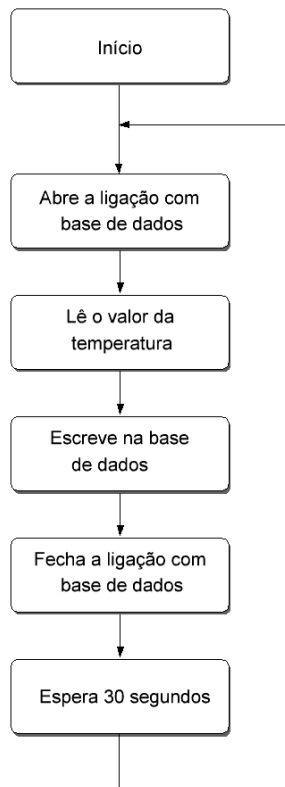


Figura 5.5 - Fluxograma do programa sensor_temp.py

Este efetua as seguintes operações:

1. abre a ligação com a base de dados “indoor” na tabela “sistema_parametros”;
2. envia o registo “0x00” para o endereço “0x49” no barramento I2C e recebe o primeiro byte com a parte inteira do valor da temperatura;
3. envia o registo “0x02” para o endereço “0x49” no barramento I2C e recebe o segundo byte com a parte decimal do valor da temperatura;
4. escreve o valor da temperatura na base de dados, no campo “temperatura”. Fecha a ligação com a base de dados;
5. espera trinta segundos.

Embora não seja necessário, pode-se compilar o programa com o seguinte comando:

```
python -m sensor_temp.py
```

Para que se execute sempre que o sistema arranque adiciona-se a seguinte linha ficheiro /etc/rc.local.

```
python /home/pi/sensor_temp.py &
```

O & no final do comando faz com que este programa fique a correr de fundo como um processo. Para verificar se o script está a correr, depois de se reiniciar o sistema, utiliza-se o comando `ps aux` o qual mostra todos os processos que estão a correr bem como o valor do id desses processos.

5.3. Sincronização de dados entre o cliente e o servidor

A sincronização de dados entre o cliente e o servidor poderá ser feita de duas formas distintas, utilizando o protocolo de comunicação CORBA, ou utilizando uma chamada a XML. A forma como os dados são sincronizados pode ser selecionada a partir da página de configuração do cliente.

A primeira é utilizada para quando o sistema está todo ligado à mesma rede local e a segundo quando o servidor se encontra fora da rede local.

A sincronização de dados entre cada cliente e o servidor é realizado na primeira forma por dois programas escritos em Python, enquanto que a sincronização de dados utilizando uma

chamada a XML é realizada por dois pequenos programas escritos em PHP, um alojado no cliente.

Em ambas as situações o programa do lado do cliente é executado a cada intervalo de tempo (10, 20, 30 ou 60 segundos) selecionável na página de configuração do respetivo cliente ou na página de gerenciamento de dispositivos no servidor.

Visto que esta forma de sincronização permite a existência de clientes fora da rede local, ficando estes ocultos do servidor, a sincronização de dados é sempre da iniciativa do cliente, já que este sabe sempre onde se encontra o servidor. O servidor recebe os dados provenientes de cada cliente e responde com os dados solicitados.

O programa de sincronização de dados do cliente, começa por ler da sua base de dados, os dados referentes ao sistema, ao ponto de acesso SimpliciTi e aos vários relógios que lhe estejam ligados, para de seguida os enviar para o servidor.

Para ser executado o programa correspondente foi escrito outro programa que é executado ao arranque do sistema operativo, ficando a correr como processo. Este lê da base de dados o intervalo de tempo de cada sincronização de dados com o servidor bem como a forma como o deve fazer.

Para que o sistema corra este programa ao arranque é necessário editar o ficheiro `/etc/rc.local` e adicionar as seguintes linha antes da linha onde aparece `exit 0`:

```
cd /home/pi  
python send_to_server.py &
```

O programa de sincronização alojado no servidor recebe os dados provenientes dos clientes, verifica a chave de segurança e se estiver correto gera uma nova chave de segurança e guarda todos os dados na base de dados do servidor. De seguida, lê os dados que estão guardados na base de dados e envia-os para o cliente.

O cliente recebe os novos dados vindos do servidor e caso verifique alteração desses com os que tem guardados na base de dados atualiza-os. Este tem a possibilidade de alterar o hostname, a sala, se está ativo e de reiniciar o cliente. A imagem seguinte apresenta o fluxograma de todo este processo.

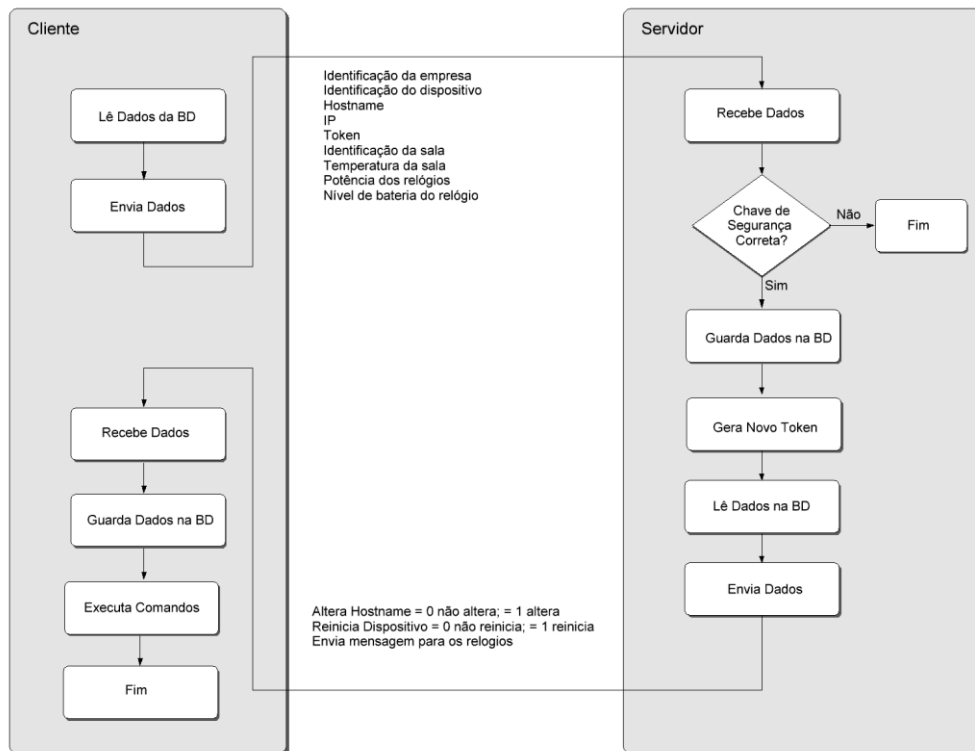


Figura 5.6 - Sincronização cliente-servidor

A sincronização de dados utilizando o protocolo CORBA é efetuado utilizando a mesma estrutura representada na Figura 5.7. Neste caso é necessário escrever um ficheiro com a extensão idl com a seguintes forma:

```

module Sincronizacao {
    interface Dados {
        string echoString(in string index);
        string echoString(in string estado);
        string echoString(in string empresa_id);
        string echoString(in string dispositivo_id);
        string echoString(in string hostname);
        string echoString(in string token_novo);
        string echoString(in string sala);
        string echoString(in string ativo);
        string echoString(in string reinicia);
        string echoString(in string atualiza_servidor);
    }
}
  
```

```
string echoString(in string msg_relogio1);  
...  
};  
};
```

Para gerar o “stub” utiliza-se o comando:

```
omniidl -bpython example_echo.idl
```

5.4. Página de configuração do cliente

A página de configuração do cliente é protegida por um sistema de login com utilizador e palavra-passe. Este sistema de autenticação foi alterado a partir do sistema de login de jpmaster77 que se encontra no sítio do “github”.

Esta página facilita a escrita e a visualização dos diferentes parâmetros que se podem alterar, utilizando um simples browser. De referir que estes parâmetros também podem ser alterados a partir do servidor em qualquer parte do mundo com acesso à internet ao contrário do acesso ao cliente que só se consegue dentro da mesma rede local.

The screenshot shows a web interface titled "Sistema de Localização Indoor de Pessoas - Configuração Cliente - rasp1". It is divided into three main sections:

- Informações:** Displays system status including Uptime (1 day, 3:11), OS (GNU/Linux), Kernel (3.10.25+), CPU (armv6l), and CPU temperature (46.5° C). It also lists Hostname (rasp1), IP (eth0) (192.168.1.129), IP (wlan0) (Desligado), Espaço em disco (3.63 GB), PHP (5.4.45-0+deb7u1), MySQL (5.5.44-0+deb7u1), and Reception (2016-05-15 22:42:00). Buttons for "Reiniciar" and "Desligar" are at the bottom.
- Dispositivo:** Configuration for the device. Fields include Estado (Ok), URL Servidor (indoor.ddns.ne), Atualiza (10 s), Empresa (ubi), Hostname (rasp1), Modo (TCP/IP), Dispositivo (0001), and Ativo (0). Buttons for "Novo", "Alterar", "Guardar", and "Testar" are at the bottom.
- Sala - Temperatura: 22°C:** Configuration for room dimensions. It lists Sala, AP, RE 1, and RE 2, each with x (m) and y (m) coordinates. Buttons for "Alterar" and "Guardar" are at the bottom.

