



Departamento de Informática

Setembro 2009

Universidade do Minho

**Sistema de monitorização
e descoberta
de *hotspots Bluetooth* por georeferenciação**

Frederico Nuno Ferreira da Costa Tavares



Departamento de Informática

Setembro 2009

Universidade do Minho

**Sistema de monitorização
e descoberta
de *hotspots Bluetooth* por georeferenciação**

Tese submetida à Escola de Engenharia da Universidade do Minho para a
obtenção do grau de Mestre em Informática

Frederico Nuno Ferreira da Costa Tavares

Orientação: Prof. Doutor Paulo Martins Carvalho

Co-orientação: Mestre Paulo Carvalhal

Agradecimentos

Um agradecimento especial aos meus orientadores, Professor Doutor Paulo Martins Carvalho e Mestre Paulo Carvalhal, a quem devo a orientação deste trabalho, o constante apoio e incentivo, empenho e disponibilidade manifestados, bem como as numerosas sugestões que contribuíram para a sua execução e enriquecimento.

Agradeço ainda à minha mãe e irmã, por sempre me terem apoiado ao longo deste projecto, pelo estímulo, incentivo e ânimo que me proporcionaram.

Um agradecimento ainda à Vânia, pelo apoio e amizade que muito prezo e cuja presença ao longo deste trabalho foi muito importante.

A todos eles, dedico este trabalho.

Resumo

A tecnologia *Bluetooth* abre inúmeras possibilidades para aqueles que pretendem comunicações eficazes, sem fios, a curta distância. Utilizada em PDA's, telemóveis, computadores e outros dispositivos, pode ser associada ao *marketing* de proximidade.

O presente trabalho pretende evidenciar de que forma a tecnologia *Google Maps* pode ser utilizada para, juntamente com sistemas móveis equipados com *Bluetooth*, descobrir e monitorizar, em tempo real, o estado de um conjunto de *hotspots*. Neste caso específico, pretendeu-se dar a conhecer de que forma é possível georeferenciar e monitorizar *hotspots Bluetooth* e de que forma era possível disponibilizar tal ferramenta numa aplicação *Web based*.

Uma vez que estes *hotspots* são utilizados na difusão de conteúdos publicitários, foi ainda feita uma abordagem ao *marketing* de publicidade, bem como às diferentes tecnologias disponíveis actualmente para dar resposta à georeferenciação de objectos.

O sistema desenvolvido, aplicado ao *marketing* de proximidade, permitirá ainda analisar em tempo real o estado da rede, redireccionando as campanhas de *marketing* e maximizando o uso dos *hotspots* no envio de mensagens de publicidade.

Abstract

The Bluetooth technology opens a wide range of possibilities for those who intend fast and efficient wireless communications in a short distance. Used in PDA's, cell phones, computers and other devices, it can also be adopted to support and foster Bluetooth Marketing.

The present work intends to show the way Google Maps technology can be used to discover and monitor, in real time, a group of Bluetooth hotspots, using a mobile system. In this particular case, we also intend to show how to georeference and locate a hotspot Bluetooth and how to do it using a web-based application.

Once those hotspots are used to diffuse publicity, it was made an approach to Bluetooth marketing, as well as to the different technologies available today to georeference objects in a map.

The developed system, applied to Bluetooth marketing, allows the user to see, in real time, the state of a Bluetooth piconet, redirecting his marketing campaign and maximizing the use of the hotspots to send publicity messages.

Índice

Agradecimentos	III
Resumo.....	IV
Abstract	V
Índice de figuras	VIII
Índice de tabelas	IX
1. Introdução.....	1
1.1. Contextualização	2
1.2. Motivação e objectivos	6
1.3. Organização da dissertação	8
2. Trabalho relacionado	10
2.1. A tecnologia <i>Bluetooth</i>	11
2.1.1. Especificações gerais de comunicação.....	11
2.1.2. Método de funcionamento	15
2.1.3. Rede e dados.....	16
2.2. <i>Marketing</i> de proximidade.....	18
2.3. A integração da tecnologia <i>Bluetooth</i> com o marketing de proximidade	20
2.4. Georeferenciação	25
2.4.1. <i>Google Maps</i>	27
2.4.2. <i>Microsoft Virtual Earth</i> (actualmente “ <i>Microsoft Bing Maps</i> ”)	29
2.5. Usabilidade na <i>web</i>	29
3. Estudo de caso	33
3.1. Fundamentos do estudo de caso	34
3.2. Sistemas de Informação	35
3.2.1. Modelo em cascata	35
3.2.2. Modelo em V e modelo em espiral	37
3.2.3. Modelo de desenvolvimento rápido de aplicações	39
3.3. Identificação de requisitos – <i>SpectralBlue</i>	41
3.4. Análise de soluções	42

4.	Desenvolvimento da aplicação.....	44
4.1.	Desenvolvimento da aplicação.....	45
4.2.	Preparação e tratamento de dados	47
4.3.	Estrutura funcional da plataforma informática	48
4.3.1.	Funcionamento da API <i>Google Maps</i>	50
4.4.	Estrutura de implementação da plataforma informática	52
4.4.1.	Discussão da estrutura e funcionamento dos ficheiros necessários ao processo de georeferenciação dos <i>hotspots</i>	56
4.5.	Implementação, testes e usabilidade do sistema informático	63
4.5.1.	Implementação e testes em ambiente real	63
4.5.2.	Usabilidade do sistema informático.....	65
5.	Conclusão e trabalhos futuros	67
5.1.	Conclusões.....	68
5.2.	Trabalhos futuros	71
6.	Bibliografia	73
6.1.	Referências específicas	73
6.2.	Outras referências.....	75
7.	Anexos.....	76
	Anexo A – Documentação da base de dados <i>GMS</i>	77
	Anexo B – Código completo dos ficheiros da plataforma	79

Índice de figuras

Figura 1 - Pilha protocolar <i>Bluetooth</i>	13
Figura 2 - Arquitectura de transporte de dados (em camadas)	14
Figura 3 - Arquitectura do <i>system core</i>	14
Figura 4 - Funcionamento de uma rede <i>piconet Bluetooth</i>	15
Figura 5 - Comunicação de proximidade via <i>Bluetooth</i>	23
Figura 6 - Representação iconográfica usando o <i>Google Maps</i>	26
Figura 7 - Mapa de projecção Mercator.....	28
Figura 8 - Modelo em cascata	36
Figura 9 - Modelo em V.....	38
Figura 10 - Modelo em espiral	39
Figura 11 - Desenho do DSI	46
Figura 12 - Funcionamento da API <i>Google Maps</i>	49
Figura 13 - Estrutura simplificada da plataforma informática	53
Figura 14 - Exemplo de visualização de georeferenciação.....	55
Figura 15 - Aspecto geral do sistema informático desenvolvido	65

Índice de tabelas

Tabela 1 - Classes de dispositivos <i>Bluetooth</i>	12
Tabela 2 - Quadro comparativo <i>Bluetooth</i> vs outras tecnologias.....	23
Tabela 3 - Quadro-resumo dos diferentes modelos de DSI	39
Tabela 4 - Quadro-resumo das tabelas da base de dados GMS.....	46
Tabela 5 - Quadro resumo dos servidores de alojamento utilizados.....	63

1. Introdução

1.1. Contextualização

Ao contrário do que acontecia há alguns anos atrás, hoje é praticamente um acto comum a utilização de um sistema GPS (*Global Positioning System*) para localizar veículos, identificar estradas ou conhecer o caminho a percorrer. Estes sistemas tornaram extremamente fácil a localização de objectos, móveis ou não, elevando a portabilidade a um grau até antes impensável.

Para que um sistema GPS funcione de forma eficaz é necessário que este receba quatro sinais distintos, sendo três sinais de posição – latitude, longitude e altitude – e um de tempo, provenientes de satélites, os quais permitem localizar com elevada precisão um objecto móvel. Contudo, se esse objecto for estático, então a sua localização adquire outra dimensão, uma vez que deixa de ser necessária a utilização de um sistema de satélite que localize, em tempo real, o objecto pretendido. Reduz-se assim a localização apenas à existência das coordenadas geodésicas e a possíveis informações fornecidas directamente pelo objecto.

Os primeiros sistemas GPS apenas indicavam as coordenadas e permitiam marcar pontos de interesse; alguns destes aparelhos (mais avançados) ofereciam a opção de memorizar a rota utilizada ou a utilizar, mas nenhum deles permitia que o utilizador tivesse uma representação a 2D ou 3D da zona em que se encontrava.

Estes aparelhos eram, sobretudo, receptores de sinais de satélite que permitiam ao utilizador guiar-se num determinado trilho ou seguir uma determinada rota, mas que por vezes apresentavam dificuldades de recepção de sinal ou até alguns metros de distância entre a rota correcta e a representada.

Contudo, o advento da tecnologia e a rápida e crescente procura por aparelhos deste tipo levou a que a tecnologia fosse substancialmente melhorada, para que nos dias de hoje estes aparelhos já possuam erros de localização mínimos (ou até nulos) e funcionalidades totalmente diferentes daquelas que eram usadas há alguns anos atrás.

Mas aliar a representação de um local ou de uma rota com um mapa real, que permitisse ao utilizador identificar locais de uma forma muito mais aproximada da realidade, era o passo seguinte para os primeiros aparelhos de GPS. A difusão de aparelhos GPS e a sua utilização, aliada a uma forte procura, acabaram por dar também uma ajuda no que toca à necessidade de mapear todo o planeta de forma digital.

Surge então a representação do mapa no próprio aparelho de GPS, sobre o qual passou a ser possível representar a localização de um objecto, de forma semelhante à que era feita nos tradicionais mapas de estradas, onde se assinalavam os hotéis, parques, caminhos-de-ferro e outras indicações que se considerassem importantes.

Com o advento da Internet, a evolução das capacidades de computação, a elevada penetração dos sistemas móveis e a facilidade em partilhar informação, era quase lógica a necessidade de possuir sistemas de localização que permitissem uma real localização de um objecto também na *Web* e não apenas em sistemas GPS. Surgem, assim, os primeiros sistemas de georeferenciação baseados em coordenadas geodésicas comuns (latitude e longitude), seguindo-se a sua utilização em *sites* como forma de o utilizador “descobrir” onde se situa uma determinada empresa, um determinado local ou um determinado objecto.

Um dos sistemas de georeferenciação mais utilizado actualmente é o *Google Earth*, quer pela sua facilidade de utilização, quer pela forma como permite a qualquer utilizador “ver” qualquer parte do Mundo, já para não falar da questão dos custos. Além do *Google Earth*, existe também o *Google Maps*, que permite a um utilizador visualizar na Internet qualquer parte do globo terrestre, bem como ver as estradas, pontos de interesse e até fotografias dos diferentes locais. Mais recentemente, foi lançado em Portugal o *Google Street View*, um sistema que permite ver, a três dimensões, as ruas e outros objectos, tais como pessoas, carros, edifícios, etc, o qual poderá vir a alterar a forma como se entende o conceito de georeferenciação actualmente.

Contudo, existem ainda problemas que estão por resolver neste tipo de aplicações, sendo um deles o número de elementos representados de forma a obter uma representação inteligível e, simultaneamente, a existência de mecanismos de filtragem que permitam reduzir o número de elementos visualizados (1).

Além disso, é necessário considerar a representação num mapa só dos diferentes elementos, o que poderá trazer algumas dificuldades em termos de área de visualização.

Além do *Google Maps* existem outros tipos de ambiente que permitem a georeferenciação, como é o caso do *MetaCarta*, que poderá ser utilizado com o próprio *Google Maps*, com o *Microsoft Virtual Earth* ou até com outros sistemas.

A georeferenciação tem assumido, cada vez mais, um papel de destaque na sociedade e no mundo do trabalho. Actualmente, descobrir uma rua ou ver a nossa casa através de fotografias de satélite utilizando um receptor GPS ou um terminal de computador tornou-se um gesto bastante simples e praticamente ao alcance de todos. Mas o uso da georeferenciação não se limita à visualização de um local ou de uma determinada rua, mas antes a aplicar uma tecnologia de localização geográfica, em todo o mundo, associando-a a lugares e objectos concretos.

Com base nesta ideia, entende-se que a aplicação da georeferenciação a um determinado grupo de sistemas – neste caso, os *hotspots Bluetooth* – pode ser a solução para monitorizar e representar num mapa os sistemas que constituem uma rede. Este conceito, baseado na integração da localização geográfica de *hotspots Bluetooth*, torna-se assim possível a partir do momento em que se sabem as coordenadas de cada objecto.

Para as empresas de *marketing* de proximidade, a utilidade de um sistema de georeferenciação poderá ser muito vasta, uma vez que lhes permitirá saber a localização dos seus *hotspots* em tempo real, além de obter informações sobre os mesmos.

O *marketing* de proximidade é um tipo de *marketing* específico onde uma empresa difusora de conteúdos publicitários utiliza a tecnologia *Bluetooth* para difundir mensagens de publicidade de um determinado cliente. Uma vez que a tecnologia *Bluetooth* presente em telemóveis e similares não permite a recepção de mensagens num raio superior a 100 metros, este tipo de *marketing* é, regra geral, utilizado em espaços limitados, como feiras, cinemas, exposições, eventos, entre outros.

Analisando-se as soluções existentes no mercado, verificou-se que a difusão de conteúdos utilizando o conceito de *marketing* de proximidade e a tecnologia *Bluetooth* era já uma realidade estudada, implementada e testada. Existem já diversas empresas, em quase todos os países desenvolvidos, onde esta tecnologia de difusão de publicidade é já utilizada de forma comum em diversos eventos. Contudo, a georeferenciação dos *hotspots* que constituem a rede de dispositivos *Bluetooth* e a sua representação num mapa utilizando um computador, é algo praticamente inexistente.

Com base neste princípio, estudaram-se as soluções adoptadas por outras empresas e concluiu-se que a georeferenciação, pese embora apresentar interesse por parte de algumas destas empresas, tem sido um conceito que não tem tido uma abordagem directa e eficaz, não existindo plataformas que permitam monitorizar, em tempo real, o estado dos *hotspots* e fazer a sua representação num mapa. Estas empresas deram primazia ao problema da difusão das mensagens dos seus clientes, não se tendo debruçado tanto sobre a construção de um sistema de localização dos *hotspots* que constituem a sua rede.

Assim, e com vista a explorar este potencial existente, surgiu a ideia de elaborar uma aplicação informática que permita aliar, de forma dinâmica, a georeferenciação às informações que um dado objecto pode transmitir, utilizando a tecnologia actualmente disponível. Entre outras informações, a aplicação deverá dar conhecimento ao utilizador do total de *hotspots* existentes, do número de mensagens difundidas, da sua localização e possíveis comentários (se está *online*, a distância a outros *hotspot*, etc).

Refira-se, contudo, que este trabalho é absolutamente dinâmico, uma vez que desde o início deste estudo e da elaboração da aplicação informática até aos dias de hoje, muita coisa mudou em termos de georeferenciação e de implementação de sistemas de localização de objectos.

Isto porque, além da introdução da linguagem *AJAX* (*Asynchronous Javascript And XML*) na *API* do *Google Maps* e da difusão da tecnologia *Android* em receptores PDA, também se verificou recentemente a entrada em cena de serviços mais robustos e poderosos, como é o caso do controverso produto da *Google*, o *Street View*, que promete revolucionar a forma como é encarada a localização de uma rua ou de um determinado objecto.

1.2. Motivação e objectivos

O presente trabalho visa o estudo, identificação e construção de uma solução informática para monitorização e descoberta de *hotspots Bluetooth* por georeferenciação. As soluções existentes e a utilização da tecnologia *Google Maps* são temas que se pretendem ver referidos, nomeadamente ao nível da sua aplicação e utilização prática.

Pretende-se também entender até que ponto é possível conciliar uma tecnologia gratuita e disponível a qualquer pessoa com uma utilização mais robusta e empresarial, aplicando-a de forma directa a uma situação específica (neste caso, a georeferenciação de um objecto).

O problema inicial centra-se na construção de uma aplicação informática que permita georeferencial, em tempo real, objectos diversos entre os quais se podem incluir *hotspots Bluetooth*, câmaras de videovigilância, antenas de rádio, entre outros.

O objectivo deste trabalho é o de colmatar uma falha específica para as empresas que oferecem tecnologia de publicidade de proximidade, uma vez que não é usual disporem de um sistema que lhes permita, em tempo real, verificar a localização dos seus *hotspots* e obter informações sobre os mesmos ao nível da georeferenciação.

Para tal, pretende-se desenvolver uma solução informática em ambiente *Web* que permita a monitorização em tempo real de uma rede de *hotspots Bluetooth*, previamente instalados, utilizando mecanismos actuais de georeferenciação e localização geográfica. Paralelamente a este desenvolvimento, efectua-se o estudo de diferentes soluções existentes, bem como a forma de implementação das diferentes tecnologias disponíveis para este fim.

São também analisadas outras soluções possíveis e estudadas as formas possíveis de utilizar as API (*Application Programming Interface*) fornecidas pelo *Google Maps* para dar resposta à questão principal deste trabalho. De igual forma, proceder-se-á à análise do código utilizado nessas API e aborda-se a forma de integração do mesmo numa aplicação *Web based*.

Adicionalmente a todo o trabalho de implementação informática, analisa-se ainda de que forma a usabilidade poderá influir na utilização da solução informática, considerando as normas existentes nesta área e, por fim, evidencia-se a maneira como esta solução pode ser utilizada para a georeferenciação de outros sistemas, que não apenas os *hotspots Bluetooth*.

Além da construção da plataforma já referida, o presente trabalho tem também como objectivo a compreensão e “dissecação” da tecnologia *Google Maps* que actualmente está presente em inúmeros *sites*, ao nível da representação em mapas cartográficos digitais de determinados objectos, que vão desde a localização de uma determinada empresa até à representação de bombas de gasolina onde o combustível é mais barato.

Por fim, pretende-se também com este trabalho fornecer sugestões e melhorias ao nível da implementação, funcionamento e optimização da solução escolhida para a georeferenciação, bem como apresentar alternativas ao sistema escolhido, sinalizando diferentes problemas encontrados e outras possíveis abordagens na aplicação e utilização da tecnologia *Google Maps* e *Google Street View* para a georeferenciação de objectos.

1.3. Organização da dissertação

O presente documento encontra-se organizado como se descreve de seguida, tendo uma estrutura de cinco capítulos: introdução, trabalho relacionado, estudo de caso, desenvolvimento da aplicação e conclusões.

O primeiro capítulo visa descrever a contextualização da área de intervenção e do problema encontrado, além de referir introduzir o tema da georeferenciação aplicada a uma rede de *hotspots Bluetooth*. Neste capítulo, inclui-se também a motivação e objectivos a atingir com o presente trabalho, bem como as contribuições dadas ao projecto, além da descrição da organização de toda a tese apresentada.

No que se refere ao segundo capítulo, nele encontra-se descrito uma parte do trabalho relacionado e que servirá de base de conhecimento ao tema a tratar. Por ser um problema que envolve várias áreas distintas – *marketing* de proximidade, sistemas *Bluetooth*, georeferenciação e programação – procedeu-se a uma abordagem genérica a cada uma das situações, de forma diferenciada, com vista a balizar as várias dinâmicas geradas entre as diferentes áreas.

No terceiro capítulo – estudo de caso – pretende-se demonstrar o problema apresentado e que deu início a este trabalho, bem como referir as principais soluções já existentes no campo da georeferenciação. Descreve-se ainda prós e contras das diferentes soluções, bem como a forma como as mesmas foram analisadas.

