

Universidade do Minho

Escola de Engenharia

Olga Sergeevna Ferreira

Utilização oportunista de redes locais sem fios por *place - based systems*.

*Opportunistic use of wireless local area networks by place-based systems.*

Tese de Mestrado

Ciclo de Estudos Integrados Conducentes ao

Grau de Mestre em Engenharia de Comunicações

Trabalho efectuado sob a orientação de:

**Professora Doutora Helena Rodrigues**

**Professor Doutor Adriano Moreira**

**Dezembro 2012**



## AGRADECIMENTOS

Direcciono os meus agradecimentos a todos que contribuíram para a realização desta dissertação de mestrado, principalmente aos orientadores: Professora Doutora Helena Rodrigues e Professor Doutor Adriano Moreira pela disponibilidade manifestada, acompanhamento e conhecimento transmitido.

Em especial aos meus pais, ao meu irmão e a toda a minha família pelo apoio dado, incentivo e motivação para continuar com os estudos.

A todos os meus colegas pela amizade e companheirismo ao longo dos anos de curso.

A todos, muito Obrigada!



## RESUMO

Esta dissertação foca-se na interacção entre as áreas de computação ubíqua e sistemas baseados na localização, proporcionando uma maior flexibilidade e mobilidade aos utilizadores e um maior desenvolvimento na virtualização de espaços físicos.

O principal desafio desta dissertação é a evolução do processo de virtualização do espaço físico, procurando integrar recursos locais com as funcionalidades dos *place-based systems* já existentes. Ao fazer face a este desafio, avaliaram-se modelos de integração resultantes da interacção de novas funcionalidades com serviços e aplicações de suporte aos *place-based systems*, assim como a integração de recursos locais do mundo real em sistemas de socialização no mundo virtual.

**Palavras-chave:** redes locais sem fios, sistemas baseados em places, nomeação dos places, virtualização de espaço, recursos locais.



## ABSTRACT

*This dissertation focuses on the interaction between the areas of ubiquitous computing and location-based systems, providing greater flexibility and mobility to users and further development in the virtualization of physical spaces.*

*The main challenge of this work is the evolution of the process of virtualization of the physical space, trying to integrate local resources with the features of place-based systems that already exist. To tackle this challenge, we evaluated models of integration resulting from the interaction of new features and services to support applications for place-based systems (PBS), as well as the integration of real world local resources with social systems in the virtual world.*

**Key-Words:** *Wi-Fi network, place-based systems, place naming, space virtualization, local resources.*





# CONTEÚDO

<b>Agradecimentos</b> .....	I
<b>Resumo</b> .....	III
<b>Abstract</b> .....	V
<b>Lista de Figuras</b> .....	XI
<b>Lista de Tabelas</b> .....	XII
<b>Acrónimos</b> .....	XV
<b>1 Introdução</b> .....	1
1.1 Enquadramento .....	1
1.2 Objectivos .....	2
1.3 Estrutura do Documento .....	3
<b>2 Trabalho relacionado</b> .....	5
2.1 Place-Based Systems vs Location-Based Systems .....	5
2.1.1 Conceitos .....	7
2.1.2 Foursquare .....	9
2.1.3 Facebook Places .....	11
2.1.4 Locamoda .....	12
2.1.5 Pinpoint.....	12
2.1.6 Análise .....	13
2.2 Tecnologia WI-FI .....	14
2.2.1 Captive portal .....	15
<b>3 Análise</b> .....	19
3.1 Espaço de desenho .....	19
3.1.1 Descrição de actores .....	19
3.1.2 Cenários de criação e utilização dos places .....	20
<b>4 Conceção do sistema</b> .....	27
4.1 Requisitos do sistema.....	27