Figura 5.7 - Página de configuração cliente

Com esta página obtêm-se a informação referente ao cliente, nomeadamente:

- estado do cliente (novo, ok, erro);
- a data da última comunicação com o servidor;
- o tempo que está a funcionar;
- o sistema Operativo que está a operar;
- a versão do núcleo do Sistema Operativo (Kernel);
- a temperatura do processador;
- a temperatura ambiente de onde está instalado;
- o Hostname;
- o IP pela interface eth0 (cabo);
- o IP pela interface wlan0 (sem fios);
- espaço livre no cartão SD.

Nesta página podem-se alterar os seguintes parâmetros:

- estado (Novo);
- nome da empresa;
- número do dispositivo;
- endereço do servidor;
- tempo de atualização com o servidor;
- protocolo que utiliza para enviar dados ao servidor, ligação via porta 80 com XML request ou utilizando o protocolo de comunicação CORBA;
- hostname;
- número da sala;
- tamanho da sala;
- posição dos repetidores.

Com os botões no final da página pode ainda desligar, reiniciar o cliente, desbloquear os parâmetros e testar a ligação com o servidor.

5.5. Lateração RSSI

O RSSI é um valor estimado da potência do sinal no canal escolhido. Este valor é baseado na corrente de ganho recebido.

O valor do *RSSI* pode ser lido a partir do registo *RSSI* status. O seguinte procedimento permite obter o valor do *RSSI* em dBm (Instruments, CC1110-CC1111, 2015)).

1. Ler o registo *RSSI*
2. Converter o valor lido em hexadecimal para decimal (*RSSI_dec*)
3. Se $RSSI_dec \geq 128$ então:

$$RSSI_dBm = \frac{RSSI_dec - 256}{2} - 74 \quad (14)$$

4. Se $RSSI_dec < 128$ então:

$$RSSI_dBm = \frac{RSSI_dec}{2} - 74 \quad (15)$$

Para a realização do algoritmo que calcula a posição de um relógio foi efetuado um conjunto de medições da *RSSI* estimada pelo ponto de acesso proveniente de um relógio eZ430 Chronos.

Ambas as medições foram efetuadas com o ponto de acesso e o relógio a 50 cm do chão. Nas Tabelas 6 e 7 apresentam-se as medições registadas, bem os cálculos da potência em dBm e em mW.

Tabela 6 - Medições *RSSI*/Distância ao ar livre

Distância	Medições			Média	Potência	
	1	2	3		dBm	mW
0	217	218	219	218,0	-93,0	5,01E-10
1	193	196	196	195,0	-104,5	3,55E-11
2	190	188	190	189,3	-107,3	1,85E-11
3	183	182	183	182,7	-110,7	8,58E-12
4	184	180	182	182,0	-111,0	7,94E-12
5	180	179	180	179,7	-112,2	6,07E-12
6	177	178	179	178,0	-113,0	5,01E-12
7	176	177	176	176,3	-113,8	4,14E-12
8	172	171	170	171,0	-116,5	2,24E-12
9	174	174	174	174,0	-115,0	3,16E-12
10	174	172	172	172,7	-115,7	2,71E-12
11	167	168	169	168,0	-118,0	1,58E-12
12	173	173	173	173,0	-115,5	2,82E-12
13	174	174	175	174,3	-114,8	3,29E-12

14	163	164	165	164,0	-120,0	1,00E-12
15	Sem Sinal					

Tabela 7 - Medições RSSI /Distância dentro do edifício

Distância	Medições			Média	Potência	
	1	2	3		dBm	mW
m						
0,0	215	220	217	217,3	-93,3	4,64E-10
1,0	201	193	195	196,3	-103,8	4,14E-11
2,0	192	189	190	190,3	-106,8	2,07E-11
3,0	187	182	186	185,0	-109,5	1,12E-11
4,0	186	183	183	184,0	-110,0	1E-11
5,0	181	178	180	179,7	-112,2	6,07E-12
6,0	185	180	186	183,7	-110,2	9,62E-12
7,0	173	166	172	170,3	-116,8	2,07E-12
8,0	183	179	178	180,0	-112,0	6,31E-12
9,0	177	174	180	177,0	-113,5	4,47E-12
10,0	170	174	174	172,7	-115,7	2,71E-12
11,0	172	172	172	172,0	-116,0	2,51E-12
12,0	175	173	173	173,7	-115,2	3,04E-12
13,0	164	163	163	163,3	-120,3	9,26E-13
14,0	Sem Sinal					

A partir dos dados registados, foi possível obter o gráfico da Figura 5.8 que contém as equações que melhor se ajustam ao valor da potência em função da distância.

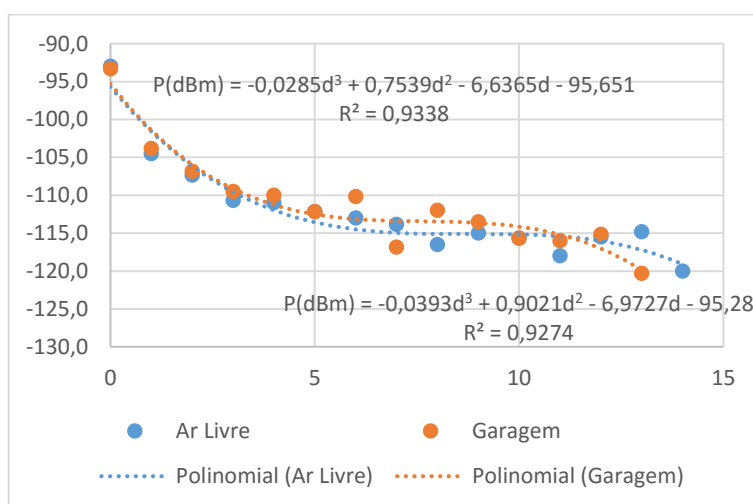


Figura 5.8 - Gráfico RSSI (dBm) em função da distância (m)

De salientar que a distância máxima atingida foi de 14 metros ao ar livre e que a propagação do sinal não é linear, pelo que para determinar a distância entre o relógio e o ponto de acesso utilizou-se uma equação de ordem 3.

O algoritmo ficou instalado no cliente e em primeiro lugar lê o valor dos três *RSSI* que estão guardados na base de dados e por fim calcula o valor da potência a partir das equações (14) e (15). De seguida, calcula a distância a cada ponto de acesso e/ou repetidor utilizando a equação:

$$P(\text{dBm}) = -0,0285d^3 + 0,7539d^2 - 6,6365d - 95,651 \quad (16)$$

Por fim determina as coordenadas (x, y) do ponto de interseção destas três distâncias, utilizando as equações (3), (4) e (5) e atualiza a base da dados. Este algoritmo é executado no mesmo intervalo de tempo em que a sincronização de dados com o servidor é efetuada. As posições do ponto de acesso e dos repetidores podem ser facilmente alteradas na página de configuração do cliente, Figura 5.7.

6. Software desenvolvido no servidor

O desenvolvimento do software do servidor utiliza a base de dados com o nome de “sistema_indoor” que contém as tabelas apresentadas na Figura 6.1. Foram utilizadas as linguagens de programação Python, PHP, HTML e javascript.

Table Name	Column Name	Data Type	Constraints/Indexes
login_users	index	INT	index
	username	VARCHAR	
	password	VARCHAR	
	userid	VARCHAR	
	userlevel	TINYINT	
	email	VARCHAR	
	timestamp	INT	
	valid	TINYINT	
	name	VARCHAR	
	genero	VARCHAR	
	telemovel	VARCHAR	
	hash	VARCHAR	
	hash_generated	INT	
	aniversario	DATE	
end_device	VARCHAR		
atualizada_data	DATETIME		
sistema_dispositivos	index	INT	index
	hostname	VARCHAR	
	empresa_id	VARCHAR	
	dispositivo_id	VARCHAR	
	token	VARCHAR	
	token_anterior	VARCHAR	
	tipo	INT	
	endereco	VARCHAR	
	ip_interface	VARCHAR	
	ativo	INT	
	reiniciar	VARCHAR	
temperatura	VARCHAR		
sala	VARCHAR		
date_time	DATETIME		
sistema_relogios	index	INT	index
	relogio_id	VARCHAR	
	utilizador	VARCHAR	
	relogio_p1	VARCHAR	
	relogio_bat	VARCHAR	
	relogio_msq	LONGTEXT	
	msq_estado	VARCHAR	
	estado	VARCHAR	
	ultima_vez_visto	DATETIME	
	sala	VARCHAR	
	sala_x	VARCHAR	
	sala_y	VARCHAR	
	x	VARCHAR	
	y	VARCHAR	
sistema_opcoes	index	INT	index
	hostname	VARCHAR	
	hostname_anterior	VARCHAR	
	corbaname	VARCHAR	
	instituicao	VARCHAR	
	sub_titulo	VARCHAR	
	tema	INT	
	fundo	INT	
	manutencao	INT	
	disp_vis	INT	
localizacao_tempo	INT		
ver_temperatura	INT		
localizacao_zoom	VARCHAR		
login_mail	mail_id	INT	PRIMARY
	Deleted	TINYINT	
	UserTo	TINYTEXT	
	UserFrom	TINYTEXT	
	Subject	MEDIUMTEXT	
	Message	LONGTEXT	
	status	TEXT	
	SentDate	TEXT	
login_active_guests	ip	VARCHAR	PRIMARY
	timestamp	INT	
	Constraints		
login_active_users	username	VARCHAR	PRIMARY
	timestamp	INT	
	Constraints		
login_banned_users	username	VARCHAR	PRIMARY
	timestamp	INT	
	Constraints		

Figura 6.1 - Estrutura da base de dados do servidor

6.1. Sistema de gerenciamento do sistema

O sistema de gerenciamento consiste numa aplicação web com interface gráfica de página de web, escrita quase na sua totalidade em html, javascript e php.