No quarto capítulo, pretende-se demonstrar um problema concreto existente, proposto por uma empresa, descrevendo-se a forma como foi construída e como funciona o sistema informático proposto. Foi ainda analisado e comentado o código utilizado na API do *Google Maps* e a forma como este é integrada na linguagem *PHP* e na utilização de bases de dados *MySQL*. Referem-se ainda algumas das introduções feitas neste código e a forma como poderá ser aplicado em diferentes tipos de objectos que não apenas dispositivos *Bluetooth*.

No quinto capítulo são apresentadas as conclusões do trabalho realizado, descritos alguns possíveis melhoramentos da solução apresentada e referenciados trabalhos a desenvolver futuramente, com base no sistema informático apresentado. Reflecte-se ainda sobre as melhorias a implementar ao nível da usabilidade e da eficácia da aplicação, possíveis desenvolvimentos a ter em trabalhos futuros e sugestões de reflexão para eventuais aplicações ou melhorias de desempenho da solução apresentada.

2. Trabalho relacionado

2.1. A tecnologia *Bluetooth*

A tecnologia *Bluetooth* veio abrir portas a todos os que necessitam de uma tecnologia de comunicação de curto alcance mas que ao mesmo tempo ofereça simplicidade de utilização, robustez e fiabilidade. Difundida sobretudo entre os utilizadores de sistemas móveis, esta tecnologia oferece muito mais possibilidades, nomeadamente na divulgação de conteúdos multimédia e na transferência de dados entre dispositivos equipados com *Bluetooth*.

A tecnologia *Bluetooth* é uma tecnologia PAN (*Personal Area Network*) para transmissão de informação de curto alcance. Esta tecnologia foi desenvolvida, originalmente, pela IBM, Ericsson, Nokia, e Toshiba que entretanto com outras empresas igualmente conhecidas criaram o *Bluetooth Special Interest Group*, um consórcio formado com o objectivo de manter, desenvolver e assegurar compatibilidade através da certificação dos dispositivos lançados no mercado global pelos diferentes fabricantes.

Fácil de implementar, barata e eficaz, tem todos os ingredientes necessários para ser uma tecnologia muito difundida e utilizada (2), sobretudo por dispositivos onde todo o gasto de energia tem de ser ponderado e pode ser controlado por mecanismos decorrentes da própria norma. Tal como as restantes tecnologias de comunicação sem fios, também a tecnologia *Bluetooth* tem presente as preocupações ao nível da segurança, a interoperabilidade e da mobilidade.

2.1.1. Especificações gerais de comunicação

Ao nível técnico, o *Bluetooth* baseia-se em três itens: um RF *transceiver*, uma faixa espectral na banda ISM (*Industrial Scientific Medic*) e uma pilha protocolar. Operando na gama dos 2,4 GHz, esta tecnologia incorpora 4 modos de operação diferentes: *master* ou *slave*, activos e passivos.

Os sinais de rádio da tecnologia *Bluetooth* operam na gama de frequência entre os 2400 MHz e os 2483,5 MHz, utilizando 79 canais RF separados por 1 MHz, que se iniciam em 2402 MHz, mudando de canal a uma frequência de 1600 Hz (*hopping frequency*), com o objectivo de evitar as interferências com outros dispositivos que operem na mesma gama de frequências, bem como para introduzir mecanismos de segurança no que toca a privacidade da informação colocada no canal, já que a sequência de comutação nesses 79 canais depende de uma chave digital, do endereço de rede do *master* e do seu *clock* interno.

Foi recentemente aprovada a versão 3.0 da norma (Julho de 2009), que permitirá larguras de banda até 24 Mbit/s. Este aumento da taxa de transferência de dados está relacionado com o protocolo 802.11 e com a utilização da tecnologia *Alternate MAC/PHY* (AMP), uma junção da tecnologia *Media Access Control* (MAC) e da tecnologia da camada física da OSI (*physical layer*) no transporte de dados. O sinal rádio continuará a ser utilizado na descoberta, conexão e configuração dos dispositivos, mas quando for necessário transmitir dados em alta velocidade, esta tecnologia MAC/PHY pode ser adoptada.

Também nesta data foi introduzido um novo protocolo de gestão de energia, compatível com os dispositivos já existentes, designado por *Bluetooth Low Energy* (BLE), a qual traz uma forte melhoria na autonomia dos dispositivos móveis equipados com esta tecnologia.

As classes de antenas continuam confinadas a 3 tipos, conforme o seu alcance (de 1 metro até 100 metros).

Classe	Potência máxima (mW/dBm)	Alcance (metros)
1	100 / 20	Até 100 metros
2	2,5 / 4	Até 30 metros
3	1 / 0	Até 1 metro

Tabela 1 - Classes de dispositivos *Bluetooth*

Em termos de especificações, o protocolo de funcionamento *Bluetooth* comporta mais de mil páginas de especificações, incluindo também descrição dos protocolos ao nível do software e de hardware, tal como das diferentes camadas.

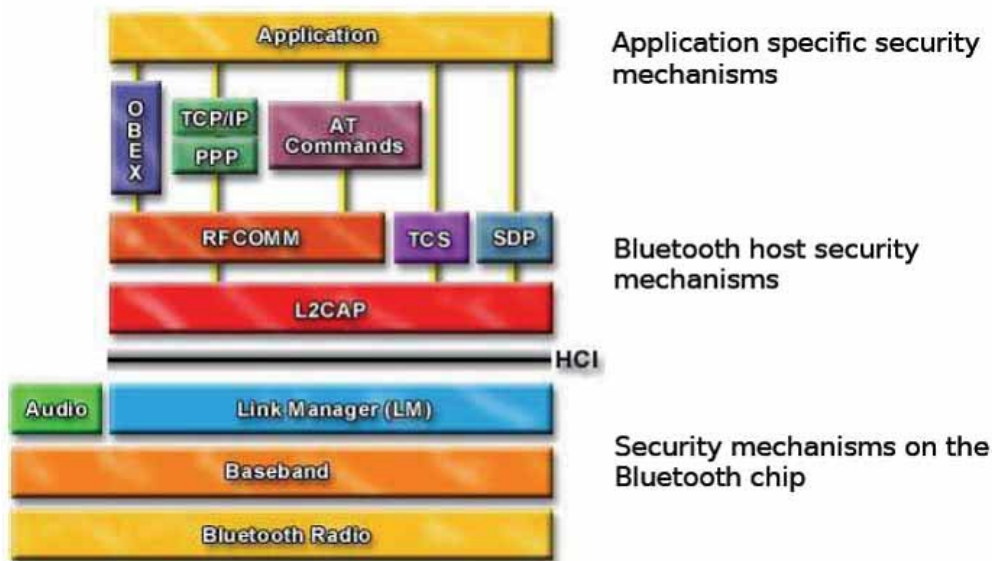


Figura 1 - Pilha protocolar Bluetooth (3)

O núcleo do protocolo é formado por uma pilha de cinco camadas, constituída por:

- *Radio*
- *Baseband*
- *Link Manager Protocol (LMP)*
- *Logical link control and adaptation control (L2CAP)*
- *Service discovery protocol*

Resumidamente, podemos dizer que a camada “Rádio” especifica os detalhes da interface, onde se incluem a frequência, o esquema de modulação e o alcance de transmissão. Ao nível da camada “Baseband”, encontram-se as questões relacionadas com a conexão na *piconet*, o endereçamento, o formato do pacote, o sincronismo e o controle de energia consumida. No LMP - *Link Manager Protocol* – tratam-se as questões relativas à segurança (autenticação e criptografia de dados), gestão das “requisições” de ligação externa e o controle do tamanho do pacote do *baseband*.

Já a camada L2CAP – *Logical link control and adaptation control* - trata da adaptação dos protocolos da camada superior à camada do *baseband* e o SDP fornece a informação dos dispositivos, os serviços e características dos serviços requeridos e que podem ser fornecidos para permitir a ligação entre dispositivos *Bluetooth*. A qualidade de serviço (QoS) é resolvida nesta camada, uma vez que os dispositivos *Bluetooth* analisam as comunicações que efectuam.

Assim, se um dos nós exigir uma largura de banda adicional e que não seja suportável, não estabelece a comunicação para esta não ficar afectada. Em alternativa, pode negociar uma taxa de transferência que consiga assegurar, de modo a estabelecer a ligação e esta não ficar comprometida.

O sistema de transporte de dados segue uma arquitectura em camadas, tal como representado na figura seguinte:

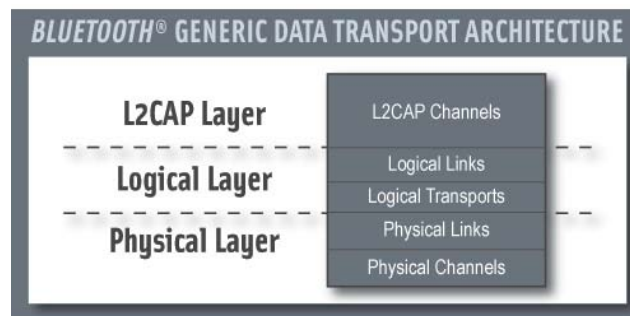


Figura 2 - Arquitectura de transporte de dados (em camadas) (6)

Já no que se refere ao *core* do sistema, este é constituído pelas quatro camadas e respectivos protocolos, definidos pelas especificações da tecnologia, bem como um protocolo de serviços em camadas, o protocolo de serviços de procura (SDP) e um perfil de acesso genérico (GAP).

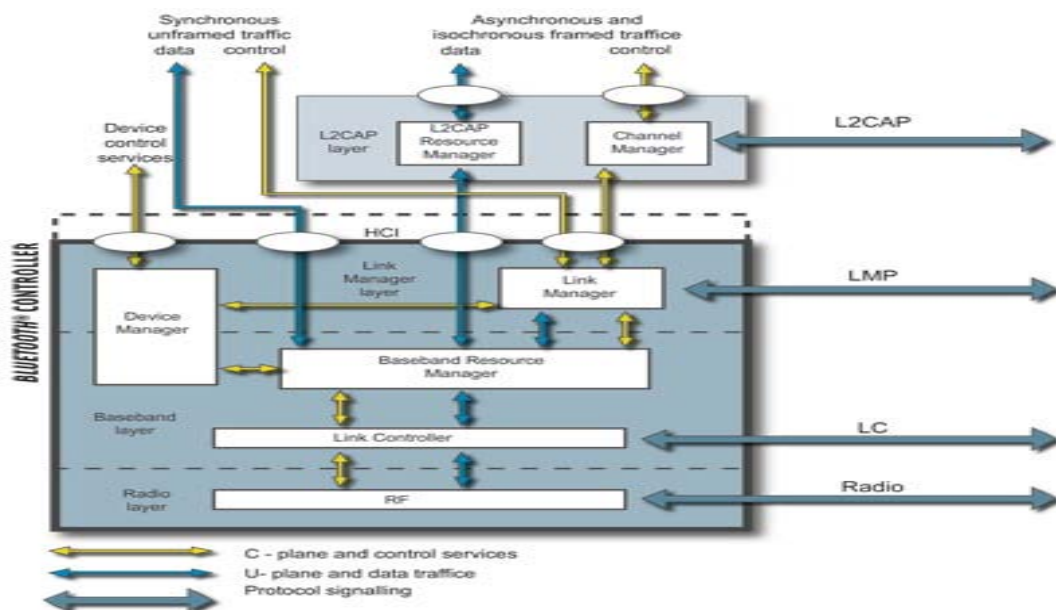


Figura 3 - Arquitectura do system core (6)

2.1.2. Método de funcionamento

Os dispositivos equipados com tecnologia *Bluetooth* ligam-se formando *piconets*, pequenas redes com um máximo de oito dispositivos, em que um dos dispositivos funciona como *master* e os restantes como *slaves*. Quando estes são conectados, forma-se uma *scatternet* com diversos nós, tal como ilustra a figura seguinte.

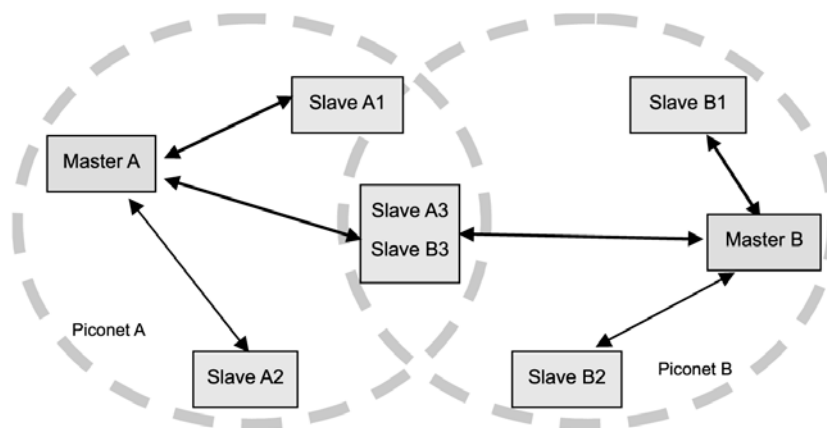


Figura 4 - Funcionamento de uma rede piconet Bluetooth (4)

Os dispositivos podem ter quatro estados básicos de operação:

- *Master* (mestre) – aquele que controla os restantes dispositivos;
- *Active Slave* (escravo activo) – aquele que está activamente ligado e que participa na rede *piconet*;
- *Passive slave* (escravo passivo) – aquele que continua ligado logicamente numa rede mas em baixa prioridade, embora permaneça sincronizado;
- *Standby* (em espera) – aquele que não está ligado na rede, aguardando pedidos ocasionais de outros dispositivos.

O modo *standby* é aquele em que os dispositivos se encontram quando não estão sincronizados ou coordenados de alguma forma. Neste modo, todos ficam “observando” a rede em momentos e frequências diferentes.

Durante este tempo, os dispositivos fazem monitorizações ocasionais, respondendo a pedidos e realizando paginações de modo a verificar se outros dispositivos tentam conectar-se. Este comportamento permite reduzir o consumo em 98%, devido às procuras serem ocasionais.

A paginação é o processo pelo qual um dispositivo fica a conhecer os restantes dispositivos que estão dentro da sua área de alcance. No caso de termos vários dispositivos, um determinado nó envia um pedido de informação aos restantes, que lhe respondem, sabendo quais os que estão dentro da sua área de alcance. Com esta informação, o dispositivo determina a identidade dos outros dispositivos, uma vez que cada um deles tem uma identificação única.

2.1.3. Rede e dados

As redes *Bluetooth* caracterizam-se por formarem uma rede *piconet* ou uma *PAN – Personal Area Network*, na qual podem estar associados entre dois a oito dispositivos. Esta rede consiste num conjunto de dois ou mais dispositivos que ocupam o mesmo canal físico, obrigando a que estejam sincronizados com um relógio comum.

Esta rede é possível devido a mecanismos de procura definidos na norma, que permitem encontrar outros dispositivos que possuam a mesma tecnologia e que se encontrem no seu raio de alcance. Assim, sempre que um dispositivo encontra outro, estes podem operar entre si e comunicar no contexto de uma rede *master-slave*, uma vez que cada dispositivo *master* suporta até oito *slaves* no estado activo.

Uma vez que os dispositivos são ligados de forma "ad-hoc", existe um *master* que define o relógio, o padrão de *frequency hopping* (FH) e define qual dispositivo que irá comunicar em cada *slot* de tempo. Esta tecnologia é uma das formas de transmitir sinais rádio, mudando rapidamente a onda portadora entre os diferentes canais, através de uma sequência conhecida entre emissor e receptor.

Os restantes dispositivos são considerados *slaves* e toda a comunicação é realizada entre *master* e *slave*, não sendo possível a comunicação directa entre *slaves*. Cada *master* pode controlar sete dispositivos *slaves* activos na rede.

No formato padrão de uma rede *piconet*, é executado um comando de paginação que estabelece uma ligação entre o *master* e o *slave*. Para criar esta conexão, suponhamos que um dispositivo A (*master*) transmite um comando de paginação a um dispositivo B (*slave*). Como cada dispositivo possui uma identificação própria, os restantes dispositivos ignoram a informação e, quando o dispositivo B responder ao pedido de A, é-lhe atribuído por este um número de identificação.

Após a conexão, o dispositivo B integra a rede *piconet* e passa a monitorizar a rede, aguardando comandos originados por A, tais como dados para sincronização ou intervalo de transmissão de dados, mantendo assim a sincronização contínua e apurada.

Quanto às *scatternets*, funcionam como um conjunto de *piconets* que estão ligadas entre si por nós comuns. Estas redes podem evoluir para estruturas bastante complexas, dado que um nó pode ser potencialmente um *master*, um *active slave* e um *standby slave*, simultaneamente. A única limitação deste tipo de rede é que cada nó pode ser somente o *master* de um *piconet* em um dado momento.

Ao nível da qualidade de serviço, também esta é assegurada pois cada dispositivo consegue informar os restantes da largura de banda máxima que suporta. Desta forma, os dispositivos conseguem assegurar uma largura de banda estável e transferir dados sem quebras no serviço, uma vez que não vão além dos seus limites e/ou das suas capacidades. O *paging* é o processo pelo qual um dispositivo fica a conhecer os restantes dispositivos que estão dentro da sua área de alcance.

No caso de termos vários dispositivos, um determinado nó envia um pedido de informação aos restantes, que lhe respondem, sabendo quais os que estão dentro da sua área de alcance. Com esta informação, o dispositivo determina a identidade dos outros dispositivos, uma vez que cada um deles tem uma identificação única, que é essencial para que seja possível o estabelecimento de uma ligação entre eles.

2.2. Marketing de proximidade

O marketing de proximidade, também conhecido como *Bluetooth marketing*, é um dos casos de aplicação prática da tecnologia em termos de meio de difusão de publicidade, o qual actualmente ainda está em expansão, utilizando um dos meios mais difundidos actualmente na sociedade: o telefone móvel.

Complementando as acções de marketing tradicional, o marketing de proximidade pode ser utilizado com vista a situações localizadas e restritas, dirigidas e personalizadas, aumentando o grau de eficácia do retorno da publicidade. Como é uma forma de publicidade relativamente pouco utilizada, permite às marcas transmitir uma ideia de inovação e de estarem na “linha da frente” em termos de difusão publicitária.

No caso do mercado português, este meio ainda não está amplamente difundido e não é comum vê-lo utilizado, ao contrário do que acontece em outros países, como o Brasil ou Espanha mas, atenta a este mercado, a *Mobile Marketing Agency* lançou em Julho de 2008 um código de conduta (5) para quem pretenda difundir publicidade neste tipo de mercado.

Apesar de em Portugal o marketing de proximidade ser um mercado em fase embrionária, com uma aplicabilidade muito baixa é, regra geral, este tipo de *e-marketing* que se começa a utilizar em eventos ou em alturas específicas, para determinados utilizadores em grupos restritos.

No caso nacional, existem algumas empresas que actualmente já fornecem serviços de marketing de proximidade, bastando os anunciantes solicitarem o serviço à empresa e esta trata da instalação e manutenção do equipamento informático e de comunicações associado.

Entre as empresas prestadoras de serviços nesta área, incluem-se casos como a *SpectralBlue* (Portugal), a *MobileBroadbands* (Portugal), *Mobimarket* ((Portugal), *Blueair* (Polónia), *BlueBlitz* (USA/Canadá), *Bluemagiq* (Síria), *Qwikker* (USA/UK), *ZonaBlu* (Espanha), *Futurlink* (Espanha), *BrasilTelecom* (Brasil), entre outras. Estas empresas são, na sua maioria, fornecedoras de serviços para utilização com campanhas de terceiros e na difusão de mensagens para diferentes clientes.