Utiliza uma biblioteca de scripts escrita em javascript de nome JQuery. Esta biblioteca é muito versátil pois em conjunto com a folha de estilo bootstrap, facilita o desenvolvimento de aplicações web dinâmicas.

Esta aplicação é constituída por:

- Página de acesso, protegida por palavra-passe para utilizadores com 3 níveis de acesso diferente:
 - Administrador - Acesso a todo o sistema.
 - Gestor - Acesso à localização de todos os utilizadores e edição dos seus dados pessoais.
 - Utilizador - Apenas pode ver a sua localização e editar os seus dados pessoais.
- Página de visualização da localização dos utilizadores.
- Página de gerenciamento de dispositivos clientes.
- Página de gerenciamento de utilizadores do sistema e de relógios.
- Página de edição de dados pessoais.

Esta é a estrutura das diretorias que constituem a aplicação web:

Nome	Tamanho	Tipo	Modificado	Permissões	Proprietári...
..					
css		Pasta de fi...	17/04/2016 18:...	drwxr-xr-x	root root
fonts		Pasta de fi...	17/04/2016 18:...	drwxr-xr-x	root root
include		Pasta de fi...	17/04/2016 18:...	drwxr-xr-x	root root
scripts		Pasta de fi...	17/04/2016 18:...	drwxr-xr-x	root root
temas		Pasta de fi...	22/04/2016 19:...	drwxr-xr-x	root root
adminprocess.php	6 491	Ficheiro PHP	17/04/2016 18:...	-rw-r--r--	root root
alerta.php	337	Ficheiro PHP	17/04/2016 18:...	-rw-r--r--	root root
alerta_manutencao.php	299	Ficheiro PHP	17/04/2016 18:...	-rw-r--r--	root root
dispositivo.xml	124	Document...	17/04/2016 18:...	-rw-r--r--	root root
dispositivos.php	17 770	Ficheiro PHP	18/04/2016 20:...	-rw-r--r--	root root
dispositivos_dados.php	1 038	Ficheiro PHP	17/04/2016 18:...	-rw-r--r--	root root
dispositivos_dados_backup.php	1 038	Ficheiro PHP	17/04/2016 18:...	-rw-r--r--	root root
dispositivos_editar.php	2 673	Ficheiro PHP	01/05/2016 02:...	-rw-r--r--	root root
dispositivos_sync.php	5 505	Ficheiro PHP	17/04/2016 18:...	-rw-r--r--	root root
dispositivos_sync_backup.php	6 025	Ficheiro PHP	17/04/2016 18:...	-rw-r--r--	root root
forgotpass.php	2 723	Ficheiro PHP	17/04/2016 18:...	-rw-r--r--	root root
funcoes.php	2 663	Ficheiro PHP	17/04/2016 18:...	-rw-r--r--	root root
index.php	12 982	Ficheiro PHP	23/04/2016 16:...	-rw-r--r--	root root
index_banner.php	1 164	Ficheiro PHP	22/04/2016 21:...	-rw-r--r--	root root
index_menu.php	7 639	Ficheiro PHP	22/04/2016 21:...	-rw-r--r--	root root
index_menu_top.php	3 300	Ficheiro PHP	22/04/2016 20:...	-rw-r--r--	root root
localizacao.php	13 420	Ficheiro PHP	22/04/2016 19:...	-rw-r--r--	root root
localizacao_ajax.php	858	Ficheiro PHP	17/04/2016 18:...	-rw-r--r--	root root
localizacao_backup.php	13 014	Ficheiro PHP	17/04/2016 18:...	-rw-r--r--	root root
login.php	1 589	Ficheiro PHP	17/04/2016 18:...	-rw-r--r--	root root
mail.php	6 413	Ficheiro PHP	17/04/2016 18:...	-rw-r--r--	root root
mensagens.php	3 270	Ficheiro PHP	22/04/2016 18:...	-rw-r--r--	root root
process.php	9 869	Ficheiro PHP	17/04/2016 18:...	-rw-r--r--	root root
relógio_estado.php	1 746	Ficheiro PHP	22/04/2016 16:...	-rw-r--r--	root root
relógios.php	9 403	Ficheiro PHP	01/05/2016 21:...	-rw-r--r--	root root
robots.txt	26	Document...	17/04/2016 18:...	-rw-r--r--	root root
sala.php	1 495	Ficheiro PHP	17/04/2016 18:...	-rw-r--r--	root root
sistema.php	24 745	Ficheiro PHP	26/04/2016 16:...	-rw-r--r--	root root
time_left.php	528	Ficheiro PHP	17/04/2016 19:...	-rw-r--r--	root root
useredit_foto_upload.php	16 340	Ficheiro PHP	17/04/2016 18:...	-rw-r--r--	root root
userinfo.php	2 700	Ficheiro PHP	17/04/2016 18:...	-rw-r--r--	root root
utilizador_add.php	5 315	Ficheiro PHP	22/04/2016 20:...	-rw-r--r--	root root
utilizador_editar.php	5 974	Ficheiro PHP	22/04/2016 15:...	-rw-r--r--	root root
utilizadores.php	15 060	Ficheiro PHP	22/04/2016 20:...	-rw-r--r--	root root
valid.php	2 559	Ficheiro PHP	17/04/2016 18:...	-rw-r--r--	root root
visitas.php	1 427	Ficheiro PHP	17/04/2016 18:...	-rw-r--r--	root root

Figura 6.2 - Estrutura das diretorias

Os administradores de sítios que estão on-line devem ter em mente que haverá muitos robôs a visitar o seu sítio. Robôs são programas que efetuam pesquisas nos sítios para os mais diversos fins. Uma forma de evitar a pesquisa destes programas é a utilização da técnica de nome “Robots Exclusion Standard” que consiste num ficheiro de nome robots.txt colocado na raiz do sítio e que contém as diretivas que um robô pode utilizar no sítio (Gourley & Totty, 2002).

O ficheiro robots.txt utilizado é constituído por duas linhas:

```
User-agent: *  
Disallow: /
```

A primeira linha permite que qualquer robô possa aceder ao sítio e a segunda linha proíbe a leitura dos ficheiros incluídos no sítio.

6.1.1. Página principal

O principal ficheiro da página de gerenciamento é o index.php, este é o primeiro ficheiro que o apache lê.

A estrutura da página é feita de forma que este ficheiro seja sempre carregado sendo que os diferentes elementos que a constituem sejam carregados conforme o que se pretende apresentar. Neste sentido a página está organizada graficamente por quatro zonas diferentes, Figura 6.3, que se alteram consoante o nível de utilizador que efetua o login e conforme o atalho que se pretende apresentar.

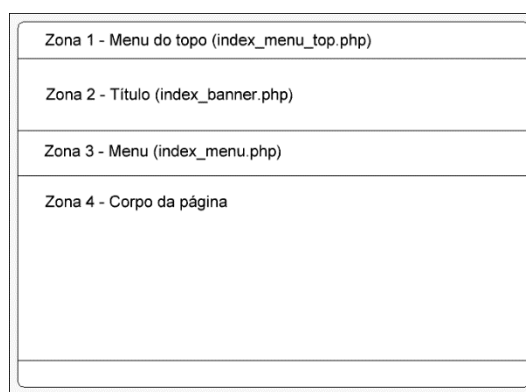


Figura 6.3 - Estrutura da página principal

Para o `index.php` poder saber qual é a página que tem a apresentar na zona 4, corpo da página, é necessário incluir essa variável, `page`, no atalho como se mostra no exemplo:

```
http://indoor.ddns.net/index.php?user=admin&page=sistema.php
```

Neste exemplo o ficheiro `index.php` fica a saber que terá de incluir na zona 4 a página `sistema.php`. Se no atalho não aparecer informação acerca da página a abrir, ele irá abrir a página que está definida por defeito que é a de localização dos utilizadores, caso o acesso já tenha sido autenticado, caso contrário irá abrir a página de acesso.

Por motivos de segurança o ficheiro `index.php` apenas poderá incluir, na zona 4, os seguintes ficheiros:

- `login.php`;
- `utilizadores.php`;
- `dispositivos.php`;
- `localizacao.php`;
- `registos.php`;
- `utilizador_editar.php`;
- `sistema.php`;
- `utilizador_add.php`;
- `alerta_manutencao.php`;
- `useredit_foto_upload.php`;
- `relogios.php`;
- `mensagens.php`.

Qualquer outro nome que se inclui no atalho apresentará uma mensagem de alerta, Figura 6.4, como mostra no exemplo seguinte:

```
http://indoor.ddns.net/index.php?user=admin&page=hacker.php
```

Uma outra medida de segurança implementada foi a adição do nome de utilizador no atalho. Sempre que o nome de utilizador incluído no atalho for diferente do guardado na sessão de acesso, este apresentará a mesma mensagem de alerta.

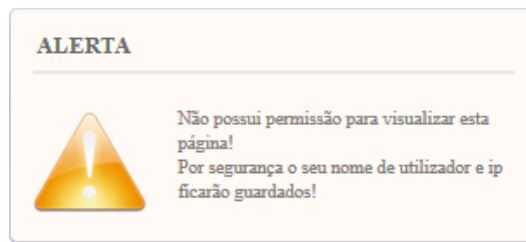


Figura 6.4 - Alerta de segurança

Se por estes dois motivos esta mensagem aparecer, ficará guardada na base de dados a informação do utilizador que efetuou o acesso, o nome do utilizador que se colocou no atalho, o seu ip, o seu ISP, browser utilizado, o nome da página, a localidade, o país, a data e a hora.

O ficheiro index.php verifica sempre o ip do computador que o está a aceder e se for um ip não permitido é apresentada uma página de erro, caso contrário passa à seguinte etapa que é determinar o tamanho do browser e ajustar as dimensões da página ao tamanho do browser.

Quando o browser acede pela primeira vez não envia comandos nem informações de configurações, passando à verificação de autenticação de acesso. Se a autenticação não estiver efetuada apresenta a página de acesso na zona 4, corpo da página, se a autenticação estiver efetuada apresenta a página de localização de pessoas.

Se a autenticação tiver sido efetuada pelo administrador aparecerá na zona 1 o primeiro menu, Figura 6.5. Neste surge o nome o utilizador do lado esquerdo e um pequeno menu no lado direito que lhe permite voltar à página principal, enviar mensagens ou terminar a sessão.



Figura 6.5 - Menu de topo para administrador

Na zona 3 surge o menu principal, este menu apresenta todas as opções do sistema.



Figura 6.6 - Menu de administrador

Se a autenticação tiver sido efetuada por um utilizador aparecerá na zona 1 o primeiro menu, Figura 6.7. Neste surge o nome do utilizador do lado esquerdo e no menu do lado direito já não surge a opção de enviar mensagem.



Figura 6.7 - Menu de topo para utilizadores

Na zona 3 surge o menu principal que apenas apresenta a opção de visualizar o próprio utilizador e a opção de editar os dados pessoais, Figura 6.8.



Figura 6.8 - Menu de utilizador

Na zona 2, surge a informação do nome da página e o logotipo que é igual para todos os níveis de utilizadores. Estes podem ser alterados na página de configuração do servidor. A Figura 6.9 apresenta a estrutura deste ficheiro.

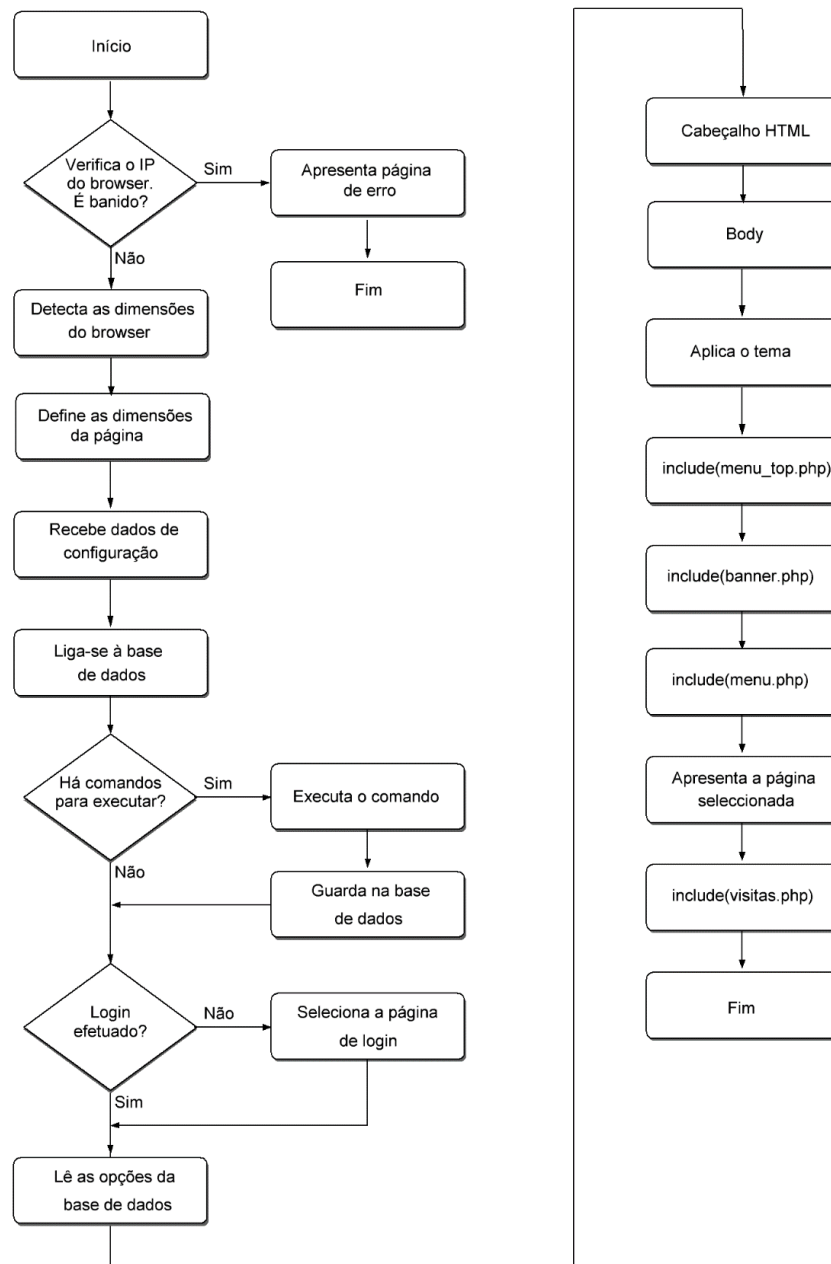


Figura 6.9 - Página principal

No fluxograma está indicada a verificação da existência de comandos a executar. Estes comandos são provenientes das outras páginas e podem ser:

- alterar o tema;
- alterar a imagem de fundo;

- alterar o título da página;
- alterar o sub-título da página;
- atualizar o estado de manutenção;
- alterar o tempo de atualização na pagina de visualização de utilizadores;
- alterar o zoom na página de visualização de utilizadores;
- alterar o estado de visualização da temperatura.

A última tarefa que o index.php efetua é guardar na base de dados a informação de quem visita a página. A implementação desta funcionalidade encontra-se descrita no Capítulo 6.1.3 e deveu-se a motivos de segurança, pois o servidor encontra-se ligado à internet.

6.1.2. Sistema de acesso

O sistema de autenticação foi alterado a partir do sistema de login de jpmaster77 que se encontra no sítio do “github”, dando-lhe nova apresentação e novas funcionalidades.



AUTENTICAÇÃO

▲ UTILIZADOR 🔒 CÓDIGO DE ACESSO

Utilizador Código de Acesso

Mostrar Palavra-passe

Entrar

Figura 6.10 - Sistema autenticação

Neste é possível mostrar/ocultar o código de acesso e se o utilizador ou o código estiverem errados ambos os campos ficam vermelhos.

6.1.3. Controlo de acessos

Estando o sistema acessível a partir da internet é necessário, por razões de segurança, ter a informação de quem e como tenta aceder aos conteúdos da página. Para isso foi escrito um script em PHP que com o auxílio do API que se encontra em <http://ip-api.com/php> realiza essa tarefa.

Podendo-se assim obter as seguintes informações de quem efetua o acesso:

- o IP;
- o ISP;
- o browser utilizado;
- a data e a hora de acesso;
- o nome da cidade e do país;
- a página visitada.

Este script guarda toda a informação de acessos na base de dados.

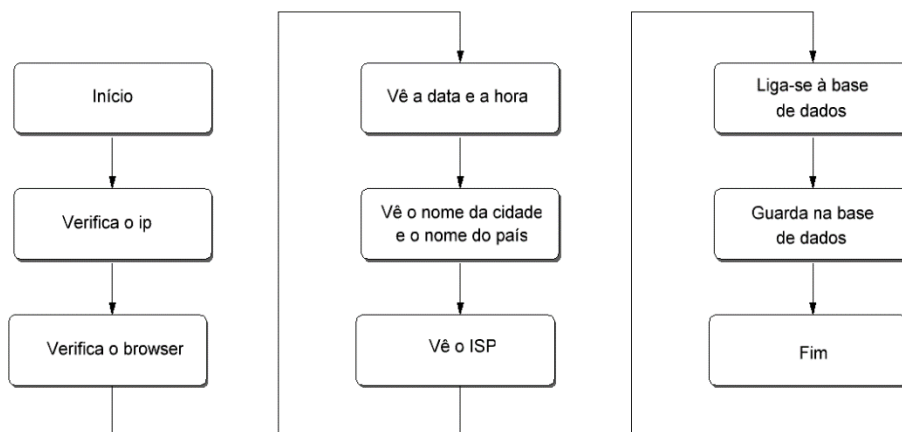


Figura 6.11 - Sistema de controle de acessos

6.1.4. Política de cookies

Quase todos os sítios da internet utilizam cookies que são pequenos ficheiros onde o browser guarda informações acerca da pessoa que visita a página.

Em maio de 2011 a União Europeia adotou uma diretiva Norte Americana que tem por nome “The Cookie Law” que obriga os sítios que utilizem cookies, a informar e pedir o consentimento do seu uso.

Esta lei foi criada para proteger a privacidade online, tornando os consumidores conscientes de como a sua informação é recolhida e utilizada online, dando-lhes a opção de permitir ou não o seu uso (Optanon, 2016).

Embora o sistema de acesso utilize apenas dois cookies, o “cookname” e o “cookid”, para identificação do utilizador, foi adicionada a mensagem de aviso da sua utilização, Figura 6.12.

```
setcookie("cookname", $this->username, time()+COOKIE_EXPIRE, COOKIE_PATH);
```

```
setcookie("cookid", $this->userid, time()+COOKIE_EXPIRE, COOKIE_PATH);
```

Esta mensagem só aparece a primeira vez que se visita o sítio. Foi adaptada da página “Begin Cookie Consent plugin” de Silktide e pode ser descarregada no seu sítio (Silktide, 2016).