No entanto, optam pela criação e desenvolvimento de *hotspots* próprios, alugando-os ou vendendo-os. Salvo raras exceções (como o caso da *SpectralBlue*), praticamente todos os fornecedores deste tipo de serviço desenvolveram os seus próprios aparelhos de comunicação, não sendo muito usual a utilização de um computador comum no envio e difusão das mensagens.

Nestas soluções, torna-se então muito mais simples a constituição de uma rede de *hotspots*, dado que deixa de ser necessário recorrer à instalação de equipamentos especializados, podendo utilizar-se os computadores que já existem nos locais, o que vai agilizar ou potenciar de forma significativa a forma como a rede cresce em termos de distribuição geográfica.

Quanto aos anunciantes que já utilizaram o serviço, referem-se alguns casos com maior exposição na sua área de actuação:

- Campeonato Nacional de Ralis
- GE – General Electric
- Impresa
- Colgate-Palmolive
- Bertrand Livreiros
- Columbia TriStar Warners
- El Corte Ingles

Algumas destas marcas também anunciam os seus produtos segundo canais menos tradicionais, como a colocação de anúncios em motores de busca e em conteúdos para serem visualizados no telemóvel. Embora possa parecer que é um mercado residual, este tipo de anúncios e de publicidade já constitui uma fatia importante nas receitas dos principais fornecedores de serviços de publicidade móvel.

O princípio de funcionamento do serviço é o de um dado cliente contratar uma campanha de *marketing* de proximidade, o prestador de serviços instalar e configurar os *hotspots*, sendo depois a monitorização do decorrer da mesma feita por ambas as partes (cliente e prestador de serviços). No final do envio do número de mensagens contratadas ou do número de dias em que o serviço vai ser prestado, finaliza a campanha e termina o contrato.

Para a difusão destas mensagens, são utilizados *hotspots Bluetooth* que emitem as mensagens para os subscritores do serviço. Sempre que um determinado utilizador está ao alcance de um *hotspot*, e desde que tenha subscrito o serviço, recebe uma mensagem no seu telemóvel com a publicidade. Estas mensagens não se repetem pois o *Mac Address* do aparelho é registado automaticamente pelo sistema, que o identifica sempre que entra na zona de alcance de outros *hotspots*, não havendo portanto repetição da difusão de conteúdos.

Uma vez que a tecnologia *Bluetooth* permite o envio de mensagens com diferentes tipos de conteúdo (mensagens de texto, toques, *wallpapers*, etc), podem ser emitidas mensagens de diferentes tipos com diferentes conteúdos, conforme a campanha em curso que se pretende difundir.

A instalação desta tecnologia apenas requer que exista uma ligação à Internet, em qualquer altura e em qualquer lugar é possível ao utilizador aceder à sua campanha de *marketing*, de forma a visualizar os *hotspots* contratados e a forma como estes se encontram posicionados.

Analisando os dados da eMarketeer (<http://www.emarketer.com/Article.aspx?R=1005936>), prevê-se que só na Europa, as pesquisas efectuadas utilizando o *mobile marketing* venha a ser responsável por transacções na ordem dos quase 1000 milhões de dólares, em 2012, ligeiramente abaixo dos quase 1500 milhões de dólares nos Estados Unidos da América. Se compararmos com os valores de 2009, estaremos a falar de um crescimento previsto de cerca de 600% ao nível de valores, em dólares americanos.

Assim se percebe o motivo pelo qual o *marketing* de proximidade utilizando a tecnologia *Bluetooth* é um mercado apetecível, quer para anunciantes, quer para as empresas que detêm a tecnologia necessária a esse mercado.

2.3. A integração da tecnologia *Bluetooth* com o marketing de proximidade

O *marketing* de proximidade, também conhecido por "*Bluetooth marketing*" por utilizar a tecnologia *Bluetooth*, chegou a um momento importante.

Diversas marcas, tais como a Coca-Cola e a NASCAR, ou até alguns estúdios de produção cinematográfica de Hollywood e diversas bandas, estão a utilizar este meio de comunicação com o público.

Actualmente, estima-se que mais de 583 milhões de telemóveis estarão equipados e prontos para utilizar cabalmente a tecnologia *Bluetooth* em finais de 2009, comparados com uns meros 140 milhões em 2005 (Gartner). Isto mostra que, em termos globais, é uma tecnologia com crescente número de utilizadores, tendo levado 85% dos utilizadores a reconhecer que esta área de tecnologia é uma área que está prestes a ter a sua grande explosão e que é de fácil utilização (6).

Só em Portugal, e de acordo com dados da Marktest, existe já um universo de 1,7 milhões de utilizadores de tecnologia *Bluetooth*, representando cerca de 17% de potenciais consumidores de marketing de proximidade.

Uma das vantagens da utilização desta tecnologia sem fios no marketing de proximidade é a de ser uma tecnologia normalizada entre dispositivos de diferentes marcas e difundida em todos os continentes. Uma empresa americana poderá, por exemplo, difundir os seus conteúdos no Brasil, sem que haja necessidade de implementar uma nova tecnologia de difusão ou alterar os conteúdos já criados.

Ao nível de equipamentos móveis, há já uma série destes dispositivos tais como telemóveis, auriculares, computadores portáteis, teclados e ratos, entre outros, possuem já esta tecnologia de baixo custo, baixo consumo e de comunicação segura e sem problemas ao nível da sincronização.

No campo da publicidade de proximidade, abre-se assim uma nova porta, com a possibilidade de comunicar essa publicidade via *Bluetooth*. O marketing adquire assim uma nova dimensão pois, ao aliar o potencial tecnológico à facilidade de utilização dos aparelhos electrónicos actuais, verifica-se a existência de um grande potencial de implementação e utilização pelos consumidores. A transmissão de dados, de imagens ricas em conteúdo, de toques para telemóveis, de jogos e até de vídeo é, actualmente, uma realidade que, aliada à facilidade de utilização desta tecnologia faz com que a publicidade de proximidade encontre aqui um novo espaço de exploração.

Do ponto de vista tecnológico, utilizando o *Bluetooth* é ainda possível enviar material promocional rapidamente para um determinado grupo de utilizadores, restrito ou não, sem que seja necessária uma infra-estrutura de rede muito complexa ou com emissores e receptores fixos, tal como acontece com outro tipo de redes (por exemplo, redes LAN – *Local Area Network* - ou de telecomunicações móveis).

Do ponto de vista do consumidor, também existem vantagens, tais como a publicidade poder ser utilizada ou consultada mais tarde ou até utilizarem-se as informações recebidas com a função de *paperless* na utilização de alguns serviços (por exemplo, bilhetes para um espectáculo).

Do ponto de vista das marcas e dos agentes publicitários, esta tecnologia abre um novo espaço de exposição das marcas, uma vez que o utilizador pode partilhar a publicidade com terceiros, aumentando a exposição das mesmas marcas ou dos produtos anunciados.

Os equipamentos digitais que se encontrem equipados com *Bluetooth* podem ainda providenciar muitas outras vantagens para operadores e publicitários. Se a rede for instalada perto do ponto de compra de um determinado serviço, o utilizador pode receber vales no seu telemóvel. Estes, em formato de código de barras, podem ser utilizados posteriormente.

Fugindo um pouco do ponto de vista da publicidade, a tecnologia *Bluetooth* é também ideal para diversas localizações, como cinemas, teatros, eventos e comércio a retalho. A proximidade com o consumidor com uma determinada loja faz com que esta tecnologia seja a ideal quando se pretende informar alguém sobre um acontecimento, quando esta se aproxima de um determinado local. Pode ainda ser utilizada a tecnologia *Bluetooth* para exibir conteúdo sincronizado com um determinado acontecimento (por exemplo: uma apresentação de um livro). Também pode ser utilizada para descarregar músicas, *wallpapers* e outros conteúdos multimédia (7).



Figura 5 - Comunicação de proximidade via Bluetooth (fonte: BlueAir.pl)

Especificamente no caso dos cinemas e teatros, esta tecnologia pode até ser encarada como um complemento ao marketing pois podem ser distribuídos conteúdos que estejam relacionados com um determinado filme, musical ou espectáculo. No desporto aplica-se o mesmo conceito: o consumidor poderá receber directamente os resultados, horários ou informações da sua equipa, sempre que se aproxima do estádio onde se realiza o jogo.

O conceito pode ainda ser estendido a outros locais que não apenas os de consumo de publicidade, como é o caso de aeroportos, estações de comboio e metro, escolas, hotéis, feiras e congressos, entre outros, onde o utilizador pode receber informações úteis no seu telemóvel, desde que tenha subscrito o serviço.

Mas do ponto de vista informático e até estratégico, estas facilidades podem ser ainda mais importantes, se pensarmos que ao ser utilizada irá permitir obter informação sobre os hábitos de consumo das pessoas, dados para análise de *Return on Investment* (ROI), localização dos respectivos consumidores através do tráfego da rede, índice de localização demográfica, entre outros.

No que se refere a comparação com outras tecnologias, o *Bluetooth* também leva vantagens, sobre outras tecnologias utilizadas em telemóveis, como as SMS, *Wi-Fi* ou a Internet móvel (3G), as quais não foram capazes de permitir uma troca de informação de imagens e vídeo tão eficaz entre as pessoas que levasse os publicitários a apostar em utilizá-las na divulgação de publicidade. Tem ainda a vantagem de que a publicidade só é entregue aqueles que se encontrem nas proximidades do ponto de acesso ou os que requisitaram o respectivo serviço para uma determinada área.

Das principais vantagens da tecnologia Bluetooth, há ainda a referir o ponto de vista energético. Com as especificações lançadas sobre a versão 3.0, esta tecnologia torna-se das mais eficientes em termos de utilização de energia, sobretudo em dispositivos móveis, uma vez que prolonga a duração das baterias e reduz substancialmente o consumo de energia.

	SMS	WiFi	3G	Bluetooth
Transmissão livre	Não	Sim	Não	Sim
Rich content	Não	Sim		Sim
Descarregar conteúdos para posterior visualização	Sim	Não	Não	Sim
“Componente viral”	Sim	Não	Não	Sim
Rápida penetração no mercado	Muito alta	Muito baixa	Média	Alta

Tabela 2 - Quadro comparativo Bluetooth vs outras tecnologias (6)

Verifica-se então que existem muitas e variadas razões para que o marketing via Bluetooth tenha o seu grande momento. De entre estas, destacam-se principalmente (8):

- 1) A elevada taxa de penetração dos telemóveis em todo o mundo que, no caso português, é superior à média europeia.
- 2) A facilidade de comunicação e de utilização dos dispositivos equipados com Bluetooth, sem que seja necessária tecnologia complexa.
- 3) O tempo que os consumidores actuais passam actualmente fora de casa, estando assim mais expostos ao consumo, aos negócios e à publicidade.
- 4) A possibilidade do consumidor receber gratuitamente os conteúdos no seu telemóvel, podendo utilizá-los mais tarde, à medida que aumenta o número de aparelhos equipados com esta tecnologia e a capacidade de memória destes. Ainda que estes conteúdos não sejam adequados para visionamento directo no telemóvel, o utilizador pode descarregá-los mais tarde no seu computador e visioná-los.
- 5) Para os publicitários, a vantagem de não terem custos elevados com os operadores de redes móveis, podendo divulgar os conteúdos que pretenderem.

- 6) A possibilidade de divulgar conteúdos multimédia já existentes para telemóvel, da mesma forma que o fazem actualmente com a televisão ou a internet.
- 7) A não utilização de recursos como o papel ou a publicidade gratuita não desejada, eliminando resíduos, trabalho e custos diversos de fabrico, distribuição e actualização de publicidade.

Não obstante todos estes pontos, há ainda a referir que a tecnologia *Bluetooth*, por aliar o baixo custo, a simplicidade e a comunicação sem fios, começa a tornar-se um caso sério de estudo para publicitários e *marketeers*, não só nos Estados Unidos da América, mas também a nível mundial.

2.4. Georeferenciação

A georeferenciação é a tecnologia que permite a localização e representação, num mapa, de um determinado edifício, veículo ou até pessoa, recorrendo às três coordenadas geodésicas comuns: latitude, longitude e altitude. Estas são as mesmas coordenadas num vulgar sistema de GPS, com a vantagem de que neste caso não é necessário ter os cuidados relativos à sincronização de tempos entre os vários relógios, razão pela qual o factor tempo é eliminado.

O principal ponto de partida de um sistema de georeferenciação consiste na obtenção de coordenadas geográficas de latitude e longitude dos objectos que se pretendem representar nos mapas cartográficos. A este processo dá-se o nome de *geocoding* e pode ser feito utilizando diferentes ferramentas, como é o caso do próprio *Google Maps* ou utilizando outros sites, como é o caso de <http://www.travelgis.com/geocode/>.

No caso de os objectos serem móveis, o processo de representação em tempo real faz-se da mesma forma que nos sistemas de navegação GPS exceptuando a variável tempo. As coordenadas recebidas através de sinais de satélite são processadas e, sobre um mapa cartográfico, são representados ícones que identificam os diferentes locais ou objectos.

Caso os objectos a representar sejam fixos, então a representação é feita através de coordenadas cartográficas a partir do endereço postal, utilizando-se *gateways* de geocodificação. Neste caso, as coordenadas de latitude e longitude do endereço são calculadas e armazenadas numa base de dados, a partir da qual é feita a leitura quando se pretende representar os diferentes objectos, após a respectiva conversão para quilómetros ou milhas das distâncias entre eles.

Este sistema de georeferenciação funciona com base em duas tecnologias distintas: a conversão de um dado local em coordenadas geodésicas (latitude e longitude) e a representação iconográfica de um determinado objecto num local preciso do mapa cartográfico. Para tal, calculam-se as coordenadas de posição do objecto e, utilizando um mapa digital, é aposto um ícone no preciso local onde o objecto se encontra. A figura seguinte ilustra esta situação.

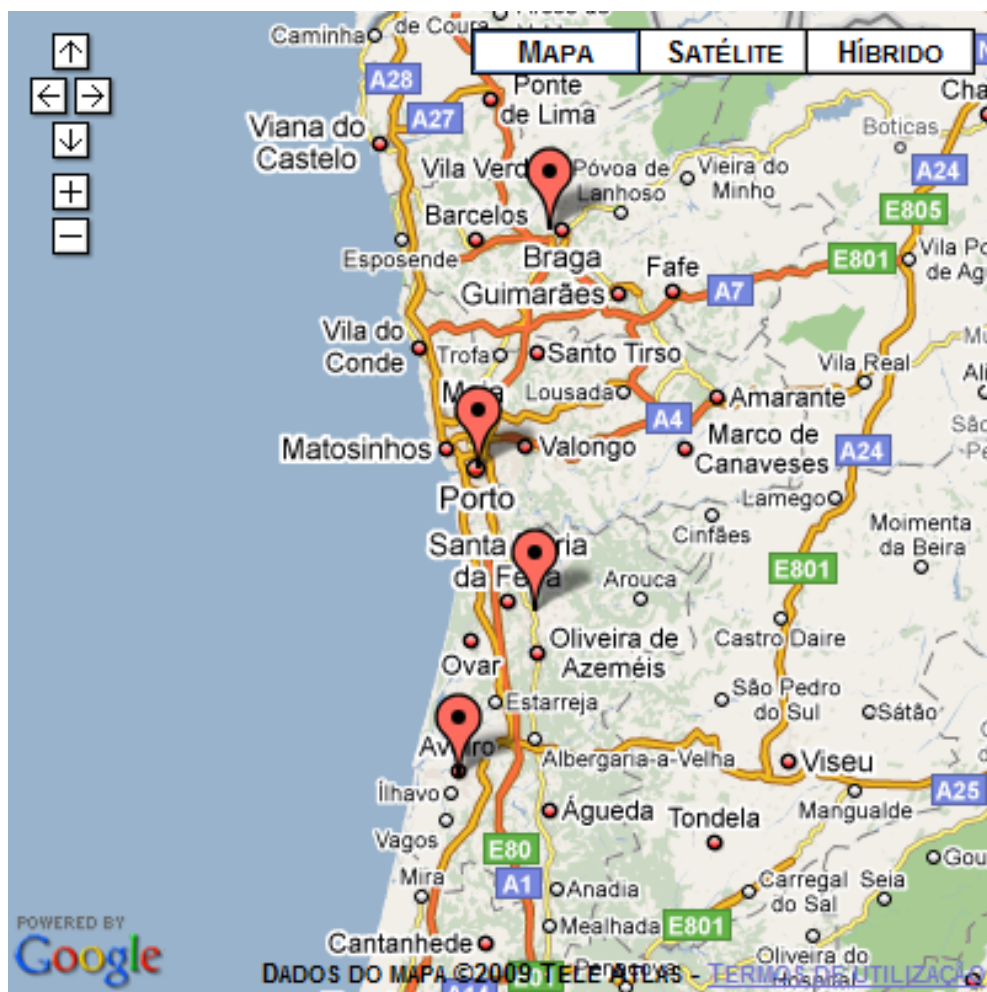


Figura 6 - Representação iconográfica usando o *Google Maps*

Actualmente, as tecnologias de representação cartográfica mais utilizadas e robustas são o *Google Maps*, o *Yahoo! Maps* e o *Microsoft Virtual Earth* (actualmente *Microsoft Bing Maps*). Estes sistemas abriram portas para a representação de informação sobre mapas virtuais, o que constituiu a base dos sistemas de georeferenciação actual, à semelhança do que já acontecia em sistemas de navegação GPS.

Contudo, e devido ao carácter mais comercial de algumas tecnologias, a facilidade de utilização e a possibilidade de alteração do código de programação é uma mais-valia que nem sempre está disponível. Outras tecnologias como a *Nasa World Wind* não permitem ainda a representação iconográfica de objectos sobre os mapas topográficos, pelo que não serão objecto de estudo neste trabalho.

2.4.1. *Google Maps*

No caso do *Google Maps*, trata-se de uma aplicação que permite mapear, via Web, um determinado local do mundo. Fazem parte deste sistema outras aplicações como o *Google Ride Finder* ou o *Google Transit*, que são representações mais detalhadas de estradas ou de tráfego rodoviário, utilizando a API do *Google Maps*.

A tecnologia empregue no *Google Maps* tem por base a projecção *Mercator*, uma projecção cilíndrica de um mapa geográfico, introduzida em 1569 pelo cartógrafo Gerardus Mercator. Este mapa tornou-se *standard* para a navegação marítima, devido à facilidade na representação de linhas e rotas através de elementos rectos.