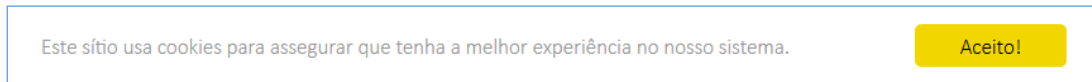


Figura 6.12 - Políticas de cookies

6.1.5. Visualização da localização de utilizadores

A página de visualização da localização de utilizadores é acessível para todos os utilizadores, mas apenas o administrador e o gestor poderão ver a localização de todos os utilizadores. Todos os outros utilizadores apenas poderão ver a sua própria localização.

A visualização da localização dos utilizadores foi construída a partir de um script de nome “Mapael”. É um plugin jQuery escrito por Vincent Brouté que permite visualizar dinamicamente mapas vetoriais. Este pode ser descarregado no seu sítio oficial, (Brouté, 2016).

No sistema desenvolvido foi utilizada uma planta em formato CAD do edifício das engenharias da Universidade da Beira Interior, facultada pelo Professor Doutor Bruno Ribeiro. Foi convertida para o formato SVG e posteriormente editada com o programa Inkscape (editor de imagens vetorial de código aberto). Depois de editada foi convertida para mapael javascript através de uma aplicação no sítio oficial do plugin.

A posição dos utilizadores é atualizada conforme o tempo de atualização selecionado na página de configuração do servidor, que pode ser de 5, 10, 15, 20 ou 30 segundos. Assim, ao fim desse tempo, a página recarrega os dados dos utilizadores utilizando dois scripts. Um deles embutido na página de visualização, escrito em AJAX, solicita a informação a outro script, escrito em PHP, ficando à espera da resposta dos dados. O script escrito em PHP é responsável por:

- Atualizar o estado dos utilizadores:
 - estado 0, inativo há mais de 5 minutos (não aparece);
 - estado 1, inativo há mais de 2 minutos (apresenta-se de cor cinza);
 - estado 2, ativo há menos de 2 minutos (apresenta-se de cor laranja);
 - estado 3, emergência (apresenta-se de cor vermelha).
- Ler os dados da base de dados:

- nome de utilizador;
- posição segundo o eixo horizontal;
- posição segundo o eixo vertical;
- estado do utilizador.
- Codificar os dados no formato JSON.
- Enviar os dados dos utilizadores para a página de visualização.

O script em AJAX é responsável pela solicitação da atualização dos dados referentes aos utilizadores e à atualização dos mesmos na planta. A Figura 6.13 apresenta a forma como este processo acontece.

A utilização do script em AJAX foi muito importante, pois este javascript é executado no browser depois da página de visualização ser carregada. Este solicita o envio da atualização dos dados referentes aos utilizadores guardados no servidor. Esta técnica permite a atualização de pequenas partes da página, neste caso a posição dos utilizadores, sem haver a necessidade de recarregar toda a página (Woychowsky, 2007), resultando numa página mais plausível tendo em conta o resultado pretendido.

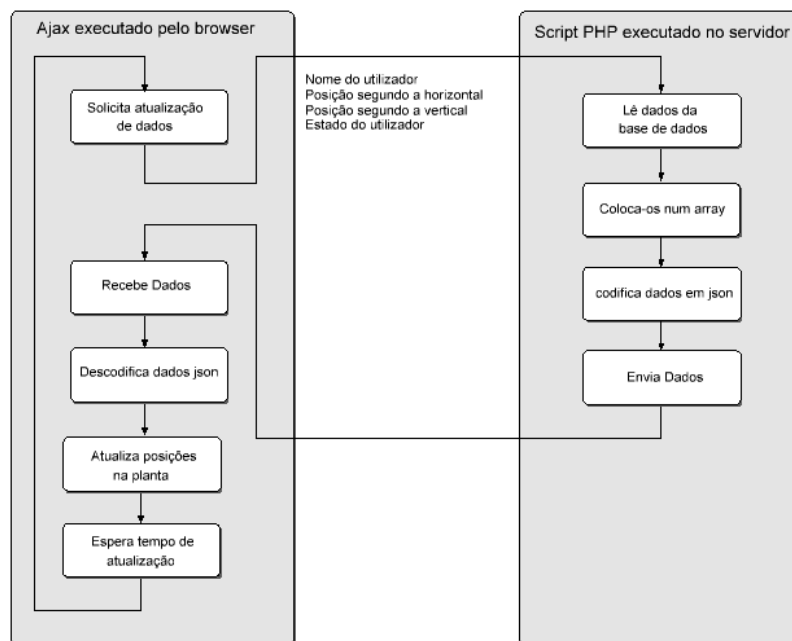


Figura 6.13 - Estrutura de comunicação AJAX

A planta possui 10 níveis de ampliação e esta inicia-se com o nível que é selecionado na página de configuração do servidor. Possui a capacidade para ampliar ou reduzir a sua escala da planta com a ajuda dos botões colocados na parte inferior da legenda.

A legenda informa que a cor corresponde ao estado do utilizador e tem a capacidade de identificar todos os que estão no mesmo estado ao clicar no círculo correspondente. É possível arrastar a planta em todas as direções quando esta se encontra ampliada.

A planta apresenta, ainda, a identificação das salas e quando se passa com o rato a cor de fundo altera-se, surgindo a informação da sala bem como a temperatura da mesma, se esta opção estiver selecionada na página de configuração do servidor.

De forma análoga sempre que se passa com o rato sobre um utilizador surge a informação do utilizador bem como a sua fotografia. Ao clicar num dos utilizadores é possível enviar mensagens para o relógio do utilizador selecionado.

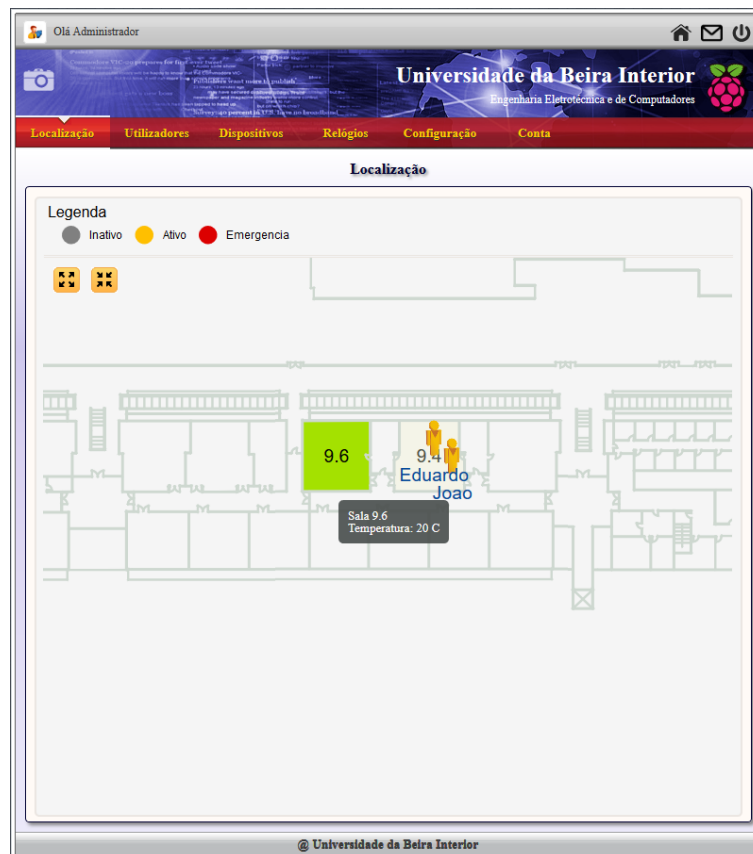


Figura 6.14 - Página de visualização de utilizadores

6.1.6. Página de mensagens

Esta página está acessível para o administrador e para o gestor do sistema e permite o envio de mensagens diretamente para o relógio dos utilizadores. Pode ser acedida de três formas distintas:

- menu superior, tendo depois a opção de seleccionar o utilizador a que se pretende enviar a mensagem;
- página de visualização de utilizadores, clicando no respetivo utilizador;
- página de gerenciamento de utilizadores, clicando no ícone de mensagem.

A página que surge ao aceder à página de mensagens é a que se apresenta na Figura 6.15.

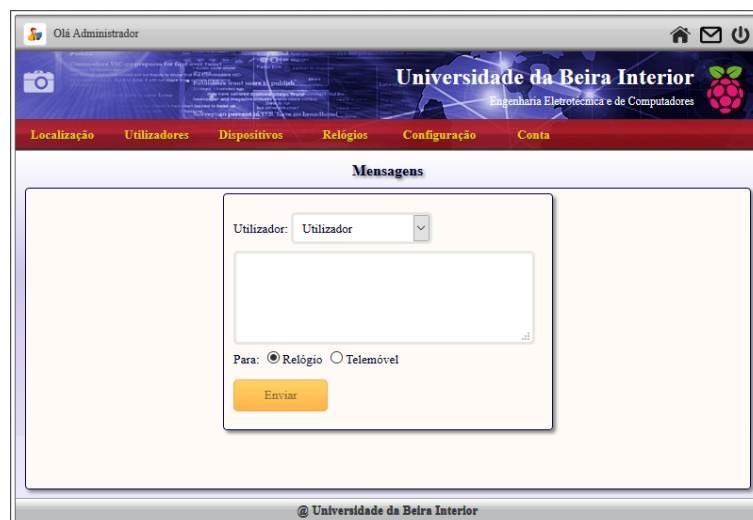


Figura 6.15 - Página para enviar mensagens

6.1.7. Gerenciamento de utilizadores

A página de gerenciamento de utilizadores está apenas acessível ao administrador. Nesta é possível:

- visualizar todos os utilizadores em listas de 5, 10 ou 15 utilizadores;
- adicionar novos utilizadores;
- apagar utilizadores;
- editar a informação dos utilizadores;
- enviar mensagens para o relógio;
- procurá-los na página de visualização de localização de utilizadores.

Na linha correspondente a cada utilizador é possível ver: o nome de utilizador; o nome completo; o nível de acesso; o número de contato; o email; data de aniversário; o número do relógio atribuído; o ícone de procura; o ícone de envio de mensagem; o ícone de edição e o ícone para apagar o utilizador.

A edição de utilizadores surge quando se clica no ícone de edição de utilizadores. Ao clicar-se surge abaixo da linha correspondente ao utilizador todos os campos possíveis de editar. Ao clicar-se no ícone de adicionar utilizador surge abaixo da barra de navegação os campos para adicionar o novo utilizador.

Foi ainda adicionada uma barra de navegação, sendo possível seleccionar o número de utilizadores visíveis por lista bem como atalhos para avançar ou retroceder nessas listas.

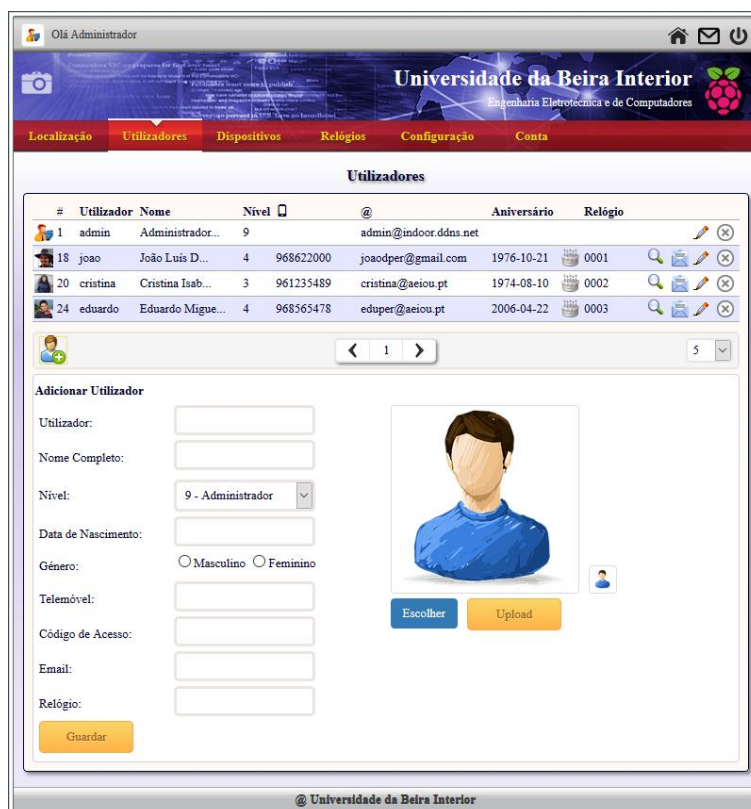


Figura 6.16 - Página de gerenciamento de utilizadores

6.1.8. Gerenciamento de dispositivos

A página de gerenciamento de dispositivos é apenas acessível ao administrador. Nesta o administrador pode gerenciar todos os dispositivos, nomeadamente:

- procurar novos dispositivos que estejam dentro da mesma rede que o servidor. Este script encontra-se descrito no Capítulo 6.3;
- adicionar manualmente um dispositivo;
- editar dispositivos remotamente:
 - nome da empresa proprietária do dispositivo,
 - número de identificação do dispositivo;
 - hostname;
 - estado;
 - reiniciar dispositivo;
 - tempo de sincronização com o servidor;
- apagar dispositivos existentes;
- reiniciar todos dispositivos ao mesmo tempo;

Pode ainda visualizar todos os dispositivos existente bem como a informação:

- tipo de dispositivo (servidor ou cliente);
- nome da empresa proprietária do dispositivo;
- identificação da divisão onde é colocado;
- número de identificação do dispositivo;
- hostname;
- estado;
- tipo de interface a que se liga à internet, a partir de ícones (wlan0 ou eth0);
- data e hora da última sincronização;
- temperatura da divisão onde está colocado.

Foi adicionado o estado do cliente a partir de ícones circulares de cores. Cor verde quando o cliente sincroniza corretamente os dados há menos de 1 minuto, cor amarela quando a última sincronização ocorreu há mais de minuto e meio e cor cinza quando o cliente não sincroniza há mais de 5 minutos.

Possui uma barra de navegação que permite selecionar o número de dispositivos visíveis por lista, podendo apresentar um máximo 15 dispositivos de cada vez. Possui atalhos para avançar e/ou retroceder no número de listas.

The screenshot displays the 'Dispositivos' (Devices) management page. At the top, there is a navigation menu with options: Localização, Utilizadores, Dispositivos (selected), Relógios, Configuração, and Conta. The main content area is titled 'Dispositivos' and contains a table with the following data:

#	Tipo	Empresa	Dispositivo	Hostname	Ativo	Endereço	Sincronização	Temp
102	Servidor	ubi	0000	rasp0	Não	192.168.1.67		
113	Ciente	jplda	0001	rasp1	Sim	192.168.1.100	2016-04-15 09:14:32	22 °C
116	Ciente	ubi	0002	rasp2	Sim	192.168.1.131	2016-05-01 19:08:12	18 °C
117	Ciente	ubi	0001	rasp1	Não	192.168.1.129	2016-05-01 19:09:33	27 °C

Below the table is an 'Editar Dispositivo' (Edit Device) form with the following fields:

- Empresa: ubi
- Sala: 9.4
- Atualiza: 10 s
- Dispositivo: 0001
- Desativo: Desativo
- Hostname: rasp1
- Reiniciar: Não

A 'Guardar' (Save) button is located below the form. At the bottom of the page, there is a 'Configurações Gerais' (General Settings) section with a 'Reiniciar todos' (Restart all) button.

Figura 6.17 - Página de gerenciamento de clientes

6.1.9. Informação de relógios

Esta página está acessível unicamente para o administrador do sistema e nela é possível a visualização da informação relativa aos relógios que se encontram no sistema, nomeadamente:

- o número de identificação do relógio;
- o utilizador ao qual está associado o relógio;
- os valores dos três RSSI recebidos;
- os valores da posição estimada do relógio;
- o valor da tensão da bateria;
- a data e hora da última interação com o sistema.

Relógio ID	Utilizador	RSSI 1	RSSI 2	RSSI 3	X	Y	Bateria	Última vez visto	Estado
0001	joao	176	167	175	19.41	1	3.03 V	2016-05-31 15:41:05	1
0002	cristina	183	168	170	6.08	-0.41	2.97 V	2016-05-31 15:41:05	1
0003	eduardo				1.05	4.99	3.83 V	2016-05-31 15:41:05	0

Figura 6.18 - Gerenciamento de relógio

Para se adicionar um novo relógio, basta inserir o seu número de identificador quando da adição de um novo utilizador na página de gerenciamento de utilizadores. Da mesma forma, é possível remover um relógio editando o número de identificação do relógio editando os dados dos utilizadores.

6.1.10. Página de configuração do servidor

Esta página está acessível unicamente para o administrador do sistema e nela pode configurar todo o sistema sem ter a necessidade de recorrer a qualquer outra ferramenta. Está organizada por três secções distintas:

- informações;
- configurações;
- aparência.

Na secção “informações” é possível visualizar:

- tempo que o servidor está ligado;
- nome do Sistema Operativo;
- versão do kernel;
- a versão do CPU;

- a temperatura do CPU;
- o IP da interface eth0 e wlan0;
- espaço disponível no cartão SD;
- atalho para a página do Phpmyadmin e Webmin;
- pode desligar e/ou reiniciar o servidor a partir de dois botões.

Na secção “configurações” é possível alterar:

- o hostname;
- o SSID e código de acesso da rede wireless.

Na secção “aparência” é possível alterar:

- o título, o subtítulo e o logotipo da empresa;
- o tema e imagem de fundo;
- o estado de manutenção da página;
- o tempo de atualização das posições;
- o zoom inicial da planta;
- seleccionar se se pretende ou não apresentar as temperaturas.