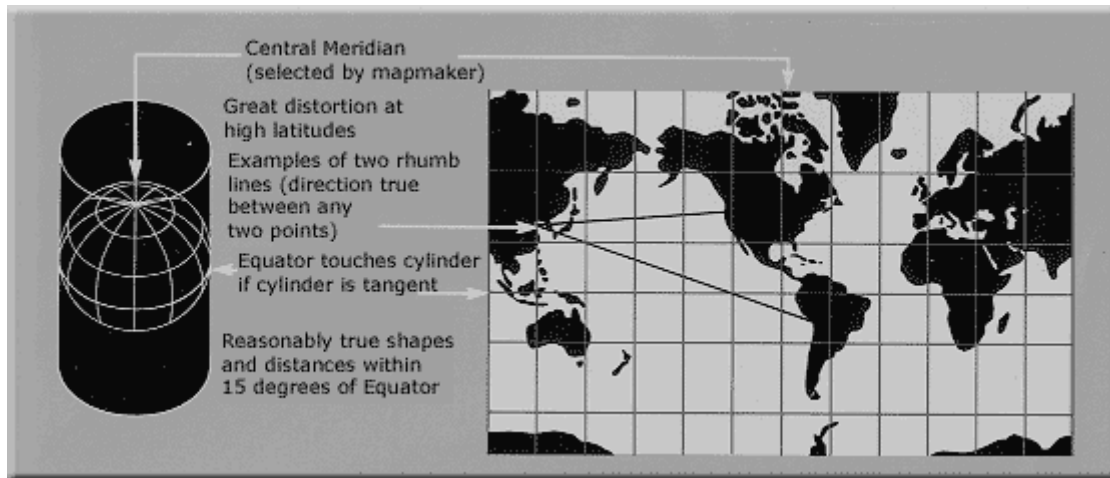


Figura 7 - Mapa de projecção Mercator - <http://erg.usgs.gov/isb/pubs/MapProjections/graphics/mercator.gif>

A tecnologia *Google Maps* converte um determinado endereço em coordenadas de latitude e longitude, representando-as depois num mapa da Terra idêntico ao da figura anterior. Pese embora a alta definição e detalhe com que as imagens são tiradas, nem todos os locais do planeta se encontram representados, como é o caso dos pólos Norte e Sul ou ainda de algumas ilhas, como a ilha da Madeira.

Ao nível de implementação, o *Google Maps* utiliza o *Javascript* de forma extensa. Inicialmente, o mapa terrestre é segmentado em pequenas “células”, que são representadas a partir de um servidor e inseridas na página Web. Quando um utilizador introduz um endereço, o mapa é centrado e é representado o local numa barra lateral. Se o utilizador arrasta o mapa, as “novas células” são introduzidas mas a página não sofre um *refresh*, mantendo-se as localizações no mapa inicial.

No que toca aos objectos, estes são representados sobre o mapa através de um ícone vermelho (imagem PNG). É também usada uma técnica denominada de JSON (9) – (*JavaScript Object Notation*) em vez de XML (*eXtensible Markup Language*), por razões de desempenho e velocidade na representação dos dados, pese embora ambas assentem em linguagem AJAX. A restante informação geográfica de representação cartográfica é disponibilizada pela TeleAtlas.

Em termos de interface de aplicação, o *Google Maps* desenvolveu uma API que permite aos programadores integrarem esta tecnologia em ambiente *Web* com a representação de objectos próprios. Este serviço é gratuito e não contém publicidade, mas nos seus termos de utilização está prevista a introdução de publicidade.

Utilizando esta API, é possível embeber todas funcionalidades do *Google Maps* num *website* externo. Esta API necessita de uma chave própria, que pode ser obtida através de um registo, e é possível ainda alterar e representar informação dos objectos, sendo apenas necessário introduzir o código *Javascript* fornecido (ver anexo A).

O utilizador apenas tem de fornecer as coordenadas de latitude e longitude, tendo a API a capacidade de georeferenciar os endereços e, posteriormente, representá-los. Outras possibilidades, como a API para dispositivos móveis também está disponível para utilização.

2.4.2. Microsoft Virtual Earth (actualmente “Microsoft Bing Maps”)

O serviço disponibilizado pela *Microsoft*, actualmente designado por *Bing Maps*, é composto por três serviços: *Bing Map Control*, *Bing Maps Web Service* e *MapPoint Web Service*.

O primeiro serviço inclui imagens e permite o desenvolvimento de aplicações para serviços empresariais, com um modelo de programação baseado em *JavaScript*. Já o segundo serviço – *Bing Map Web Services* – oferece imagens estáticas, além de *geocoding* e *reverse geocoding*. No que toca ao 3º serviço, este é um tipo de serviço dirigido para a programação *Web*, alojado pela *Microsoft* e usado por empresas e programadores de *software*, com vista à integração de serviços e aplicações em processos de negócio.

2.5. Usabilidade na web

O termo “usabilidade” é a “medida pela qual um determinado produto pode ser usado por um utilizador específico, com vista a alcançar objectivos específicos com efectividade, eficiência e satisfação, num contexto de uso específico” (10).

Além de funcionalidades correctas, eficiência de utilização e facilidade em aprender, a usabilidade incorpora também a tolerância aos erros do utilizador e a satisfação desse mesmo utilizador.

A medida da usabilidade é entendida como os valores resultantes da medição dos processos utilizados para avaliar a efectividade e capacidade de um utilizador alcançar os seus objectivos iniciais, quer em termos de qualidade, quer em termos de resultado final obtido. O termo efectividade refere-se à quantidade de recursos e ao esforço necessário para alcançar um objectivo, sendo que os desvios efectuados e a quantidade de erros cometidos pelo utilizador também são avaliados para avaliar, no cômputo geral, o nível de eficiência de um *site*.

A satisfação é a medida mais difícil de medir e quantificar, já que se relaciona com factores subjectivos, pese embora o facto de se referir ao nível de conforto sentido pelo utilizador e a aceitação da maneira com que alcança os seus objectivos ao navegar num determinado *site*.

A medida da usabilidade também pode ser, segundo a mesma norma (10), utilizada segundo outras perspectivas como sejam:

- Facilidade de aprendizagem: o utilizador consegue rapidamente realizar as tarefas que pretende.
- Facilidade de memorização: algum tempo após utilizar um *site*, o utilizador ainda se recorda do seu funcionamento e não necessita reaprender a utilizá-lo.
- Baixa taxa de erros: o utilizador realiza as tarefas com poucos erros e consegue recuperar rapidamente dos mesmos.

Na Internet, a usabilidade de uma determinada ferramenta informática é fundamental para o sucesso. Por exemplo, numa loja *online*, se o utilizador não encontrar o artigo pretendido, acaba por mudar para outro sítio pois, actualmente, a oferta é enorme e basta uma simples selecção no rato. Se antes um utilizador clássico ia a uma determinada loja pagava primeiro o artigo para experimentar depois a usabilidade, o utilizador da Internet experimenta a usabilidade primeiro e só paga depois.

Entre as regras básicas de usabilidade de um *site* destacam-se as seguintes:

- 1) Clareza no acesso à informação: um *sítio Web* bem estruturado e desenhado é uma mais-valia para cativar o utilizador, pois este terá sempre dificuldade em encontrar o que procura.
- 2) Facilidade de navegação: o utilizador deve conseguir a informação que pretende com apenas três acções de selecção.
- 3) Qualidade e rapidez: quem navega quer encontrar rapidamente aquilo que procura, com qualidade e simplicidade.
- 4) Conteúdos claros: quem navega, pretende que o texto seja o mais conciso e fiável possível. Os leitores *online* terão a necessidade de imprimir e, por isso, os textos devem ser curtos.
- 5) Consistência: um *site* deve manter a sua estrutura nas diferentes páginas, de forma a que o utilizador navegue mais rapidamente e veja o *site* como um projecto único.
- 6) Simplificar, reduzir e otimizar: o tempo para carregar uma página deve ser curto, pois o tempo que as pessoas aguardam por uma página antes de perder o interesse situa-se entre os 10 e os 15 segundos.
- 7) Colocar as conclusões no início: se o utilizador vir as “metas” no início, satisfaz as suas intenções mais rapidamente e poderá, *a posteriori*, dedicar algum tempo a explorar o *site*.

Para se realizarem testes de usabilidade, é necessário dispor de um determinado número de utilizadores que utilizem o *site*. Estes utilizadores são designados para desenvolver tarefas específicas e os dados obtidos após a observação da realização dessas tarefas são tratados, com vista a perceber como é que o utilizador interage com o *site*.

Entre os processos observados incluem-se a interacção do utilizador com o ambiente gráfico, os erros cometidos, a frustração, a rapidez com que realiza uma dada tarefa e a satisfação do utilizador. Da mesma forma é considerado o julgamento individual de seu uso através de um conjunto implícito ou explícito de utilizadores (10).

Este processo é mais dispendioso pois necessita de pessoas que terão de testar a usabilidade de um site e, para tal, são utilizados métodos para desenvolver e testar protótipos, avaliar alternativas, analisar problemas de usabilidade, propor soluções e realizar testes com utilizadores. A este processo dá-se o nome de “Engenharia da Usabilidade” (11) e apresenta uma lista de etapas que descreve a sequência do processo de engenharia da usabilidade:

- Definir os objectivos de usabilidade utilizando métricas adequadas.
- Especificar os níveis de usabilidade planeados e que têm de ser alcançados.
- Analisar o impacto das possíveis soluções adoptadas no projecto final.
- Incorporar o retorno devido à experiência do utilizador no projecto final.
- Realizar iterações no ciclo “projecto-avaliação-projecto”, até que os níveis planeados sejam alcançados.

No que toca à avaliação da usabilidade, esta pode ser feita através de testes aos utilizadores, onde se incluem as técnicas de avaliação da ergonomia em sistemas interactivos, e que são:

- Avaliação heurística.
- Critérios de ergonomia.
- Inspeção baseada em padrões, guias de estilos e guias de recomendações.
- Inspeção utilizando uma lista de verificação (*checklist*).
- Teste empírico com utilizadores.
- Entrevistas e questionários.

Contudo, o ideal será utilizar diferentes métodos de avaliação da usabilidade, como forma de compreender as dificuldades dos utilizadores. Entre os mais utilizados encontra-se a avaliação heurística – por permitir apurar alguns erros menos frequentes – e as entrevistas e questionários – por permitirem ouvir a opinião dos utilizadores reais e as suas sugestões de melhoramentos.

3. Estudio de caso

3.1. Fundamentos do estudo de caso

Tal como foi referido no ponto 1.2. do presente documento, os objectivos deste trabalho centravam-se em duas vertentes: por um lado, desenvolver uma solução informática em ambiente *Web* que permitisse à empresa *SpectralBlue* a monitorização em tempo real de uma rede de *hotspots Bluetooth*, utilizando mecanismos actuais de georeferenciação e localização geográfica. Por outro lado, também era objectivo deste trabalho tentar compreender de que forma funciona a tecnologia *Google Maps* e como poderia a mesma ser utilizada para dar resposta a este problema específico.

Além disso, pretendia-se também que o sistema fosse capaz de representar informação adicional, tal como o número de mensagens enviadas, total de utilizadores, capacidade de alcance das antenas, entre outros itens.

Os referidos *hotspots Bluetooth* estavam já instalados e, dado que já estavam em funcionamento, tornou-se necessário proceder à aplicação de um modelo de desenvolvimento de sistemas de informação que incluísse a análise de requisitos, o desenvolvimento do sistema de informação, a sua posterior validação e a utilização por parte de diferentes indivíduos.

O trabalho foi assim dividido em várias etapas, tendo por base a utilização de um modelo em cascata resumido mas com prototipagem. Assim, o desenvolvimento do sistema de informação incluiu as etapas de identificação de requisitos, desenho do sistema de informação, implementação e testes, a par do estudo de soluções, que respondam às necessidades evidenciadas na identificação dos requisitos.

A parte referente à identificação de requisitos encontra-se neste capítulo (ponto 3.3), enquanto a parte de desenho e desenvolvimento do sistema de informação, implementação e testes se encontra desenvolvida no capítulo seguinte (pontos 4.2, 4.3 e 4.4, respectivamente).

3.2. Sistemas de Informação

Para desenvolver um sistema de informação (SI), seja em que área for, é sempre necessário introduzir mudanças específicas nos sistemas já existentes, de forma a obter melhorias no desempenho, baseando-se tais mudanças nas tecnologias da informação (TI) (12). Assim, o planeamento de um sistema de informação deve ser construído de forma a evitar informação fragmentada, passando esta a ser mais directa, concisa e fiável.

Um processo de desenvolvimento de um sistema de informação (DSI) baseia-se sobretudo no princípio de que não há um desenvolvimento ideal, mas antes um desenvolvimento que permita otimizar um determinado processo de informação. Contudo, é por vezes notória a confusão entre desenvolvimento de sistemas de informação e desenvolvimento de *software*, uma vez que muitos dos processos de desenvolvimento de um sistema de informação terminam com o desenvolvimento de *software*.

Para desenvolver um sistema de informação existem actualmente alguns processos bem definidos, que vão desde o desenvolvimento sequencial ao desenvolvimento evolutivo, passando também pelo desenvolvimento incremental. No primeiro, destaca-se o modelo em cascata, enquanto no segundo temos o desenvolvimento em espiral e, no terceiro, aparece o desenvolvimento por graus de importância.

Assim, poderíamos classificar os paradigmas do desenvolvimento de sistemas de informação, conforme os modelos a adoptar, originando-se a seguinte distribuição:

- Modelo em cascata;
- Modelo em V;
- Modelo em espiral;
- Desenvolvimento rápido de aplicações (RAD);

3.2.1. Modelo em cascata

Na base de um modelo em cascata (*Waterfall Model*), desenvolvido nos anos 70, encontra-se o desenvolvimento por etapas.

Neste tipo de modelo, o processo pode voltar atrás as vezes necessárias até que cada etapa esteja cumprida, havendo assim interação entre as diferentes fases. A figura seguinte ilustra este processo:

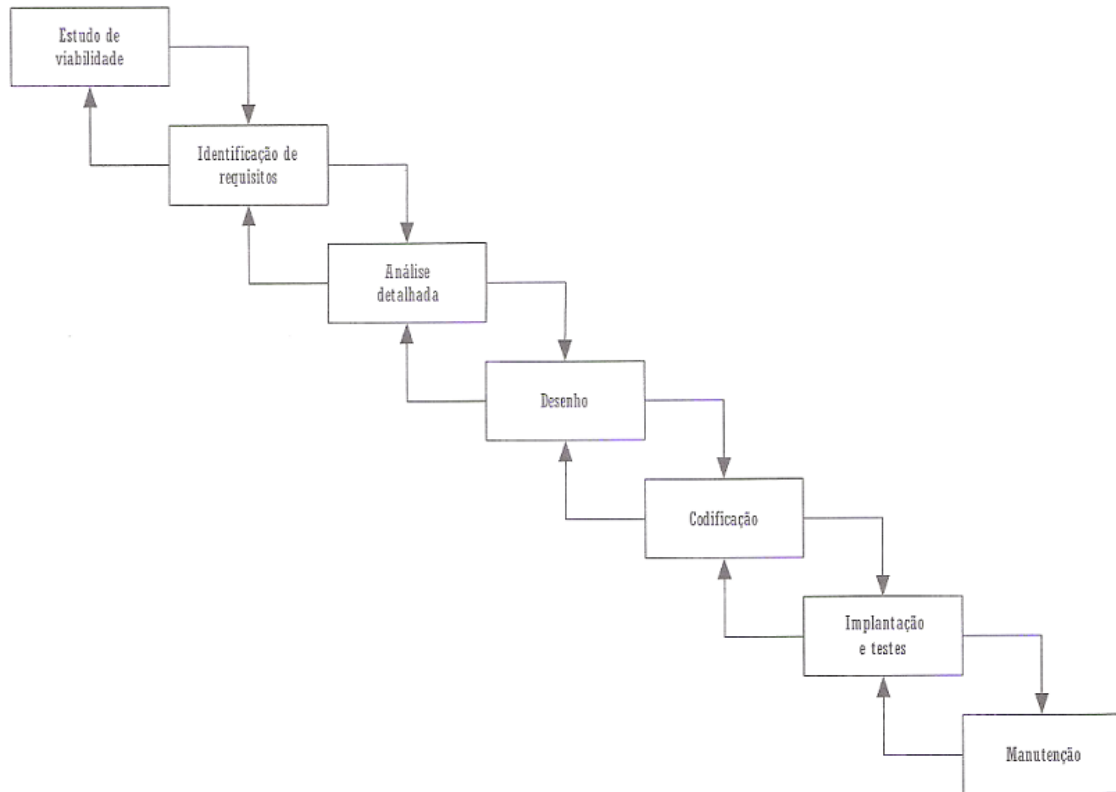


Figura 8 - Modelo em cascata

Todas as diferentes fases descritas na figura são obrigatórias para o desenvolvimento do sistema de informação. No entanto, destacam-se o estudo de viabilidade – análise do problema e apresentação de possíveis soluções – e o desenho – desenvolvimento da arquitectura de um sistema através da especificação de cada componente, uma vez que são etapas cruciais no desenvolvimento de um sistema de informação por serem aquelas que constituem os pilares do desenvolvimento destes sistemas.

Além disso, há ainda a destacar a parte de implantação e testes, que consiste na etapa em que se executam testes e se substitui ou converte o sistema antigo para o novo, incluindo-se nesta parte a formação dos utilizadores.

Apesar da abordagem através deste método poder parecer bastante interessante, existem algumas críticas a este tipo de desenvolvimento de SI, nomeadamente (13):

- Pouca informação sobre o contexto de negócio para o qual o SI é desenvolvido;
- O baixo envolvimento do utilizador no processo de desenvolvimento do SI;
- A possível demora entre a especificação de requisitos e a entrega do sistema, levando a que os requisitos por vezes se alterem com o tempo.
- A não inclusão da validação por parte dos utilizadores durante o processo de desenvolvimento.

Contudo, apesar destas lacunas, é possível ultrapassar as mesmas se se considerar a associação deste modelo com outros modelos existentes, como é o caso do modelo RAD ou o modelo em V. Para tal, é usual incluir neste modelo a prototipagem, um processo que resulta na inclusão de um protótipo, cujo objectivo é ser explorado, experimentado e avaliado através de sucessivas iterações, com vista a introduzir as necessárias melhorias. Neste caso, o desenho final do sistema só é elaborado após a identificação dos requisitos e a análise concreta e detalhada de todos os objectivos aos quais o sistema de informação terá de dar resposta.

3.2.2. Modelo em V e modelo em espiral

Ainda na linha do modelo em cascata surgiram, no final dos anos 80, o modelo em V (*V Model*), e o modelo em espiral (*Spiral Model*). Apesar de as diferenças serem bastantes entre os dois modelos, o modelo em V destaca-se por apresentar um conjunto de fases ordenadas, estando o desenvolvimento do sistema de informação dividido em duas partes, conforme se trate do desenho ou da construção do sistema de informação.