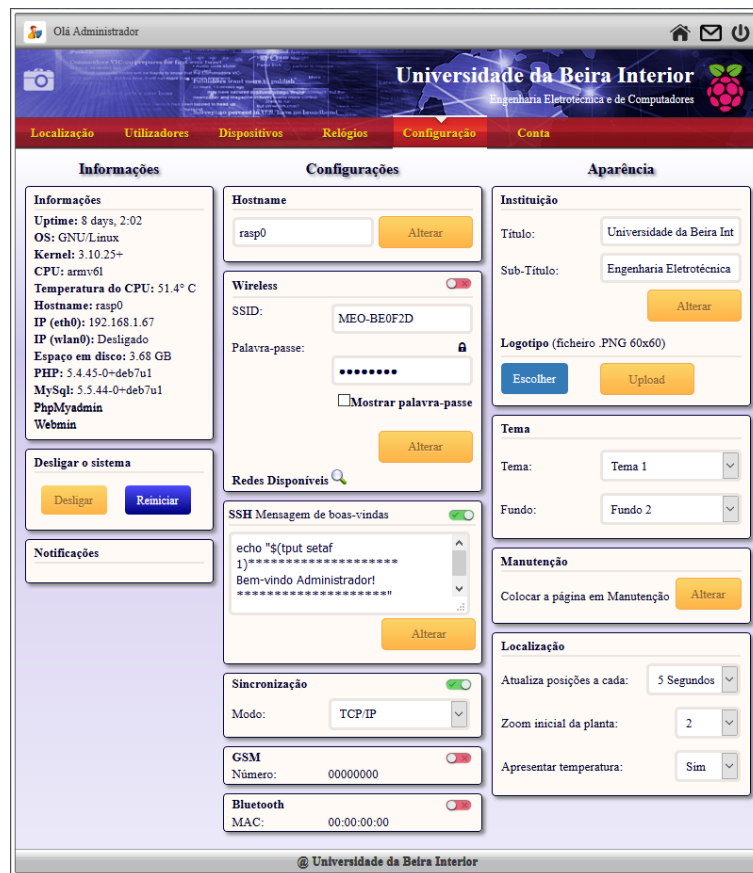


Figura 6.19 - Página de configuração do servidor

6.1.11. Página de edição de dados pessoais

A página de edição de dados pessoais está acessível para todos os utilizadores, podendo o utilizador alterar os seguintes dados:

- nome;
- data de nascimento;
- género;
- número de contato;
- palavra de acesso;
- email;
- fotografia.

Para facilitar a edição da data de nascimento, no formato correto, foi colocado um calendário que surge quando se clica no respetivo campo de edição.

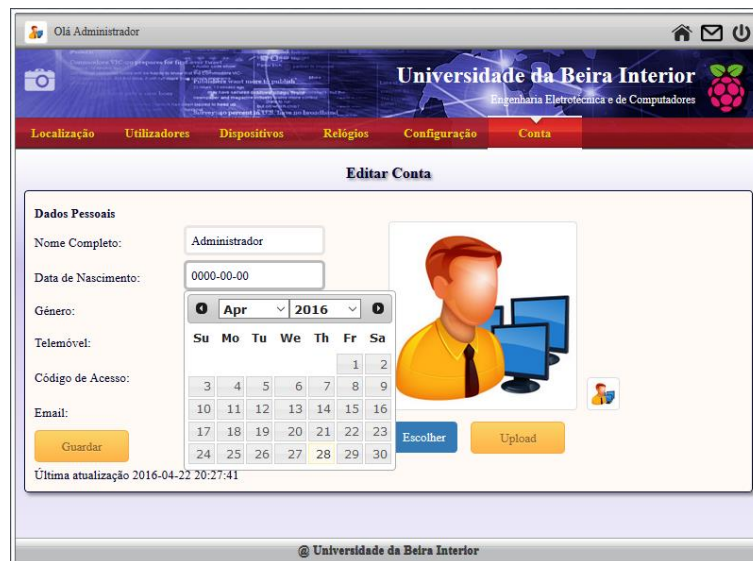


Figura 6.20 - Página de edição de dados pessoais

O script que controla o envio da fotografia, recebe a fotografia, redimensiona-a para 250×250 pixels e guarda-a em /temas/fotos com o nome <utilizador>250.jpg de seguida faz uma cópia, redimensiona-a para 25×25 pixels e guarda-a na mesma diretoria com o nome de <utilizador>25.jpg. Para o envio da fotografia é necessário que a diretoria /temas/fotos tenha a permissão de ler e escrever, assim utiliza-se o comando `chmod` para atribuir essas permissões:

```
cd /home/indoor/www
sudo chmod 755 /temas/fotos
```

6.2. Alteração do hostname

Para facilitar o gerenciamento do hostname dos clientes foi adicionada a opção de se alterar o hostname pela página de gerenciamento de dispositivos. O programa escrito em python necessita de editar dois ficheiros, como já foi referido no Capítulo 4.3.6.

Depois deste estar a funcionar foi encontrada uma outra forma mais eficiente de efetuar a mesma tarefa, tendo sido desenvolvido um programa em PHP.

6.3. Detecção de novos dispositivos

Uma funcionalidade desenvolvida é a possibilidade do servidor encontrar e adicionar clientes automaticamente ao sistema. Funcionalidade apenas disponível quando os clientes estão dentro da mesma rede que o servidor. Este script foi escrito em PHP e pode ser executado a partir da página de gerenciamento de dispositivos.

O programa começa por se ligar à base de dados e ler o nome dos dispositivos que tem guardado para depois procurar novos dispositivos na rede. Se encontrar novos dispositivos que não se encontram na base de dados adiciona-os. Se estão na base de dados mas não se encontram na rede, coloca-os no estado de inativos.

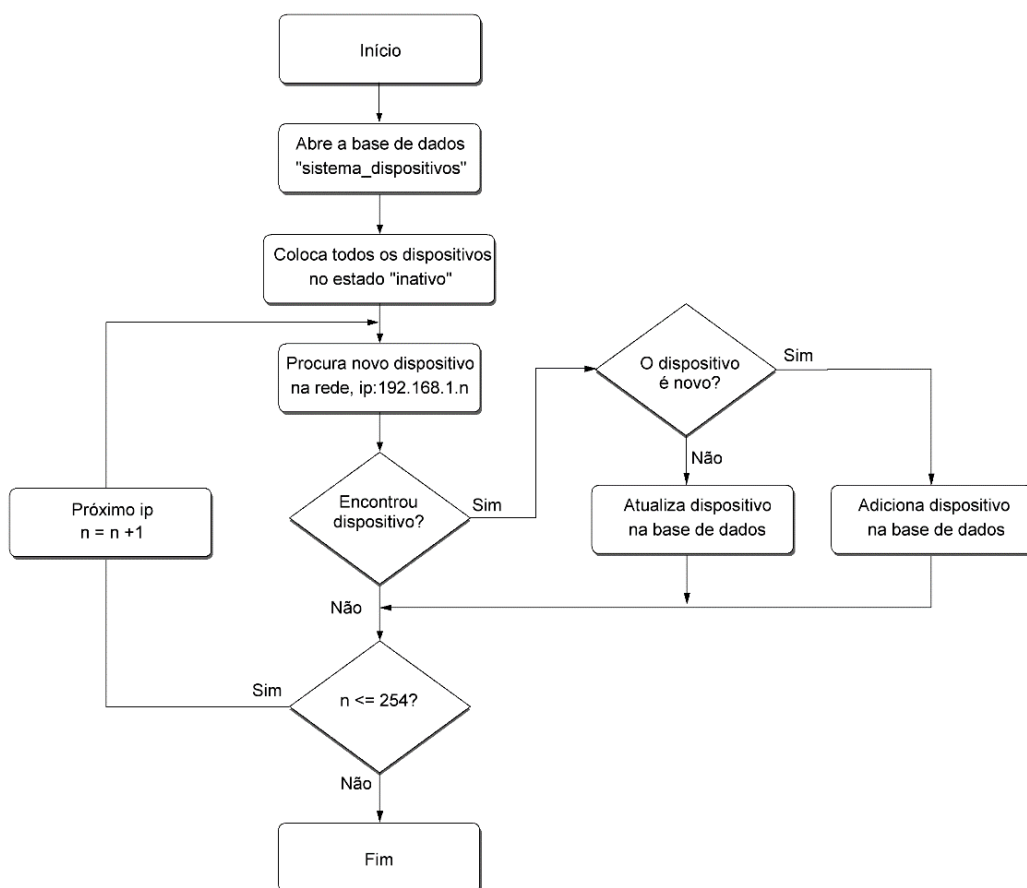


Figura 6.21 - Estrutura find_hosts.php

6.4. Detecção de redes sem fios

Para uma maior facilidade em ligar o servidor a uma rede sem fios, foi desenvolvido uma interface gráfica, Figura 6.22, onde se pode:

- ligar e desligar a placa de rede sem fios;
- alterar o SSID e a palavra-passe da rede sem fios que se pretende ligar;
- procurar redes sem fios disponíveis;



Figura 6.22 - Gerenciamento de redes sem fios

Para se ligar ou desligar a placa de redes sem fios utiliza-se o botão no canto superior direito. Este executa o comando `sudo ifup wlan0` para o caso de se pretender ligar e o comando `sudo ifdown wlan0` para o caso de se pretender desligar. Quando a placa de rede é ligada o botão passa da cor vermelha para a cor verde.

Para procurar as redes disponíveis utiliza-se o ícone de lupa. Este ao ser clicado executa o comando `sudo iwlist wlan0 scan` o qual apresenta o SSID de todas as redes sem fios disponíveis. Ao clicar-se num dos SSID que surge, basta clicar nele e introduzir a palavra-passe e clicar em alterar.

7. Conclusão

A trilateração de potências é uma abordagem que contém muitas condicionantes, como interferência, múltiplas reflexões, diferenças de temperatura e até condições atmosféricas. Estas introduzem muitos erros no sistema e só com algoritmos rápidos e que contemplassem todos estes fenómenos que influenciam a transferência de energia em ondas eletromagnéticas poderia obter-se uma precisão apreciável. Atualmente ainda não existem sistemas que consigam ultrapassar totalmente estas condicionantes. Não se pode esquecer que dentro de um edifício o erro de ± 1 metro corresponde a uma área considerável.

A melhor solução passaria pela utilização de dois métodos diferentes de forma a poder-se correlacionar informações. Por exemplo a utilização da localização inercial associada a pontos de acessos que poderiam fazer um reiniciar da posição inicial.

Constatei que este sistema poderia ser mais apelativo se fossem utilizados outro tipo de relógios mais modernos e com outros recursos, como: melhores baterias, melhores displays. Uma outra solução seria o desenvolvimento de um hardware específico para esta tarefa.

Trabalhar com o Raspberry pode por vezes ser ingrato, pois como todo o sistema se encontra instalado num cartão SD, quer por falta de energia ou por um encerramento mal efetuado pode corromper os ficheiros, resultando na perda de horas de trabalho. Um sistema deste tipo, não deverá ser instalado sem ter um sistema de alimentação com um acumulador de energia.

Para que esta situação seja evitada poder-se-á desenvolver uma pequena UPS para o Raspberry Pi.

Um outro módulo que se poderá desenvolver é um módulo GSM, para que o sistema possa enviar mensagens de texto para os utilizadores diretamente da página de gerenciamento do sistema.

Foram implementadas inúmeras medidas de segurança no desenvolvimento deste trabalho. Estas deveram-se ao facto do servidor se encontrar ligado à internet e com duas portas abertas (a 80 para servidor de páginas e a 22 para o servidor de ssh). No decorrer do desenvolvimento deste trabalho prático de dissertação, dezenas de vezes foram as tentativas de acesso.

Ao sistema desenvolvido facilmente se poderá adicionar mais serviços, como por exemplo mais plantas diferentes.

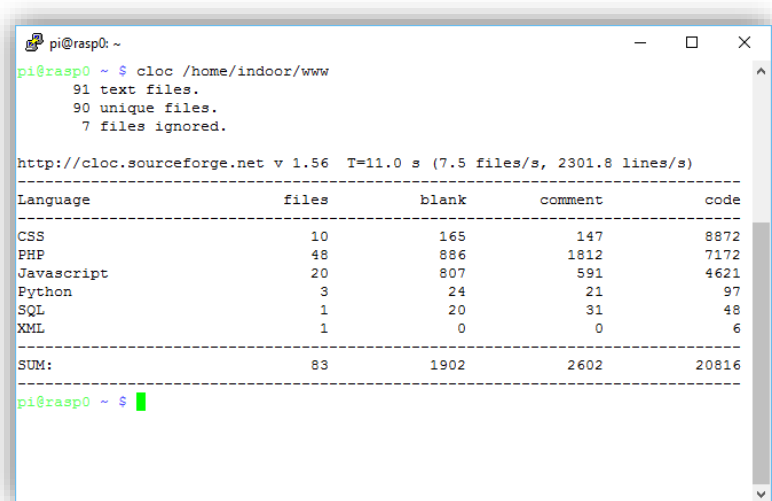
De salientar que o sistema aqui desenvolvido será facilmente instalado noutra tipo de placas menos dispendiosas e com maior poder computacional.

Da componente prática deste trabalho resultou um total de mais de 20 mil linhas de código, contabilizando apenas o código desenvolvido no servidor, Figura 7.1, o que demonstra que por detrás da aparência gráfica fica um extenso trabalho de programação.

Pensando num trabalho futuro de desenvolvimento, em vez de utilizar os pontos de acesso ligados ao porto USB, desenvolveria uma ligação SimpliCI ligando o CC2500 diretamente ao Raspberry Pi comunicando via SPI. Com esta abordagem poupar-se-ia um porto USB.

Com a realização desta dissertação aprofundei os meus conhecimentos acerca de técnicas de localização no interior de edifícios, em particular a lateração de distâncias obtidas a partir da potência recebida, micro-computador Raspberry Pi e do sistema operativo Linux.

Da componente prática resultou um trabalho positivo provando que é possível utilizar os relógios ez430 Chronos da Texas Instruments como sistema de localização de pessoas dentro de edifícios.



```
pi@rasp0: ~  
pi@rasp0 ~ $ cloc /home/indoor/www  
91 text files.  
90 unique files.  
7 files ignored.  
  
http://cloc.sourceforge.net v 1.56 T=11.0 s (7.5 files/s, 2301.8 lines/s)  
-----  
Language          files      blank     comment     code  
-----  
CSS                10         165         147         8872  
PHP                48         886        1812         7172  
Javascript         20         807         591         4621  
Python             3           24          21           97  
SQL                1           20          31           48  
XML                1            0            0            6  
-----  
SUM:              83        1902        2602        20816  
-----  
pi@rasp0 ~ $
```

Figura 7.1 - Número total de linhas de código no servidor

8. Referências bibliográficas

Adelstein, T., & Lubanovic, B. (2007). *Linux System Administration*. United States of America: O'Reilly Media, Inc.

Ahson, S., & Ilyas, M. (2011). *Location-Based Services Handbook*. United States of America: CRC Press.

Brouté, V. (26 de fevereiro de 2016). *jQuery Mapael 1.1.0*. Obtido de jQuery Mapael: <http://www.vincentbroute.fr/mapael>

CC1111 USB Evaluation Module Kit 868/915 MHz. (12 de fevereiro de 2016). Obtido de Texas Instruments: <http://www.ti.com/graphics/tool/CC1111EMK868-915.jpg>

Chronos: Wireless development tool in a watch. (12 de fevereiro de 2016). Obtido de Texas Instruments: http://www.ti.com/diagrams/med_ez430-chronos_ez430-chronos_800.jpg

Cox, T. (2014). *Raspberry Pi Cookbook for Python Programmers*. Birmingham, UK: Published by Packt Publishing Ltd.

Dennis, A. K. (2013). *Raspberry Pi Home Automation with Arduino*. United Kingdom: Packt Publishing.

Fink, A., Lange, J., & Beikirch, H. (2012). *Radio-Based Indoor Localization using the eZ430-Chronos Platform*. Rostock University.

Gourley, D., & Totty, B. (2002). *HTTP: The Definitive Guide*. United States of America: O'Reilly & Associates, Inc.

Instruments, T. (14 de novembro de 2015). *CC1110-CC1111*. Obtido de Texas Instruments: <http://www.ti.com/product/CC1110-CC1111>

Instruments, T. (21 de 11 de 2015). *Chronos: Wireless development tool in a watch*.

Obtido de Texas Instruments: <http://www.ti.com/tool/ez430-chronos>

Instruments, T. (21 de 11 de 2015). *Sub-1 GHz RF System-on-Chip with Integrated USB*.

Obtido de Sub-1 GHz RF System-on-Chip with Integrated USB:

<http://www.ti.com/corp/docs/landing/cc1111>

Kirch, O., & Dawson, T. (2000). *Linux Network Administrator's Guide*. Estado Unidos da América: O'Reilly.

Kofler, M. (2005). *The Definitive Guide to MySQL 5* (Third Edition ed.). United States: Apress.

Labrador, M., Pérez, A., & Wightman, P. (2011). *Location-Based Information System Developing Real-Time Tracking Applications*. Boca Raton: CRC Press.

Lyon, G. (2008). *Nmap Network Scanning Official Nmap Project Guide to Network Discovery and Security Scanning*. United States: Insecure.Com LLC.

Membrey, P., & David, H. (2012). *Learn Raspberry Pi with Linux*. United States of America: Apress.

Microchip Technology Inc. (6 de 4 de 2016). *Microchip Technology Inc*. Obtido de EMC1001 1.5 Degree C SMBus Temperature Sensor in Miniature SOT-23 - 1001:

<http://ww1.microchip.com/downloads/en/DeviceDoc/1001.pdf>

Norris, D. (2014). *Raspberry Pi Projects for the Evil Genius*. United States: McGraw-Hill Education.

omniORB Documentation. (23 de janeiro de 2016). Obtido de omniORB:

<http://omniorb.sourceforge.net/docs.html>

Optanon. (26 de 04 de 2016). *The Cookie Law Explained*. Obtido de EU Cookie Law - Audit, Cookie Compliance, Analytics: <https://www.cookie-law.org/the-cookie-law/>

PHP, M. (24 de 4 de 2016). *PHP: Tags PHP - Manual*. Obtido de PHP: Hypertext Preprocessor: https://secure.php.net/manual/pt_BR/language.basic-syntax.php.tags.php

Redpin. (10 de maio de 2016). *Redpin - Indoor Positioning for the Rest of Us*. Obtido de Redpin: <http://redpin.org/index.html>

Silktide. (12 de fevereiro de 2016). *SilkTide*. Obtido de Cookie Consent: <http://silktide.com/cookieconsent>

Skyhook. (12 de 12 de 2015). Obtido de Actionable Context Global Location: <http://www.skyhookwireless.com>

Sverdlov, E. (5 de dezembro de 2015). *Como criar um novo usuario e conceder permissoes no mysql*. Obtido de Digital Ocean: <https://www.digitalocean.com/community/tutorials/como-criar-um-novo-usuario-e-conceder-permissoes-no-mysql-pt>

Tahaghoghi, S. M., & Williams, H. E. (2007). *Learning MySQL*. United States of America: O'Reilly Media.

Vugt, S. v. (2006). *The Definitive Guide to SUSE Linux Enterprise Server*. United States of America: Apress.

Watson, A. (2013). *The Big Book of Raspberry Pi* (First Edition ed.). CreateSpace Independent Publishing Platform.

Welling, L., & Thomson, L. (2009). *PHP and MySQL Web Development* (Fourth Edition ed.). United States of America: Pearson Education, Inc.

Werner, M. (2014). *Indoor Location-Based Services Prerequisites and Foundations*. Munich, Germany: Springer International Publishing.

Wester, I. (6 de janeiro de 2016). *Como usar o apt-get*. Obtido de Info Wester: <http://www.infowester.com/aptget1.php>

- William E. Shotts, J. (2009). *The Linux® Command Line*. A LinuxCommand.org Book.
- Woychowsky, E. (2007). *Ajax Creating Web Pages with Asynchronous*. Pearson Education, Inc.