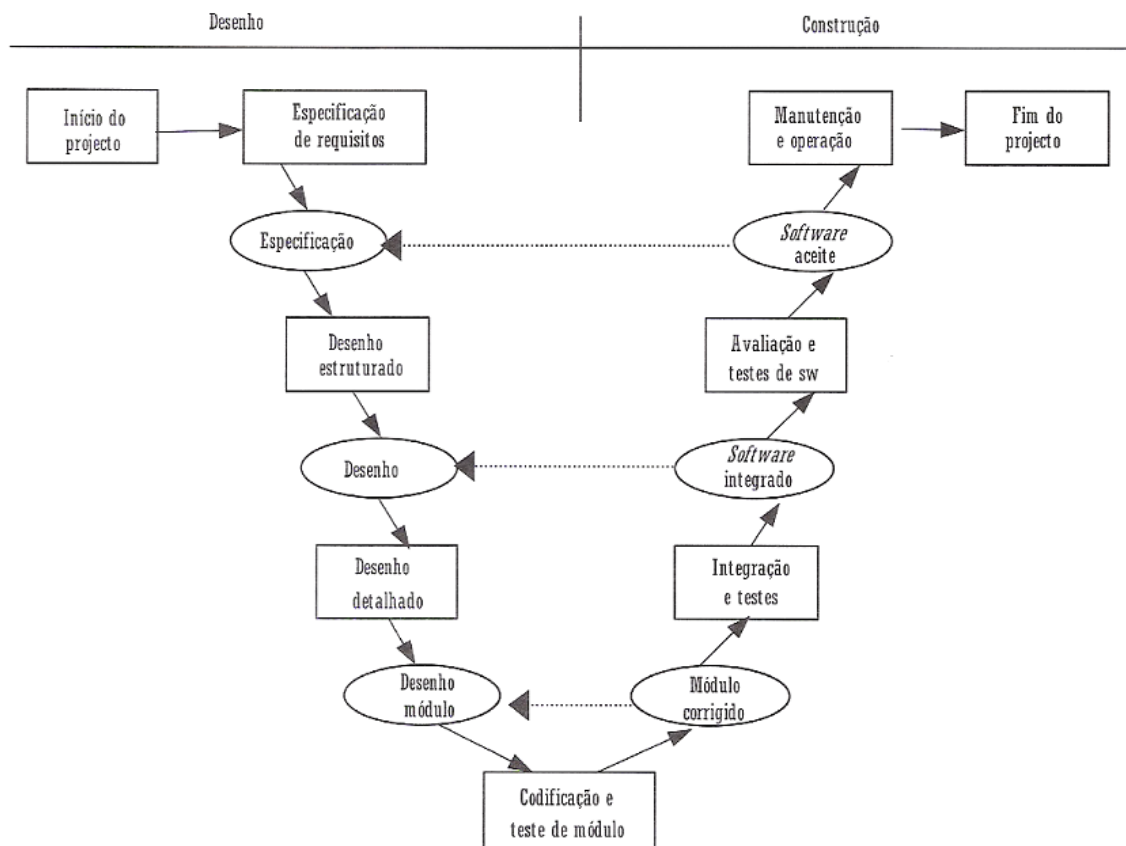


Figura 9 - Modelo em V

Já no modelo em espiral desenvolvido, também nos finais da década de 80, apresenta uma nova fase aos modelos existentes, ao incluir a análise de risco como uma das partes cruciais do desenvolvimento de um SI. Este modelo segue uma lógica de planeamento, análise de risco, engenharia (desenvolvimento da aplicação) e avaliação do sistema de informação.

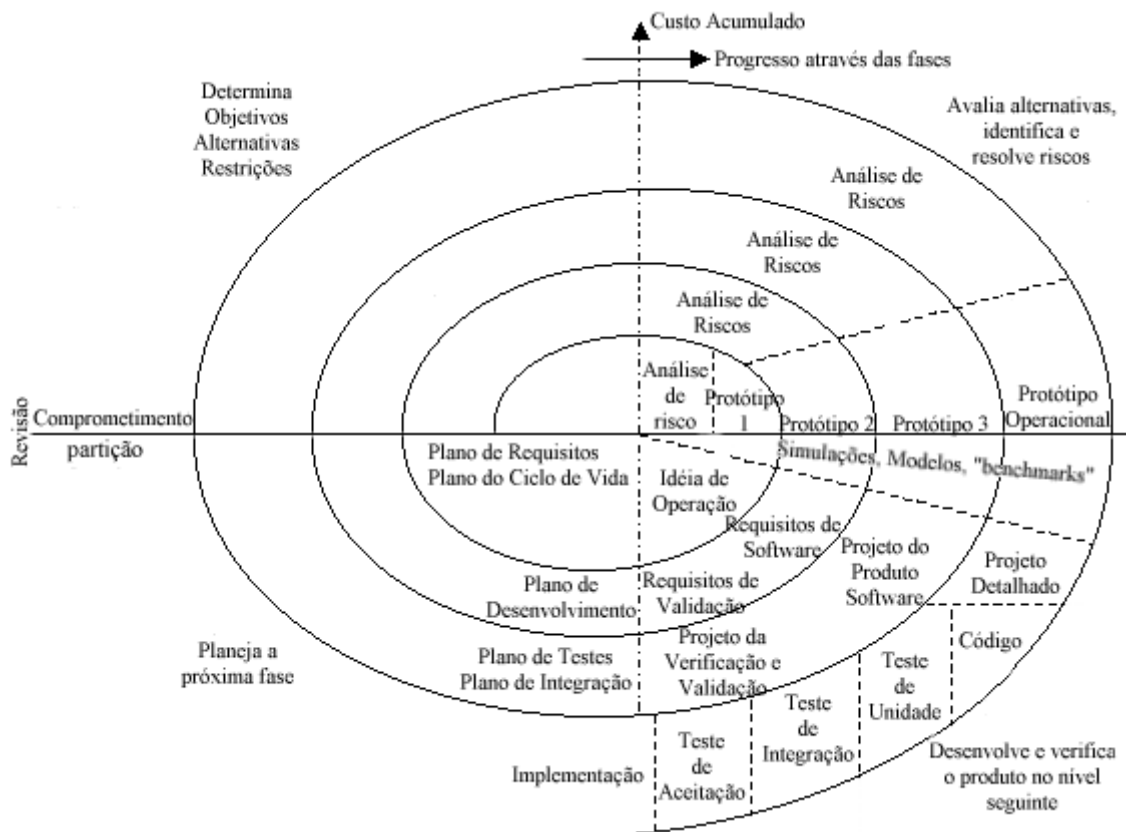


Figura 10 - Modelo em espiral

Este modelo pretende assim integrar os melhores aspectos do modelo em cascata, aliando a análise do risco em cada parte do processo, o que poderá reduzir os problemas do sistema de informação final.

3.2.3. Modelo de desenvolvimento rápido de aplicações

O modelo RAD é um tipo de modelo incremental, que visa sobretudo diminuir o tempo de desenvolvimento do SI. Baseia-se no desenvolvimento de um grupo de componentes que funciona como uma espécie de núcleo, a partir dos quais todo o sistema de informação vai sendo gerado.

Este “núcleo” de componentes permite assim aos utilizadores utilizarem mais cedo o sistema de informação, uma vez que algumas das funções necessárias nem sempre são utilizadas no imediato, as quais poderão ser desenvolvidas mais tarde, numa regra de 80/20 (80% das funcionalidades são rápidas a desenvolver, enquanto apenas requerem 20% requerem um grande esforço)

Num projecto deste tipo, adopta-se assim a aproximação MoSCoW, ou seja, a aproximação em que temos requisitos escalonados conforme o seu grau de importância. A saber:

- *Must have*: aqueles requisitos sem os quais o sistema não é viável;
- *Should have*: aqueles requisitos que não põem em causa o sucesso do SI, mas que trazem benefícios ao mesmo;
- *Could have*: requisitos sem impacto no projecto mas que virão ser implementados mais tarde;
- *Won't have*: tipo de requisitos que podem nunca vir a ser implementados, sem que haja prejuízos para o SI.

Este tipo de modelo é usado quando se pretende um desenvolvimento rápido de um sistema de informação, muitas vezes utilizado em programação por módulos, como é o caso de ambientes *intranet* ou quando se torna necessário ir adicionando funções.

O quadro-resumo seguinte ilustra as principais fases de cada um dos processos atrás referidos.

Modelo em cascata	Modelo em V	Modelo em espiral	Modelo RAD
<ul style="list-style-type: none"> • Estudo de viabilidade • Identificação de requisitos • Desenho • Codificação • Implementação e testes 	<ul style="list-style-type: none"> • Especificação de requisitos • Desenho estruturado • Avaliação e testes de <i>software</i> 	<ul style="list-style-type: none"> • Identificação de requisitos • Análise de risco • Desenho de <i>software</i> • Validação e verificação 	<ul style="list-style-type: none"> • Análise e especificação de requisitos • Desenho • Codificação e testes • Implementação

Tabela 3 - Quadro-resumo dos diferentes modelos de DSI

Em todos os modelos está prevista a etapa de manutenção, posterior à implementação em ambiente real dos sistemas de informação obtidos.

3.3. Identificação de requisitos – *SpectralBlue*

A empresa *SpectralBlue* é, actualmente, uma empresa dedicada à intervenção em áreas de mercado emergentes, como é o *marketing* de proximidade. Neste contexto, desenvolveu uma plataforma em linguagem *Java*, que permite o envio de mensagens de publicidade através de *hotspots Bluetooth*, contendo diversos conteúdos, que vão desde a simples imagem ao som, animações e vídeos.

Ao nível tecnológico, esta estrutura assenta numa base de dados elaborada em *MySQL*, para a qual as antenas enviam diferentes informações como, por exemplo, o *MAC address* do aparelho receptor, o número de mensagens enviadas, o modelo das antenas que estão a emitir a emitir, a largura de banda, entre outras informações. Essas informações ficam armazenadas na base de dados, podendo ser utilizadas para diversos fins, alguns deles utilizados pela aplicação informática desenvolvida pela *SpectralBlue*.

O desenvolvimento e utilização de um sistema informático de georeferenciação foram os objectivos principais propostos pela *SpectralBlue*. Contudo, na identificação de requisitos, concluiu-se que a mesma deveria dar resposta aos seguintes itens, não obstante virem a ser incluídos outros:

- Localizar no mapa todos os *hotspots (online e offline)*, bem como o número de mensagens enviadas.
- Apresentar, para cada *hotspot* da respectiva área de cobertura e sabendo a georeferência, a escala actual do mapa e o tipo de antena usada.
- Calcular o acumulado das áreas de cobertura, dando informação em qualquer instante da área de cobertura global de todos os nossos *hotspots*.
- Providenciar uma ferramenta de georeferenciação que possa ser utilizada pelo cliente no sentido de lhe permitir georeferenciar o seu *hotspot*, sem ser necessário o recurso a um GPS.

- Manter um “top 10” dos *hotspots* com o maior número de *hits*, o maior número de mensagens enviadas e os dias com mais *hotspots* online.
- Associar a cada *hotspot* um conjunto de atributos, cujos valores podem variar dinamicamente, com o objectivo de posteriormente se poderem filtrar “vistas” do mundo em que esses *hotspots* apareçam representados de acordo com algum paradigma (por exemplo, a sua localização por país ou distrito).

A fim de suportar as funções que pudessem dar resposta a estes requisitos, a *SpectralBlue* forneceu o esquema da base de dados já em utilização pela sua aplicação informática. Contudo, e de acordo com estas exigências, teve-se o cuidado de analisar previamente a base de dados, a qual revelou algumas lacunas, nomeadamente ao nível do registo dos utilizadores e à possível hierarquia dos mesmos (por ex: administrador de sistema ou cliente). Por esta razão, teve de se proceder a algumas modificações na mesma, aquando do seu tratamento e que se encontram descritas no ponto 4.2.

3.4. Análise de soluções

Com vista a responder aos requisitos pré-estabelecidos, foi efectuado um estudo em duas vertentes: 1) analisar de que forma a concorrência georeferenciava os seus *hotspots* e 2) estudar a melhor solução a desenvolver, tendo por base as tecnologias *Google Maps* ou *Virtual Earth*.

Em relação à primeira situação, tentou-se contactar diferentes empresas que prestam um serviço semelhante ao da *SpectralBlue*. De entre estas empresas, destacam-se a *BluAir* que é, actualmente, o maior operador na área do *marketing* de publicidade no espaço europeu. Curiosamente, nas diferentes pesquisas e contactos efectuados, foi possível verificar que nenhum dos diferentes operadores possuía uma plataforma informática que realizasse a georeferenciação dos seus *hotspots*, apenas existindo desenvolvida a interface informática de envio de mensagens.

Já ao nível da integração da tecnologia *Google Maps* ou *Microsoft Virtual Earth*, a opção recaiu sobre a solução *Google Maps*, devido às seguintes condições:

- O *Google Maps* tem praticamente georeferenciada toda a superfície terrestre.
- Está prevista a utilização de bases de dados em *MySQL* de raiz, de forma a georeferenciar objectos.
- A API fornecida pelo *Google Maps* está desenvolvida em PHP e XML, sendo rápida a sua integração em ambiente *Web*.
- É uma solução *open-source*, baseada na licença GNU/GPL (*General Public License*).
- É fornecida a API e é possível alterar o código da mesma, sem que haja restrições temporais ao seu uso.
- Pode ser integrada na tecnologia *Android*, com vista ao desenvolvimento de aplicações para aparelhos móveis (telemóveis e PDA's).
- Existe, actualmente, uma comunidade bastante alargada de pessoas que se dedicam ao desenvolvimento e partilha de experiências e conhecimento sobre a plataforma *Google Maps*.

Por todas estas razões apresentadas, e pelo facto de a *SpectralBlue* ter também desenvolvido a sua base de dados em *MySQL*, a escolha natural acabou por ser a solução *Google Maps*, em detrimento da solução da *Microsoft*. Além disso, é uma solução robusta e testada, que permite a utilização sem custos e sem restrições, estando já bastante referenciada e amadurecida, quer ao nível de código, quer ao nível de opções que permite.

Já no que toca aos modelos de desenvolvimento de sistemas de informação, e cruzando todos os requisitos identificados com a tecnologia já existente, optou-se por apresentar uma solução baseada no desenvolvimento em cascata com prototipagem, uma vez que isto permitiria à *SpectralBlue* testar o sistema desenvolvido, introduzindo-se ao longo do tempo as melhorias que fossem sendo assinaladas.

4. Desenvolvimento da aplicação

4.1. Desenvolvimento da aplicação

Tal como referido anteriormente, utilizou-se um modelo em cascata com prototipagem. Desta forma, identificaram-se os requisitos (ponto 3.3) e, de seguida, procedeu-se ao desenho da aplicação que deveria dar resposta a esses mesmos requisitos. Mais tarde, procedeu-se ao desenvolvimento da aplicação (onde se inclui o tratamento de dados), seguindo-se a implementação e a fase de testes. A fase de manutenção não está prevista neste trabalho, uma vez que a aplicação informática não está totalmente desenvolvida.

Na fase de desenho da aplicação, foram produzidas especificações sobre o sistema informático, descrevendo as componentes de dados, interfaces e processos, com base na especificação dos requisitos resultantes, considerando também as soluções possíveis descritas no ponto 3.4. Foram também elaborados os formulários e menus para entrada de dados, definida a arquitectura do sistema informático, a linguagem e os processos necessários para satisfazer os requisitos necessários e ainda o código usado no sistema para codificar as diferentes entidades.

Uma vez que parte deste trabalho já tinha sido realizado por parte da *SpectralBlue*, sobretudo ao nível da base de dados, não foi necessário proceder à realização de diagramas E-R (entidade-relação) para os dados, pese embora o facto de as dependências funcionais terem de ser ligeiramente alteradas, para dar resposta a requisitos da aplicação. A figura seguinte ilustra, de forma resumida, as etapas do desenho da aplicação até ao início da codificação.

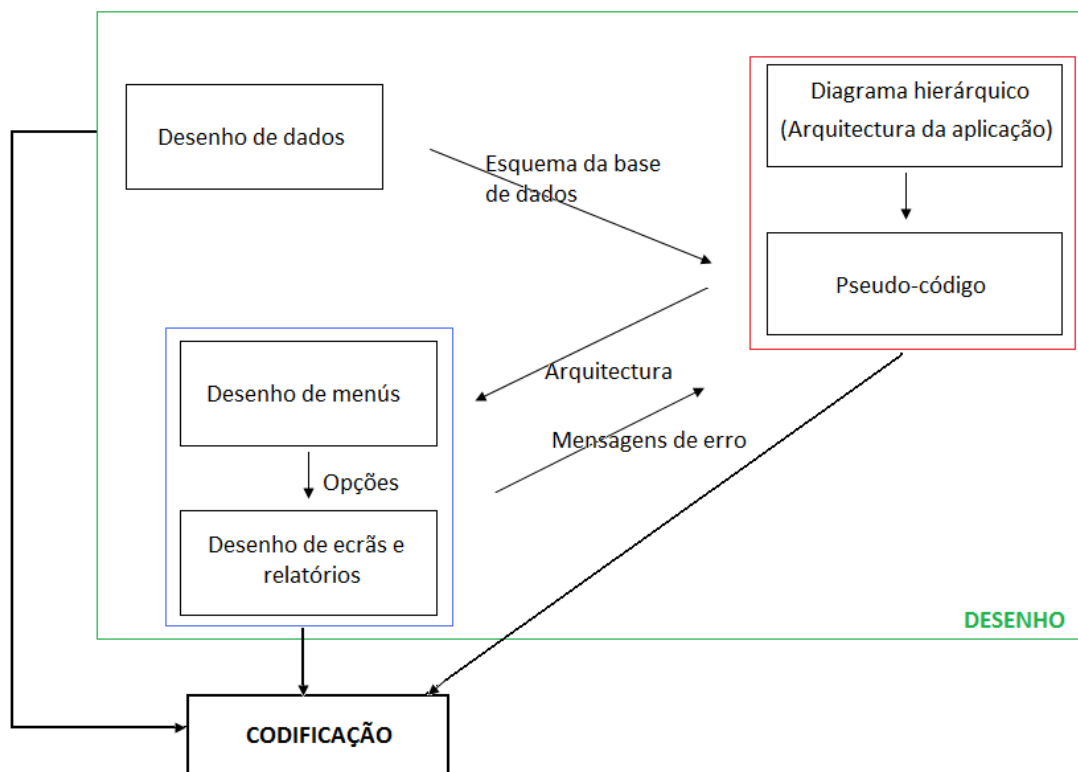


Figura 11 - Desenho do DSI

Após a fase de desenho da aplicação, passou-se à fase de desenvolvimento e implementação. No que se refere ao desenvolvimento específico da aplicação, este comportou duas etapas: numa fase inicial, teve de se proceder à preparação e tratamento de dados já existentes, bem como efectuar a normalização dos mesmos. Posteriormente, pretendia-se o desenvolvimento de uma plataforma informática que desse resposta às necessidades exigidas nos requisitos anteriormente expostos.

De seguida, e já após o desenvolvimento da aplicação, passou-se à fase de testes onde o objectivo foi maximizar a usabilidade da plataforma já criada e incrementar o número de funções que a mesma pode oferecer, bem como explorar ao máximo o seu potencial (por exemplo, adaptando-a para sistemas móveis). Neste momento, a aplicação encontra-se nesta fase, uma vez que se pretende ainda incrementar o número de funções a disponibilizar e testar as já existentes.

As diferentes etapas do processo de desenvolvimento da aplicação encontram-se descritas nos pontos seguintes com mais pormenor.

4.2. Preparação e tratamento de dados

Numa primeira fase, o trabalho centrou-se sobretudo na análise da base de dados já existente, na sua organização e na normalização e identificação dos mecanismos de recolha de dados já em funcionamento.

Iniciou-se a análise e preparação prévia da base de dados, cujo nome é *GMS*, a qual foi fornecida pela *SpectralBlue*. O desenho final desta base de dados pode ser consultada no anexo A e a sua análise consistiu na verificação dos parâmetros utilizados e no tipo de dados associados, analisando-se também a mesma no que toca a relações E-R, usando-se “dados-teste” para verificar que todas as entidades primárias possuíam as devidas correspondências. Além disso, foram analisados os tipos de dados previstos para a base de dados, campo a campo, com vista a utilizá-los e integrá-los mais tarde com a plataforma informática que viria a ser desenvolvida.

Uma vez que no sistema informático utilizado pela *SpectralBlue* não é necessário o registo de utilizadores, não existia nenhuma tabela na base de dados que contivesse informações sobre utilizadores. Assim, foi também necessário acrescentar algumas tabelas que dizem respeito aos utilizadores e às permissões (“níveis”) que estes venham a deter, tal como ficou decidido aquando do planeamento da plataforma informática. Acrescentaram-se, por isso, as tabelas “Utilizadores” e “Níveis”, ficando a base de dados final com um total de 10 tabelas, e que são:

Nome da tabela	Contém informações sobre:
<i>Btmajorclass</i>	Classes de dispositivos.
<i>Btmanufacturer</i>	Fabricantes secundários de dispositivos.
<i>Btminorclass</i>	- - -
<i>Btremotedevice</i>	Dispositivo (nome, classe e fabricante)
<i>Btremotedevicefound</i>	Estado do dispositivo (QoS, RSSI, quantidade de mensagens enviadas, etc)
<i>Btscanner</i>	- - -
<i>Btscannerstart</i>	Localização do dispositivo (nome, latitude, longitude, endereço, etc)
<i>Btsupermanufacturer</i>	Fabricantes principais de dispositivos.
<i>Utilizadores</i>	Diferentes utilizadores (nome, endereço, e-mail, nível, etc).
<i>Niveis</i>	Diferentes níveis de utilizador (administrador, cliente, marketeer, etc)

Tabela 4 - Quadro-resumo das tabelas da base de dados GMS

Os campos contidos em cada uma das tabelas encontram-se documentados no anexo A deste documento.

Por fim, refira-se que como a base de dados fornecida se encontrava vazia (“despovoadá”), foi também necessário proceder ao seu povoamento. Neste povoamento, teve-se o cuidado de analisar e verificar os possíveis erros, entre os quais, nomes de ruas incompletos, erros ortográficos e erros de digitação do nome das vias tanto nas moradas dos participantes quanto no mapa digital.

4.3. Estrutura funcional da plataforma informática

A introdução do *Google Maps* e a possibilidade de utilização de API’s para integração desta solução em páginas de *Web* veio permitir a ampla aplicação da georeferenciação em diversas situações.

Se por um lado é possível referenciar diversos itens no *Google Maps*, como o local onde moramos ou onde trabalhamos, por outro é também possível importar parte da tecnologia utilizada pelo próprio *Google* para marcar e identificar locais nas nossas próprias páginas de internet.

Ao serem introduzidas as coordenadas geodésicas comuns (latitude e longitude), pelo utilizador, dá-se a sua conversão no servidor de modo a que possam ser interpretadas pelo próprio servidor do *Google Maps*, desde que este possua a chave própria.

Para tal, é usada uma API (15) que analisa as coordenadas introduzidas e cria um ficheiro XML que contém as informações sobre a localização pretendida. Este ficheiro é depois interpretado por outra API que identifica as coordenadas fornecidas, importando do *Google Maps* uma parte do mapa onde estas coordenadas se encontram e, de seguida, procede à marcação do local, através de ícones. A figura seguinte ilustra esta situação:

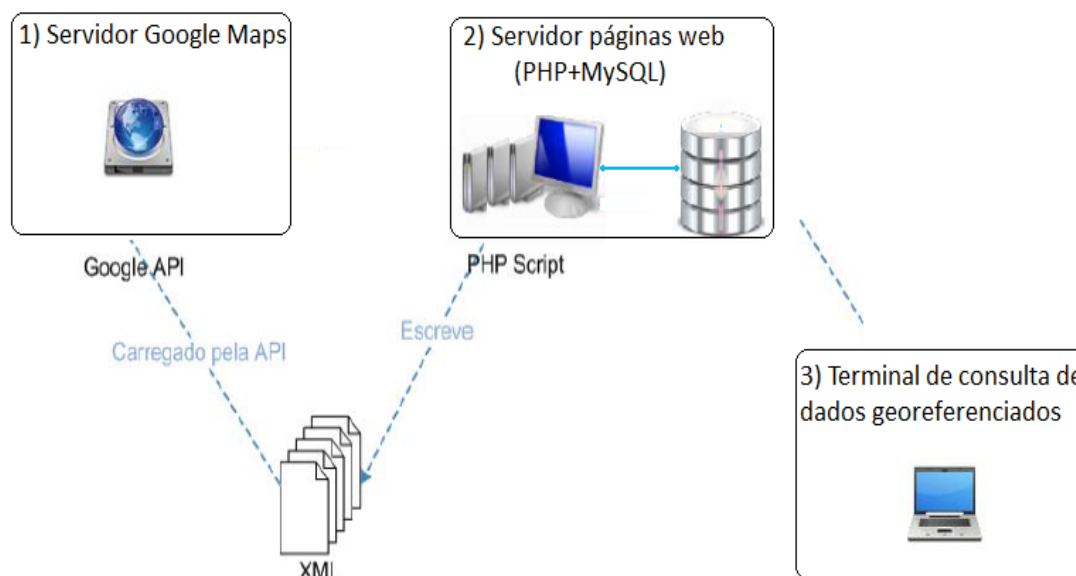


Figura 12 - Funcionamento da API Google Maps

Se associarmos o envio de informação em tempo real para uma base de dados com as coordenadas dos diferentes *hotspots*, então é possível utilizar a tecnologia de georeferenciação aplicada a um caso concreto: a georeferenciação em tempo real dos *hotspots Bluetooth*.

Esta tecnologia já era utilizada em sistemas GPS mas actualmente existe mais interesse na mesma, uma vez que é possível qualquer empresa ou instituição mostrar um mapa electrónico aos seus visitantes do local onde se encontra, com informações que vão desde o dito local até ao seu número de contacto ou e-mail.

Concluído que estava o tratamento da base de dados e a sua adaptação e normalização, passou-se à fase de desenvolvimento da plataforma informática. Para tal, procedeu-se ao desenho da plataforma informática, ao nível de *layout*, funções a suportar, tipo de menus, entre outros. Paralelamente, fez-se o estudo do funcionamento da API do *Google Maps*, a qual permite a integração directa do código em ambiente PHP, sendo o seu modo de funcionamento descrito no ponto 4.3.1.

A solução informática desenvolvida teve como objectivo o de permitir georeferenciar os *hotspots Bluetooth* de uma rede *piconet*, formada por emissores e receptores, em tempo real. Para tal, construiu-se uma plataforma que permite monitorizar toda a actividade de uma rede de *hotspots* que difundem mensagens de publicidade, utilizando tecnologia *Bluetooth*, os quais se encontram espalhados em diferentes locais.

A solução assenta numa plataforma que lê os locais e informações dos *hotspots* a partir de uma base de dados que, em tempo real, recebe informações sobre o estado desses mesmos *hotspots*. Através de uma interface estilo *Web*, o utilizador pode visualizar informação diversa dessa rede, redireccionando o envio de mensagem para diferentes *hotspots* e monitorizar de que forma é que pode rentabilizar a difusão de mensagens, através de antenas emisoras mais ou menos sobrecarregadas.

4.3.1. Funcionamento da API Google Maps

Para poder utilizar iniciar o *Google Maps* é necessário efectuar um registo em <http://code.google.com/intl/pt/apis/maps/signup.html> , sendo atribuída uma chave para cada utilizador, o qual deverá fornecer informações diversas como o servidor onde vai estar alojada a página que contém o código da API, a fim de estar ser monitorizada pelo *Google Maps*. Sendo uma chave restritiva, é necessário efectuar um novo registo caso se pretenda ter mais do que um mapa a funcionar, não sendo possível utilizar uma chave para *localhost* numa página que fique alojada num *Web server*.

Uma vez na posse da chave, é possível utilizá-la e criar o mapa que vamos utilizar, através do seguinte código:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN""http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xmlns:v="urn:schemas-Microsoft-com:vml">
  <head>
    <meta http-equiv="content-type" content="text/html; charset=UTF-8"/>

    <title>Introdução a Google Maps - CriarWeb</title>

    <script src="http://maps.google.com/maps?file=api&v=1&key=INTRODUZIR_CHAVE"
type="text/javascript"></script>
  </head>
```

```

<body>
  <div id="map" style="width: 400px; height: 300px"></div>
  <script type="text/javascript">

    var map = new GMap(document.getElementById("map"));
    map.setMapType(G_SATELLITE_TYPE);
    map.addControl(new GLargeMapControl());
    map.addControl(new GMapTypeControl());
    map.centerAndZoom(new GPoint (-3.688788414001465, 40.41996541363825), 3);

  </script>
</body>
</html>

```

Analisando uma a uma as diferentes variáveis e linhas de comando, é possível concluir que existem diferentes partes de código, cada uma com uma função específica, e que são:

- `var map = new GMap(document.getElementById("map"));`

Esta condição vai gerar o mapa, especificando o destino pretendido como sendo o `id="map"`. O mapa gerado vai ainda adaptar-se ao tamanho que foi especificado para este “map” anteriormente.

- `map.setMapType(G_SATELLITE_TYPE);`

Esta instrução vai permitir especificar o tipo de mapa que queremos visualizar. Existem três tipos – mapa vectorial, satélite e híbrido – sendo assim possível especificar que tipo de informação se pretende que seja mostrada. No caso exemplificado, encontra-se em utilização a vista de satélite (fotografia aérea).

- `map.addControl(new GLargeMapControl());`
`map.addControl(new GMapTypeControl());`

Estas duas instruções vão permitir incorporar elementos para controlar o mapa, tal como *zoom* e tipo de mapa. Podem ainda ser incorporadas outras funcionalidades, como é o caso do *GSmallMapControl*, caso se pretenda reduzir a versão do mapa.

- `map.centerAndZoom(new GPoint (-3.688788414001465, 40.41996541363825), 3);`

Esta instrução permitirá centrar e dimensionar o mapa, para que este fique localizado numa determinada latitude e longitude. Os parâmetros vão gerar um *Gpoint* que mostrará um ponto com uma coordenada de latitude e uma de longitude, além do nível de *zoom* pretendido (no caso específico, é igual a 3).

4.4. Estrutura de implementação da plataforma informática

A solução apresentada no ponto 4.3. comporta diversas vantagens, uma vez que a utilização do código necessário é gratuita, as API's do *Google Maps* são fornecidas em linguagem PHP, é permitida a utilização e representação de diversos pontos de referência recorrendo a uma base de dados e, além disso, o PHP é uma linguagem que é suportada por servidores de forma transparente, facilitando assim a integração entre o *Google Maps*, a georeferenciação e as páginas *Web*.

A plataforma informática desenvolvida apresenta-se estruturada segundo uma lógica de um *website*, uma vez que esse é o seu objectivo final, pese embora o facto de poder funcionar em ambiente local (*localhost*). Além disso, permite o desenvolvimento por módulos, uma vez que basta reformular parte dos menus para incorporar e aceder automaticamente a novas funções.

Para que cumpra as suas funções, são necessários dois grupos de ficheiros: os que permitem gerar toda a estrutura e operações da plataforma e os que são específicos para a georeferenciação dos *hotspots* e que foram importados e trabalhados a partir do *Google Maps*. Existem ainda outros ficheiros que têm como objectivo a formatação da plataforma e a ligação à base de dados. Estes ficheiros são:

- **Ficheiros comuns à plataforma, com objectivo de executar as funções de um site:**
 - `index.php`
 - `login.php`
 - `logout.php`
 - `registo.php`
 - `registo_correcto.php`
 - `lista_antenas.php`

- **Ficheiros utilizados especificamente para a georeferenciação:**
 - *phpsqlsearch_dbinfo.php*
 - *phpsqlsearch_genxml.php*
 - *phpsqlsearch_map.php*

- **Ficheiros utilizados para a ligação à base de dados (ficheiros que se encontram na pasta “Connections”):**
 - *Ligacao_gms.php*

- **Ficheiros utilizados para a formatação e estilos da plataforma (ficheiros que se encontram na pasta “css”)**
 - *Avisos.css*
 - *Links.css*
 - *Navegacao.css*
 - *Texto.css*
 - *Titulos.css*

Esquemáticamente, temos:

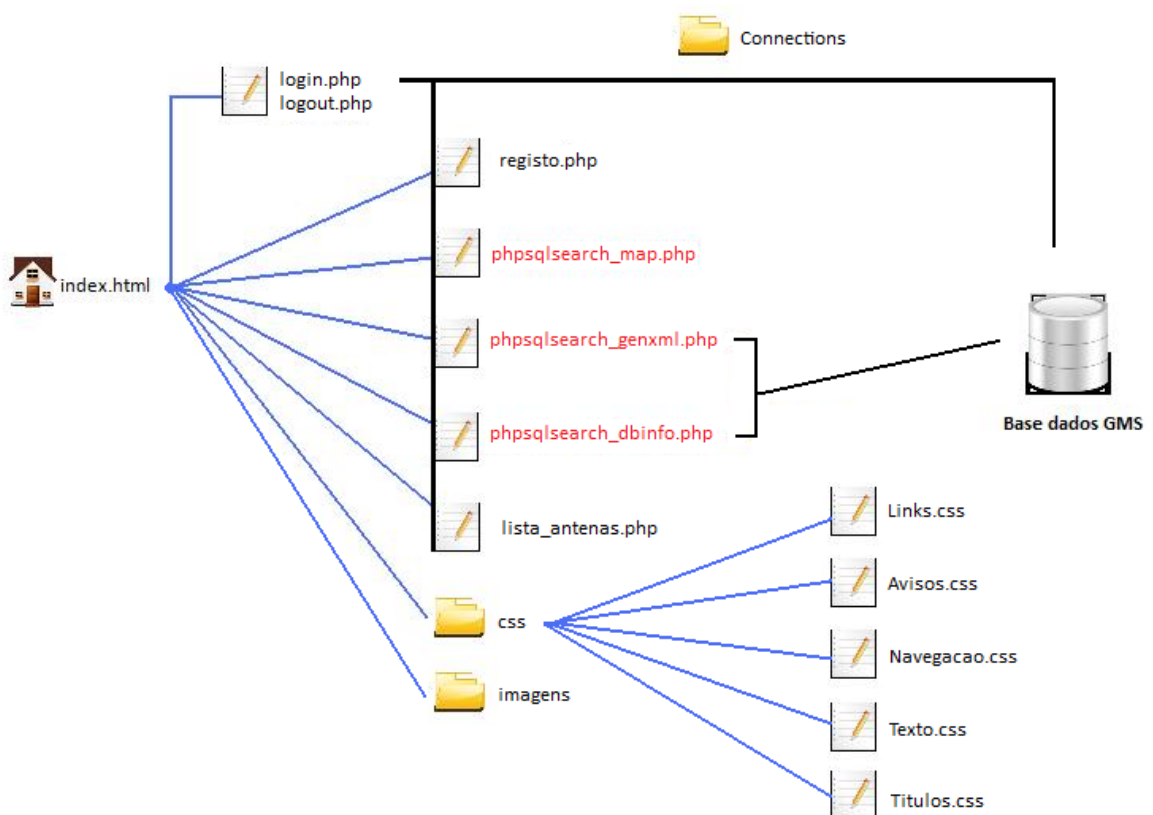


Figura 13 - Estrutura simplificada da plataforma informática

O ficheiro *index.php* é aquele que dá início ao arranque do site e que permite entrar na *homepage* do mesmo, criando a interface para o utilizador, a qual mantém o seu aspecto e estrutura ao longo de toda a navegação.

Nesta plataforma, só é permitida a visualização de mapas a utilizadores registados e autorizados, razão pela qual é necessário efectuar um *login*, sendo para tal chamado o ficheiro *login.php* que executa a ligação à base de dados e verifica se o utilizador tem acesso à plataforma. Caso contrário, é mostrada uma informação de erro e o utilizador é alertado para os eventuais erros de digitação ou para a falta de credenciais de acesso. Assim, poderá repetir a tentativa de entrada ou, em alternativa, proceder a um registo de utilizador.

No caso de o utilizador estar autorizado, é feita a sua autenticação e ficam disponíveis diversas operações, que vão desde a consulta por georeferenciação dos *hotspots* activos, até à visualização de todas as antenas disponíveis, bem como a execução de outras tarefas. No caso da georeferenciação, são utilizados os ficheiros assinalados a vermelho na figura anterior - *phpsqlsearch_dbinfo.php* , *phpsqlsearch_genxml.php* e *phpsqlsearch_map.php* , enquanto que para a listagem das antenas disponíveis é utilizado o ficheiro *lista_antenas.php*.

Caso o utilizador pretenda sair da plataforma informática, poderá fazer o *logout*, que é executado pelo ficheiro *logout.php*. O código de todos os ficheiros referidos anteriormente encontra-se no anexo B deste documento.

Para implementar a estrutura anterior, é necessário apenas um computador que suporte MySQL e PHP, de forma a podermos integrar a base de dados criada com a tecnologia do *Google Maps*. O código é fornecido gratuitamente pelo *Google Maps*, estando disponível para utilização na respectiva página *web*.

Após o registo feito no *Google Maps*, é-nos fornecida uma chave para integrar na API que deverá ser introduzida no código de programação, de forma a podermos aceder aos mapas que necessitarmos.

Para representar os respectivos *hotspots*, esta solução assenta na integração de uma base de dados onde são registados todos os dados referentes aos *hotspots* com a tecnologia fornecida pelo *Google Maps*. O princípio desta representação faz-se marcando num mapa os *hotspots* activos, de forma a monitorizar aqueles que são mais utilizados, os que difundem mais mensagens e se estes podem ser utilizados por diferentes campanhas de publicidade.

Para tal, é gerado um ficheiro XML que cria sinalizações sobre todos os locais pretendidos, a partir de um ponto principal. Após isso, a API interpreta esses locais e marca-os num mapa, através de ícones, os quais representam a informação das coordenadas de latitude e longitude inseridas na base de dados.

Assim que um *hotspot* está activo, esta informação é representada em tempo real e é mostrada ao utilizador. A figura seguinte apresenta um caso prático de aplicação desta tecnologia.



Figura 14 - Exemplo de visualização de georeferenciação

4.4.1. Discussão da estrutura e funcionamento dos ficheiros necessários ao processo de georeferenciação dos *hotspots*

Dada a relevância que os ficheiros *phpsqlsearch_dbinfo.php* , *phpsqlsearch_genxml.php* e *phpsqlsearch_map.php* possuem no funcionamento da plataforma informática, serão aqui objecto de análise pois são os que possuem parte do código importado da API do *Google Maps*, tendo sido posteriormente modificado para satisfazer as formatações e condições da base de dados GMS. Permitem georeferenciar e mostrar os *hotspots* existentes na base de dados, conforme parâmetros de distância entre eles. Os restantes ficheiros possuem código necessário para a visualização das respectivas informações, como é o caso do ficheiro *lista_antenas.php* que mostra uma lista de antenas, sua localização e o seu estado num determinado momento.

No que se refere especificamente aos 3 ficheiros de georeferenciação, após ser estabelecida a ligação à base de dados, utilizando o ficheiro *phpsqlsearch_dbinfo.php*, é utilizada a função DOM (*Document Object Model*) de PHP para gerar o ficheiro XML, função executada pelo ficheiro *phpsqlsearch_genxml.php*.

Conforme os valores fornecidos para a latitude e longitude do ponto inicial, este ficheiro vai calcular a distância a que os restantes pontos que se encontram na base de dados distam do ponto inicial, criando sinalizações ("*markers*"). A partir daqui, faz iterações para todas as distâncias possíveis e que satisfaçam os parâmetros pretendidos.

Uma das alterações introduzidas no código deste ficheiro prendeu-se com a conversão da distância para quilómetros e para metros, uma vez que o ficheiro original continha a conversão para milhas. Esta conversão é feita pela seguinte parte do código:

```
$query = sprintf("SELECT btScannerStartAddress, btScannerStartName, btScannerStartLatitude, btScannerStartLongitude, ( 6371 * acos( cos( radians('%s') ) * cos( radians( btScannerStartLatitude ) ) * cos( radians( btScannerStartLongitude ) - radians('%s') ) + sin( radians('%s') ) * sin( radians( btScannerStartLatitude ) ) ) ) AS distance FROM btScannerStart HAVING distance < '%s' ORDER BY distance LIMIT 0 , 20",
```

O código completo do ficheiro utilizado é o apresentado de seguida:

```
<?php
// Inicia a ligação a base de dados, utilizando o ficheiro referido
include("phpsqlsearch_dbinfo.php");
```

```

// Verificar os parâmetros a partir do URL
$center_lat = $_GET["btScannerStartLatitude"];
$center_lng = $_GET["btScannerStartLongitude"];
$radius = $_GET["radius"];

// Inicia o ficheiro XML e cria os "parent node"
$dom = new DOMDocument("1.0");
$node = $dom->createElement("btScannerStart");
$parnode = $dom->appendChild($node);

// Inicia a ligação ao mySQL server
$conection=mysql_connect (localhost, $username, $password);
if (!$conection) {
    die("Not connected : " . mysql_error());
}

// Activa a ligação ao mySQL database
$db_selected = mysql_select_db($database, $conection);
if (!$db_selected) {
    die ("Can't use db : " . mysql_error());
}

// Procura as linhas para verificar os marcadores a incluir
$query = sprintf("SELECT btScannerStartAddress, btScannerStartName, btScannerStartLatitude,
btScannerStartLongitude, ( 6371 * acos( cos( radians('%s') ) * cos( radians( btScannerStartLatitude ) ) * cos( radians(
btScannerStartLongitude ) - radians('%s') ) + sin( radians('%s') ) * sin( radians( btScannerStartLatitude ) ) ) ) AS
distance FROM btScannerStart HAVING distance < '%s' ORDER BY distance LIMIT 0 , 20",
mysql_real_escape_string($center_lat),
mysql_real_escape_string($center_lng),
mysql_real_escape_string($center_lat),
mysql_real_escape_string($radius));
$result = mysql_query($query);

if (!$result) {
    die("Invalid query: " . mysql_error());
}

header("Content-type: text/xml");

// Faz as iterações sobre as linhas, adicionando os "XML nodes" para cada linha
while ($row = @mysql_fetch_assoc($result)){
    $node = $dom->createElement("marker");
    $newnode = $parnode->appendChild($node);
    $newnode->setAttribute("btScannerStartName", $row['btScannerStartName']);
    $newnode->setAttribute("btScannerStartAddress", $row['btScannerStartAddress']);
    $newnode->setAttribute("btScannerStartLatitude", $row['btScannerStartLatitude']);
    $newnode->setAttribute("btScannerStartLongitude", $row['btScannerStartLongitude']);
    $newnode->setAttribute("distance", $row['distance']);
}

echo $dom->saveXML();
?>

```

No final de todas as iterações feitas, é gerado temporariamente um ficheiro XML que vai ser analisado pelo ficheiro *phpsqlsearch_map.php* que, lendo os dados deste ficheiro, vai conseguir apresentar os *hotspots* no ecrã do utilizador.

No caso específico deste trabalho, o código do ficheiro XML contém os dados completos que vão ser representados no mapa, lidos a partir da base de dados. O ficheiro XML apresenta a seguinte estrutura.

```
<?xml version="1.0" ?>

<btScannerStart>

  <marker btScannerStartName="Torre de Belem" btScannerStartAddress="Lisboa"
    btScannerStartLatitude="38.696487" btScannerStartLongitude="-9.207398" distance="4404.3221684651" />

  <marker btScannerStartName="Universidade de Aveiro" btScannerStartAddress="Campus Universitario de
    Santiago, Aveiro" btScannerStartLatitude="40.631325" btScannerStartLongitude="-8.657495"
    distance="4601.9560740066" />

  <marker btScannerStartName="Centro Comercial" btScannerStartAddress="Gaia"
    btScannerStartLatitude="40.910271" btScannerStartLongitude="-8.485308" distance="4628.9177791173" />

  <marker btScannerStartName="Siloauto" btScannerStartAddress="Avenida dos Aliados, Porto"
    btScannerStartLatitude="41.147976" btScannerStartLongitude="-8.610862" distance="4657.0332769146" />

  <marker btScannerStartName="Universidade do Minho" btScannerStartAddress="Largo do Paco, Braga"
    btScannerStartLatitude="41.550541" btScannerStartLongitude="-8.456542" distance="4697.8313830402" />

</btScannerStart>
```

A partir do momento em que este ficheiro está formado, é possível iniciar o processo de apresentação do mapa pretendido. Para tal, é executado o ficheiro *phpsqlsearch_map.php*, com código semelhante ao referido no ponto 4.3.1, mas com adaptações e complementos para dar resposta às necessidades da plataforma informática. Neste ficheiro, é indicado o local a partir do qual se pretende iniciar a busca e também é fornecida a distância entre este local e os pontos que se querem que sejam apresentados.

O processo de busca das localizações próximas, a partir de um dado endereço, é realizada pela seguinte parte do código:

```
function searchLocations() {
  var btScannerStartAddress = document.getElementById('addressInput').value;
  geocoder.getLatLng(btScannerStartAddress, function(btScannerStartLatitudebtScannerStartLongitude) {
    if (!btScannerStartLatitudebtScannerStartLongitude) {
      alert(btScannerStartAddress + ' Endereço inválido!');
    } else {
      searchLocationsNear(btScannerStartLatitudebtScannerStartLongitude);
    }
  });
}
```

Uma vez que a maioria dos utilizadores sabe o seu endereço mas não as suas coordenadas de latitude e longitude, esta parte do código tem como função analisar os parâmetros de associar ambos. Após isso, é enviada a localização em termos de latitude e longitude para a função seguinte (*searchLocationsNear*), a qual procura os pontos mais próximos a partir da distância pretendida.

Se forem encontradas localizações que se encontram dentro do raio de acção pretendido, estes são temporariamente armazenados e a iteração prossegue até que sejam analisados e encontrados todos os pontos constantes na base de dados. Para cada ponto são armazenadas informações como o nome, endereço, distância e coordenadas de latitude e longitude. A parte do código que realiza esta operação é a seguinte:

```
function searchLocationsNear(center) {
    var radius = document.getElementById('radiusSelect').value;
    var searchUrl = 'phpsqlsearch_genxml.php?btScannerStartLatitude=' + center.lat() + '&btScannerStartLongitude='
+ center.lng() + '&radius=' + radius;
    GDownloadUrl(searchUrl, function(data) {
        var xml = GXml.parse(data);
        var markers = xml.getElementsByTagName('marker');
        map.clearOverlays();

        var sidebar = document.getElementById('sidebar');
        sidebar.innerHTML = '';
        if (markers.length == 0) {
            sidebar.innerHTML = 'Sem resultados!';
            map.setCenter(new GLatLng(40, -8), 5);
            return;
        }

        var bounds = new GLatLngBounds();
        for (var i = 0; i < markers.length; i++) {
            var btScannerStartName = markers[i].getAttribute('btScannerStartName');
            var btScannerStartAddress = markers[i].getAttribute('btScannerStartAddress');
            var distance = parseFloat(markers[i].getAttribute('distance'));
            var point = new GLatLng(parseFloat(markers[i].getAttribute('btScannerStartLatitude')),
                parseFloat(markers[i].getAttribute('btScannerStartLongitude')));

            var marker = createMarker(point, btScannerStartName, btScannerStartAddress);
            map.addOverlay(marker);
            var sidebarEntry = createSidebarEntry(marker, btScannerStartName, btScannerStartAddress,
distance);
            sidebar.appendChild(sidebarEntry);
            bounds.extend(point);
        }
        map.setCenter(bounds.getCenter(), map.getBoundsZoomLevel(bounds));
    });
}
```

A informação gerada vai ser passada para as funções seguintes e que são a função *createmarker* e *createSideBarEntry* que, em conjunto, vão permitir marcar no mapa todos os pontos referenciados. A função *createmarker* tem como objectivo o de assinalar um ícone numa determinada latitude e longitude, além de permitir que seja visualizada informação quando se clica sobre esse ícone. O seu código é:

```
function createMarker(point, btScannerStartName, btScannerStartAddress) {
    var marker = new GMarker(point);
    var html = '<b>' + btScannerStartName + '</b> <br/>' + btScannerStartAddress;
    GEvent.addListener(marker, 'click', function() {
        marker.openInfoWindowHtml(html);
    });
    return marker;
}
```

Quanto à função *createSideBarEntry*, tem como objectivo a criação de uma barra lateral onde é mostrada informação relativa ao local, podendo ser formatada e personalizada, já que se trata de uma *div* idêntica à utilizada num *website* comum. O código desta função é o seguinte:

```
function createSidebarEntry(marker, btScannerStartName, btScannerStartAddress, distance) {
    var div = document.createElement('div');
    var html = '<b>' + btScannerStartName + '</b> (' + distance.toFixed(1) + ')<br/>' + btScannerStartAddress;
    div.innerHTML = html;
    div.style.cursor = 'pointer';
    div.style.marginBottom = '5px';
    div.style.backgroundColor = '#eee';
    GEvent.addDomListener(div, 'click', function() {
        GEvent.trigger(marker, 'click');
    });
    GEvent.addDomListener(div, 'mouseover', function() {
        div.style.backgroundColor = '#eee';
    });
    GEvent.addDomListener(div, 'mouseout', function() {
        div.style.backgroundColor = '#fff';
    });
    return div;
}
```

O código completo e comentado apresenta-se de seguida:

```
<script type="text/javascript">
    <![CDATA[

// Inicializa o mapa, centrado nas coordenadas (40,-8), com um zoom = 5

        var map;
        var geocoder;

function load() {
    if (GBrowserIsCompatible()) {
        map = new GMap2(document.getElementById('map'));
```

```

        map.addControl(new GSmallMapControl());
        map.addControl(new GMapTypeControl());
        map.setCenter(new GLatLng(40, -8), 5);
            geocoder = new GClientGeocoder();
        }
    }
// Procura a localizacao em latitude e longitude, a partir do endereco

function searchLocations() {
    var btScannerStartAddress = document.getElementById('addressInput').value;
        geocoder.getLatLng(btScannerStartAddress, function(btScannerStartLatitudebtScannerStartLongitude) {
    if (!btScannerStartLatitudebtScannerStartLongitude) {
        alert(btScannerStartAddress + ' Endereço inválido!');
    } else {
        searchLocationsNear(btScannerStartLatitudebtScannerStartLongitude);
    }
    });
}

// Procura as localizações próximas do ponto pretendido

function searchLocationsNear(center) {
    var radius = document.getElementById('radiusSelect').value;
    var searchUrl = 'phpsqlsearch_genxml.php?btScannerStartLatitude=' + center.lat() + '&btScannerStartLongitude='
+ center.lng() + '&radius=' + radius;
    GDownloadUrl(searchUrl, function(data) {
        var xml = GXml.parse(data);
        var markers = xml.getElementsByTagName('marker');
        map.clearOverlays();

        var sidebar = document.getElementById('sidebar');
        sidebar.innerHTML = "";
        if (markers.length == 0) {
            sidebar.innerHTML = 'Sem resultados!';
            map.setCenter(new GLatLng(40, -8), 5);
            return;
        }

        var bounds = new GLatLngBounds();
        for (var i = 0; i < markers.length; i++) {
            var btScannerStartName = markers[i].getAttribute('btScannerStartName');
            var btScannerStartAddress = markers[i].getAttribute('btScannerStartAddress');
            var distance = parseFloat(markers[i].getAttribute('distance'));
            var point = new GLatLng(parseFloat(markers[i].getAttribute('btScannerStartLatitude')),
                parseFloat(markers[i].getAttribute('btScannerStartLongitude')));

            var marker = createMarker(point, btScannerStartName, btScannerStartAddress);
            map.addOverlay(marker);
                var sidebarEntry = createSidebarEntry(marker, btScannerStartName, btScannerStartAddress,
distance);
            sidebar.appendChild(sidebarEntry);
            bounds.extend(point);
        }
        map.setCenter(bounds.getCenter(), map.getBoundsZoomLevel(bounds));
    });
}

// Mostra a janela de informações no ícone do hotspot

```

```

function createMarker(point, btScannerStartName, btScannerStartAddress) {
    var marker = new GMarker(point);
    var html = '<b>' + btScannerStartName + '</b> <br/>' + btScannerStartAddress;
    GEvent.addListener(marker, 'click', function() {
        marker.openInfoWindowHtml(html);
    });
    return marker;
}

// Cria a barra lateral e mostra as informações sobre cada hotspot

function createSidebarEntry(marker, btScannerStartName, btScannerStartAddress, distance) {
    var div = document.createElement('div');
    var html = '<b>' + btScannerStartName + '</b> (' + distance.toFixed(1) + ')<br/>' + btScannerStartAddress;
    div.innerHTML = html;
    div.style.cursor = 'pointer';
    div.style.marginBottom = '5px';
    div.style.backgroundColor = '#eee';
    GEvent.addDomListener(div, 'click', function() {
        GEvent.trigger(marker, 'click');
    });
    GEvent.addDomListener(div, 'mouseover', function() {
        div.style.backgroundColor = '#eee';
    });
    GEvent.addDomListener(div, 'mouseout', function() {
        div.style.backgroundColor = '#fff';
    });
    return div;
}

//]]>
</script>

```

O código apresentado diz respeito apenas ao processo de elaboração e representação do mapa. Contudo, e tal como já foi referido, este foi alterado e adaptado para a elaboração e apresentação do fim pretendido, encontrando-se o código completo do ficheiro *phpsqlsearch_map.php* no anexo B deste documento.

4.5. Implementação, testes e usabilidade do sistema informático

4.5.1. Implementação e testes em ambiente real

Para se ter uma noção real do desenvolvimento e funcionamento do projecto, o mesmo tinha de ser trabalhado em ambiente real. Uma vez que foi elaborado um protótipo e se pretendia saber de que forma este protótipo era funcional, o mesmo foi alojado num servidor *Linux*, com ambiente PHP e com servidor de base de dados do tipo MySQL.

Uma vez que em Portugal o serviço de alojamento para um sistema deste tipo teria de ser pago, optou-se por usar um serviço gratuito num servidor estrangeiro (neste caso, dos Estados Unidos da América), com as configurações pretendidas, ficando o mesmo num subdomínio.

Inicialmente, o site foi alojado no servidor *X10hosting* (<http://www.x10hosting.com>), estando disponível no endereço <http://mestrado.x10hosting.com/gms> e, desta forma, foi possível testar diversas variáveis de funcionamento da plataforma, como sejam o tempo de carregamento de uma página, o tempo de acesso aos dados que se encontravam na base de dados (leitura e escrita), a apresentação no *browser*, entre outros.

Dos diversos testes efectuados, apenas um desses testes falhou devido às configurações do próprio servidor que não suportava ambiente DOM/XML com linguagem PHP. Uma vez que é necessário gerar um ficheiro XML que tem de ser lido por outro ficheiro em PHP (tal como descrito no ponto 4.4.1) e, atendendo a que o ficheiro *phpinfo.php* não se encontrava disponível para analisar, foi contactado o administrador de sistema sobre esta situação, facto que até à data não teve resposta.

No entanto, e perante esta dificuldade, o site foi também alojado num outro servidor gratuito, de forma a compreender se tal dificuldade se encontrava no servidor ou se seria alguma função de PHP não instalada. O novo alojamento, efectuado em <http://mestrado.byethost10.com/gms> revelou que, provavelmente, a dificuldade se deveu às configurações próprias do servidor *X10hosting*.

A tabela seguinte apresenta as principais características de cada um dos servidores:

Características	Servidor X10hosting	Servidor Byethos
Espaço disponível	500 Mb	5000 Mb
Publicidade (<i>ad-ons</i>)	Não	Não
<i>Scripting</i>	CGI, PHP, ASP.NET, SSI, FrontPage Extensions, Perl, Python	PHP
<i>Upload</i>	FTP, Browser	FTP, Browser
MySQL	Sim	Sim
PHP 5.0 (ou superior)	Sim	Sim

Tabela 5 - Quadro resumo dos servidores de alojamento utilizados

Uma vez que a aplicação se encontrava agora alojada num novo servidor, foram testados diversos parâmetros para os dois casos, de forma a comparar a *performance* da aplicação em dois ambientes similares. Entre estes parâmetros incluem-se os já referidos anteriormente e que revelaram estar ao mesmo nível, donde se destaca:

- Todas as páginas foram carregadas em menos de dez segundos;
- Todos os itens foram visualizados;
- O ficheiro em XML foi gerado automaticamente em ambos os casos, mas ligeiramente mais rápido no servidor *Byethost*;
- Falha na leitura do ficheiro XML a partir do ficheiro *phpsqlsearch_map.php*;
- Acesso rápido e sem falhas ao mapa (*Google Maps*);
- Facilidade de criação e manipulação da base de dados.

Assim, de uma forma geral, concluiu-se que o alojamento é muito idêntico e que a rapidez com que são gerados os dados para o sistema informático desenvolvido é bastante semelhante, pese embora as diferenças possíveis entre os dois servidores, que se situam mais ao nível da oferta das funcionalidades do que na rapidez de execução.

4.5.2. Usabilidade do sistema informático

A usabilidade num sistema informático é algo que, devido às suas características, terá de ser medida ao longo do tempo e de acordo com o perfil de utilizadores. Uma vez criado o sistema informático, torna-se imperativo submetê-lo a testes e analisar o comportamento das pessoas que com ele trabalham, independentemente da sua formação ou *background* com os sistemas informáticos.

Ao longo deste trabalho, pretendeu-se seguir algumas regras evidenciadas pela norma ISO-9241-11 (10), tendo em atenção diversos parâmetros, como a consistência das páginas, a normalização dos ecrãs de trabalho, a utilização de um menu padrão ao longo de toda a navegação e a aproximação do protótipo concebido ao modelo final, tal como referido no ponto 2.5. Para tal, foi elaborado um menu de navegação (lado esquerdo do ecrã), onde o utilizador acede à maioria das funções disponibilizadas. A figura seguinte ilustra esta situação.



Figura 15 - Aspecto geral do sistema informático desenvolvido

Contudo, e devido ao carácter específico deste sistema informático, existem situações que não são possíveis de ser controladas, nomeadamente a representação do próprio mapa – uma vez que este é fornecido pelo *Google Maps* – ou a projecção dos ícones sobre este, podendo-se apenas alterar o seu formato ou aspecto gráfico.

Não obstante este facto, optou-se por utilizar a cor preta como fundo de apresentação do sistema informático. Por um lado, devido ao facto de o site da *SpectralBlue* já se encontrar nestas cores e ser necessário manter a padronização inicial; por outro, devido ao facto de se pretender aumentar a possibilidade de contraste, caso se realize uma versão para pessoas com limitações visuais.

Ao longo do desenvolvimento do trabalho, teve-se também em conta as especificações da norma ISO (10), uma vez que se respeitaram os seguintes itens desta norma:

- Clareza no acesso à informação: o utilizador consegue, através do menu localizado no lado esquerdo, aceder a praticamente todas as funcionalidades pretendidas e navegar na aplicação.
- Facilidade de navegação: tal como referido na norma, o utilizador consegue, com apenas três toques de selecção, fazer o seu *login* e aceder à área de representação dos *hotspots*, lista de *hotspots*, entre outros conteúdos. Caso seja necessário registar um utilizador, esse número de passos reduz-se a apenas dois toques de selecção.
- Consistência: ao longo do sistema informático desenvolvido, manteve-se a estrutura das páginas, de forma a que o utilizador “veja” o sistema como um projecto único e integrado.
- Simplificação: neste projecto, pretendeu-se otimizar a rapidez das páginas. Em ambiente real, as páginas demoram menos de dez segundos a serem carregadas.

Apesar destes resultados, não foram realizados outros testes como é o caso da usabilidade para cidadãos com limitações visuais ou físicas, não tendo sido também realizados testes através de métricas como entrevistas ou questionários, devido ao facto de o sistema informático ainda não ter terminado totalmente a fase de prototipagem e desenvolvimento.

5. Conclusão e trabalhos futuros

5.1. Conclusões

No que se refere ao trabalho teórico, este implicou uma pesquisa contínua, uma vez que foca 4 áreas distintas: marketing de proximidade, sistema *Bluetooth*, tecnologia *Google Maps* e integração das diferentes aplicações ao nível do desenvolvimento informático.

Isto obrigou a uma permanente actualização e atenção às novidades tecnológicas, não só no campo dos sistemas *Bluetooth*, sobretudo com o lançamento das especificações da versão 3.0, mas também nas inovações da tecnologia *Google Maps* e no lançamento de sistemas móveis com plataformas compatíveis (por exemplo, tecnologia *Android*).

Aliás, uma das maiores dificuldades sentida prendeu-se precisamente com a rápida e constante mudança da tecnologia informática, assistindo-se a várias alterações cruciais ao longo do processo de desenvolvimento da parte prática, das quais se destacam:

- Alterações no sistema do Google Maps, donde se salienta o lançamento do *Street View* e a difusão do sistema *Android* (actualmente presente já em telemóveis de operadoras nacionais);
- Desenvolvimento e lançamento da versão 3.0 da especificações *Bluetooth*, que irá substituir a versão 2.0 actualmente utilizada pelos dispositivos existentes;
- *Upgrade* da versão 2 para a versão 3 da API do Google Maps, utilizada para a marcação dos *hotspots* e para a representação iconográfica dos mesmos;
- *Upgrade* às versões de PHP e MySQL utilizadas inicialmente, quer ao nível de desenvolvimento local, quer ao nível de servidores de alojamento.

Contudo, apesar do tempo que foi necessário para realizar este trabalho teórico, constatou-se que o desenvolvimento de soluções tecnológicas específicas para georeferenciação de sistemas *Bluetooth*, aliados ao marketing de proximidade, continua inexistente. As soluções apresentadas até ao momento são, sobretudo, as que nos permitem georeferenciar pontos de forma isolada, como acontece actualmente com alguns sites de empresas, sobretudo para divulgar a sua localização junto dos seus clientes.

Apesar de existirem inúmeras aplicações de representação de informação geográfica e desta já ser bastante utilizada, a georeferenciação de sistemas *Bluetooth* é muito pouco explorada, talvez fruto do não desenvolvimento deste tipo de sistemas por parte de empresas dedicadas à publicidade de proximidade.

Mesmo com os avanços que a tecnologia *Bluetooth* tem evidenciado, a difusão de publicidade de proximidade ainda não atingiu a maturidade desejada, não sendo actualmente um forte meio de difusão de informação e publicidade. Pese embora existirem diferentes ofertas no mercado, Portugal é um dos casos onde ainda não é um meio de transmissão de publicidade amplamente difundido, sobretudo pelo preço e pelas soluções apresentadas. Esta situação não deixa de ser curiosa e contrastar com a elevada taxa de penetração e utilização de telemóveis no nosso país, com praticamente todos os aparelhos equipados actualmente com tecnologia *Bluetooth*.

Um outro aspecto a referir é a pouca oferta de soluções no mercado em termos de georeferenciação de sistemas deste tipo, os quais não promovem a ampla difusão destes serviços. Isto poderá indicar que existe um nicho de mercado a explorar mas também que terá de ser um tipo de serviço atractivo, de baixo custo e de fácil implementação. Uma vez que o sistema desenvolvido foi realizado em linguagem *open-source* e a sua implementação é de menor custo que a de um comum *website*, esta solução poderá ser rapidamente adoptada pelas empresas que pretendam monitorizar os seus *hotspots*, sejam eles *Bluetooth* ou não.

Em termos de trabalho informático realizado, concluiu-se que a tecnologia *Google Maps* ainda apresenta algumas limitações. Após povoamento da base de dados, foi possível verificar que alguns dos locais não se encontram georeferenciados devido a limitações do próprio sistema, como é o caso da ilha da Madeira ou de outros locais de menores dimensões. Outros locais não permitem um *zoom* total até ao nível da vista de rua, apesar de já ser possível utilizar a tecnologia *Street View* em diversas localidades.

Contudo, foram desenvolvidas as interfaces que permitem visualizar os *hotspots* que se encontrem em funcionamento, bem como toda a estrutura do sistema informático. No entanto, a tecnologia *Google Maps* propõe o uso de ficheiros XML e PHP, através da utilização de *Document Object Model* (DOM), que permite ler o conteúdo do ficheiro gerado em XML, usando uma aplicação desenvolvida em PHP.

Esta tecnologia revelou-se, em algumas situações, um pouco difícil de trabalhar, uma vez que nem todas as versões de PHP suportam DOM.

Ainda assim, e após o *upgrade* da linguagem PHP para a versão 5.0, foi possível construir uma aplicação estável que permitiu a georeferenciação e monitorização de uma rede de *hotspots*, de forma eficaz e que permite analisar, em tempo real, a difusão da publicidade de proximidade. Além dessa monitorização, este sistema permite analisar o estado da rede e redireccionar o envio de mensagens através de *hotspots* localizados nas proximidades, quando o *hotspot* inicialmente previsto para a tarefa tiver uma falha.

Por outro lado, o facto de se ter de usar *Javascript* com PHP, XML e MySQL, requereu algum trabalho extra, uma vez que com o lançamento das novas versões do *Internet Explorer* 8.0 verificaram-se algumas incompatibilidades, geradas sobretudo devido a questões de segurança e que bloqueavam o funcionamento da plataforma desenvolvida.

Concluiu-se também que a utilização da tecnologia PHP e MySQL foi uma boa aposta pois, além da compatibilidade com os ficheiros disponibilizados pelo *Google Maps* para a representação de locais, torna-se possível construir pequenos módulos conforme as necessidades, o que é uma mais-valia pois permite continuar a desenvolver este trabalho e poder adaptá-lo para outras situações (por exemplo, georeferenciação de câmaras de vídeo, localização de painéis publicitários, etc).

No entanto, importa ressaltar que este trabalho não se encontra totalmente concluído, uma vez que o desenvolvimento modular do sistema informático pode, a qualquer altura, ser provido de mais funcionalidades e funções. Além disso, a incorporação de novas tecnologias poderão trazer, no futuro, maiores possibilidades ao nível de funções a desenvolver. Importa contudo referir que o desenvolvimento deste núcleo de componentes é o factor primordial para a utilização e compreensão de um sistema deste tipo, podendo-se rapidamente incrementar o seu potencial continuamente ao longo do tempo.

5.2. Trabalhos futuros

Actualmente encontra-se desenvolvida a parte de georeferenciação dos *hotspots* activos na rede, bem como a sua representação num mapa, o qual permite a vista de terreno e a vista de satélite.

Em termos de trabalho presente, está sobretudo a desenvolver-se e a testar-se a usabilidade, de forma a poder melhorar o sistema criado, quer ao nível da apresentação, quer ao nível do funcionamento da mesma. Uma outra melhoria que se está actualmente a desenvolver tem a ver com a possível integração desta plataforma para aplicação em sistemas que não apenas as redes formadas por sistemas *Bluetooth*, mas também em aplicações diversas como sistemas de vigilância por vídeo, ou outros onde se possa tirar partido da distribuição geográfica.

A integração do sistema para localização de *hotspots Bluetooth* ainda não é, actualmente, uma tecnologia bastante desenvolvida para ser utilizada em tempo real, uma vez que seria necessário ter um sistema que gerasse dados em tempo real para poder testar a aplicação desenvolvida. Assim, a solução de georeferenciação apresenta, essencialmente, informação de forma estática.

Contudo, com o desenvolvimento deste sistema, torna-se possível georeferenciar qualquer objecto ou recurso em tempo real, desde que o mesmo seja capaz de enviar informações sobre o seu estado para uma base de dados, abrindo assim a possibilidade para a monitorização em tempo real de embebidos diversos.

Para trabalho futuro, sugere-se também a integração da versão 3.0 da API do *Google Maps* que, segundo os próprios autores, será mais rápida de executar, com um código mais simplificado e com menos incompatibilidades ao nível de *browser*. Além disso, esta nova versão permitirá a consulta de mapas em tempo real a partir de um dispositivo móvel (por exemplo, um PDA), ampliando assim o espectro de utilização desta plataforma, mesmo para sistemas não equipados com tecnologia *Android*.

Também como possíveis trabalhos futuros, poder-se-ia sugerir uma série de módulos a desenvolver com base no trabalho já realizado, como sejam:

- o desenvolvimento de *hotspots* que se “auto-identifiquem”, enviando a sua localização directamente para a base de dados;
- a criação de uma ferramenta que incorpore a tecnologia *Street View* para a visualização dos *hotspots* com imagens 3 D;
- a reprogramação da aplicação para tecnologia WAP;
- a introdução da localização de *hotspots* em tempo real, utilizando um dispositivo móvel;
- introdução da função *reverse geocoding* na aplicação já desenvolvida;
- a possibilidade de acesso à informação via *browser* instalado num PDA, utilizando a versão 3.0 da API *Google Maps*.

Com base nas ideias que foram surgindo, foram feitos alguns contactos com empresas que trabalham no ramo da difusão de mensagens multimédia, tendo a maioria mostrado receptividade à adopção de uma plataforma de georeferenciação que permitisse monitorizar, em tempo real, o estado dos seus *hotspots*.

Assim, este poderá ser um mercado a explorar, ampliando-se o número de funções e as possibilidades que a georeferenciação e monitorização de *hotspots Bluetooth* oferece actualmente.

6. Bibliografia

6.1. Referências específicas

1. **Matos, Paulo Pombinho, Carmo, Maria Beatriz e Afonso, Ana Paula.** Visualização de informação georeferenciada em dispositivos móveis. *Visualização de informação georeferenciada em dispositivos móveis*. Lisboa : Departamento de Informática, Faculdade de Ciências da Universidade de Lisboa, 2004.
2. **Cassoli, Dajana, et al.** The Bluetooth Technology: State of the Art and Networking Aspects . *Book Series Lecture Notes in Computer Science* . University of Rome : SpringerLink/Heidelberg, 2004 . Vol. 2345/2002 , ETWORKING 2002. Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; and Mobile and Wireless Communications. ISSN 0302-9743 (Print) 1611-3349 (Online).
3. **CNSS - Pilha** protocolar Bluetooth. [Online] <http://cnss.files.wordpress.com/2007/09/bt2.jpg>
4. **Insight, Emerald.** [Online] <http://www.emeraldinsight.com/fig/2630230505001.png> .
5. **MMA.** MMA Code of Conduct. [Online] <http://mmaglobal.com/codeofconduct.pdf>.
6. **SIG, Bluetooth.** SIG-Bluetooth. [Online] <http://www.bluetoothsig.org>.
7. **Haase & Martin GmbH - Neue Medien Dresden, Germany (Member of Bluetooth SIG).** Methods of Bluetooth® Marketing. 2008.
8. **Buttery, S. e Sago, A.** Future applications of Bluetooth. *BT Technology Journal*. Netherlands : Springer, 2004. Vols. Volume 21, Number 3 / July, 2003, Computer Science. ISSN: 1358-3948 (Print) 1573-1995 (Online) - DOI:10.1023/A:1025102930228.
9. **IETF-RFC4627.** [Online] <http://tools.ietf.org/html/rfc4627>.
10. **International Standard Organization.** Norma ISO 9241-11. *Ergonomic requirements for office work with visual display terminals*. s.l. : International Standard Organization, 1998. Vol. First Edition.
11. **Vassilas, Nikolaus, et al.** Intelligent Techniques for Spatio-Temporal Data Analysis in Environmental Applications. *Pag. 318-324*. Attiki, Greece : University of Patras, 1999. Vols. DOI: 10.1007/3-540-44673-7_21. ISBN: 978-3-540-42490-1.

12. **Jenny Preece, Dean.** Online Communities: researching sociability and usability in hard to reach populations. Disponível em: <http://dl.acs.org.au/index.php/ajis/article/viewPDFInterstitial/132/112?ads=> Maryland, USA : University of Maryland, 1994. Online Communities.
13. **Avison, D. e Fitzgerald, G.** Information Systems Development: methodologies, techniques and tools. s.l. : McGraw-Hill, 2003. Vol. 3º volume.
14. **Lopes, Filomena Castro, Morais, Maria Paula e Carvalho, Armando Jorge.** Desenvolvimento de sistemas de informação. *Sistemas de informação*. Lisboa : FCA, 2005. Vol. 1, Sistemas de Informação. ISBN-10: 972-722-461-X.
15. **Maps, Google.** Google Maps API Code. [Online] <http://code.google.com/intl/pt/apis/maps/>.
16. **Bobbie, Patrick O. e Yussiff, Abdul Lateef.** Modeling and simulation of IEEE 802.11 Wireless-Lan and Bluetooth Piconet Range Interference. *Final CIIT Conference Paper*. Georgia (USA): School of Computing and Software Engeneering - Southern Polytechnic State University, 2008. Vol. DOI 10.1.1.83.3221.
17. **Pina, Maria Fátima e Olhero, Andrea.** Georeferenciação de dados em micro-áreas. Desafio à utilização dos SIG na área da saúde. Porto : Instituto de Engenharia Biomédica - INEB, 2005. SIG.

6.2. Outras referências

Damas, Luís, *“SQL - Structured Query Language”*, FCA Editora, 2006, Vol. 1, Tecnologias da Informação. ISBN-10: 972-722-443-1.

DuBois, Paul, *“MySQL Cookbook”*, Editora O’Reilly, 2002, ISBN-10: 0-596-00145-2

Greenspan, Jay e Bulger, Brad, *“MySQL/PHP Database Applications”*, M&T Books, 2001. ISBN-10: 0-7645-3537-4

Kobayashi, Carlos Yassunori, *“A tecnologia bluetooth e suas aplicações”*, Universidade de São Paulo, 2004.

Oliveira, Hélder, *“Curso avançado de Dreamweaver CS3 e 8”*, FCA Editora, 2007, ISBN-13: 978-972-722-558-3

Pina, Maria de Fátima, Andreia Olhero, *“Georeferenciação de dados em Micro-Áreas. Desafio à utilização dos SIG na área da saúde”*, Instituto de Engenharia Biomédica – INEB, 2005.

Remoaldo, Pedro, *“Guia Prático do Dreamweaver”*, CentroAtlântico, 2006, Colecção Tecnologias, ISBN-10: 989-615-027-3

Rogers, Gary S., John Edwards; *“An Introduction to Wireless Technology”* Prentice Hall, 2003.

Serrão, Carlos, Joaquim Marques, *“Programação com PHP 5”*, FCA Editora, 2007, Biblioteca Software Livre. ISBN-13: 978-972-722-549-1.

Tanenbaum A. S.; *“Computer Networks, 4th edition”* Prentice Hall, 2003, ISBN-10: 0-13-066102-3.

7. Anexos

Anexo A – Documentação da base de dados GMS

A base de dados GMS é composta por 10 tabelas, e que são:

Nome da tabela: Btmajorclass	
Nome	Tipo de dados
<u>btMajorClassID</u>	Int(11)
btMajorClassHEX	Varchar(40)
btMajorClassName	Varchar(40)

Nome da tabela: Btminorclass	
Nome	Tipo de dados
<u>btMinorClassID</u>	Int(11)
btMajorClassID	Int(11)
btMajorClassHEX	Varchar(40)
btMajorClassName	Varchar(40)

Nome da tabela: Btmanufacturer	
Nome	Tipo de dados
<u>btManufacturerID</u>	Int(11)
btSuperManufacturerID	Varchar(40)
btManufacturerHEX	Varchar(40)
btManufacturerName	Varchar(40)
btManufacturerAddress	Varchar(40)

Nome da tabela: Btremotedevice	
Nome	Tipo de dados
<u>btRemoteDeviceID</u>	Int(11)
btRemoteDeviceLastname	Varchar(40)
btDeviceClass	Varchar(40)
btMajorClassID	Int(11)
btMinorClassID	Int(11)
btManufacturerID	Int(11)

Nome da tabela: Btscanner	
Nome	Tipo de dados
<u>btScannerID</u>	Int(11)
btScannerLastname	Varchar(40)
btMajorClassID	Int(11)
btMinorClassID	Int(11)
btManufacturerID	Int(11)

Nome da tabela: Btremotedevicefound	
Nome	Tipo de dados
<u>btRemoteDeviceFoundID</u>	Int(11)
btRemoteDeviceID	Int(11)
btScannerID	Int(11)
btRemoteDeviceFoundName	Varchar(40)
btRemoteDeviceFoundServiceClass	Varchar(40)
btRemoteDeviceFoundClockOffset	Varchar(40)
btRemoteDeviceFoundLinkQuality	Varchar(40)
btRemoteDeviceFoundRSSI	Varchar(40)
btRemoteDeviceFoundDate	Varchar(40)
btRemoteDeviceFoundURLtoSend	Varchar(40)
btRemoteDeviceFoundCicleCounter	Varchar(40)
btRemoteDeviceFoundLatitude	Varchar(40)
btRemoteDeviceFoundLongitude	Varchar(40)

Nome da tabela: Btscannerstart	
Nome	Tipo de dados
<u>btScannerStartID</u>	Int(11)
btScannerID	Int(11)
btScannerStartDiscoverableMode	Varchar(40)
btScannerStartName	Varchar(40)
btScannerStartAddress	Varchar(80)
btScannerStartDate	datetime
btScannerStartLatitude	Float(10,6)
btScannerStartLongitude	Float(10,6)
btScannerStartType	Varchar(30)

Nome da tabela: Btsupermanufacturer	
Nome	Tipo de dados
<u>btSuperManufacturerID</u>	Int(11)
btSuperManufacturerName	Varchar(40)

Nome da tabela: Utilizadores	
Nome	Tipo de dados
<u>Cod_utilizador</u>	Int(11)
nome	Varchar(60)
Utilizador	Varchar(20)
Palavra_passe	Varchar(20)
Email	Varchar(80)
Data_registo	Datetime
Activo	Enum(S,N)
Id_nivel	Int(2)

Nome da tabela: Niveis	
Nome	Tipo de dados
<u>Id_nivel</u>	Int(2)
nivel	Varchar(10)

No que diz respeito ao desenho da base de dados, esta é composta por todas as tabelas interligadas entre si. Por ser uma base de dados relacional, foram respeitadas as regras inerentes à criação de bases de dados relacionais, tendo-se ido até à 3ª forma normal. A figura seguinte apresenta o esquema geral da base de dados.

Anexo B – Código completo dos ficheiros da plataforma

- **Ficheiros gerais:**

- Ficheiro *index.php*
- Ficheiro *login.php*
- Ficheiro *logout.php*
- Ficheiro *registo.php*
- Ficheiro *registo_correcto.php*
- Ficheiro *lista_antenas.php*

- **Ficheiros de georeferenciação:**

- Ficheiro *phpsqlsearch_dbinfo.php*
- Ficheiro *phpsqlsearch_genxml.php*
- Ficheiro *phpsqlsearch_map.php*

- **Ficheiro de ligação à base de dados:**

- Ficheiro *Ligacao_gms.php*

- **Folhas de estilo**

- Ficheiro *Avisos.css*
- Ficheiro *Links.css*
- Ficheiro *Navegacao.css*
- Ficheiro *Texto.css*
- Ficheiro *Titulos.css*