4.2	Arquitectura do Sistema.....	28
4.2.1	FMP- Interface Web .....	30
4.2.2	Servidor FMP .....	30
4.2.3	Servidor BD FMP .....	31
4.2.4	Modelo de dados.....	31
4.2.5	Descrição das tabelas.....	33
4.3	Protótipo 1.....	36
4.4	Protótipo 2.....	39
4.5	Limitações e Desafios .....	41
4.6	Integração do FMP com um Modelo de Espaço Simbólico. ....	42
4.6.1	Criar o place com o auxílio do modelo de espaço simbólico e respectiva Interface de utilizador.....	43
4.6.2	Representação de um espaço físico em função do modelo de espaço simbólico. ....	44
4.7	Integração do Sistema FindMePlaces com o Sistema EPI.....	46
<b>5</b>	<b>Desenvolvimento e implementação do Sistema FMP.....</b>	<b>49</b>
5.1	Interface Web FMP.....	49
5.1.1	Mapa de navegação .....	49
5.1.2	Interface com o utilizador.....	51
5.2	Servidor FMP.....	57
5.3	Servidor BD FMP .....	58
5.4	Interacção entre o sistema FindMePlaces e Modelo de espaço simbólico. ....	58
5.4.1	Criar um novo place com auxílio do modelo de espaço simbólico. ....	58
5.4.2	Inserir objectos no modelo de espaço simbólico sempre que o utilizador acabe de criar um place. ....	61
5.5	Modo de integração do Sistema FindMePlaces com o Sistema EPI. ....	65
5.5.1	Submissão de assinaturas de rádio ao sistema EPI.....	65

5.6	Testes de integração de funcionalidades do sistema findmeplaces em ambiente real	66
5.6.1	Metodologia de divulgação do sistema FMP .....	66
5.6.2	Estatísticas de monitorização do sistema FMP.....	66
5.6.3	Análise dos Resultados .....	69
6	<b>Conclusões</b> .....	71
	<b>Referências</b> .....	73
	<b>Anexos</b> .....	75
	Anexo 1- Funções EPI (Fonte (16)).....	75
	Anexo 2 - Interface entre FMP e sistema EPI Função f1 (Fonte (16)) .....	77
	Anexo 3 - WilmaGate .....	79



## LISTA DE FIGURAS

Figura 1: Diagrama de sequência processo de autenticação de cliente com Mecanismo de Captive Portal , Fonte (13).....	15
Figura 2: Representação de diferentes privilégios de actores no sistema FindMePlaces. ....	19
Figura 3: Cenários de utilização do Sistema FindMePlaces.....	21
Figura 4: Arquitectura funcional do sistema FindMePlaces. ....	29
Figura 5: O modelo Entidade – Relacionamento da base de dados FindMePlaces.....	32
Figura 6: Funcionamento da API de localização de browser ou W3C.....	37
Figura 7: Protótipo Geográfico.....	39
Figura 8: Funcionamento da API de localização no protótipo geográfico. ....	40
Figura 9: Mapeamento dos Places no protótipo 2. ....	41
Figura 10: Representação de um espaço físico em dimensão de modelo simbólico.....	45
Figura 11: Mapa de navegação e a respectiva legenda.....	50
Figura 12: Interface de login no sistema FindMePlaces.....	51
Figura 13: Registo de um novo utilizador. ....	52
Figura 14: Apresentação da lista dos places. ....	53
Figura 15: Interface para criação de um novo place.....	55
Figura 16: Página do place. ....	56
Figura 17: Interface para partilha de conteúdos e lista de presença. ....	57
Figura 18: Diagrama de sequência: criar um novo place com auxílio de modelo de espaço simbólico. ....	59
Figura 19: Diagrama de sequência: inserção de objectos no modelo de espaço simbólico sempre que o utilizador acabe de criar um place.....	62
Figura 20: Diagrama de sequência: Submissão de assinaturas de rádio ao sistema EPI.....	65
Figura 21: Novos utilizadores registados por semana. ....	67
Figura 22: Novos places registados por semana.....	68
Figura 23: Publicações registadas por semana. ....	69
figura 24: descrição das funções de interface de contextualizer simbólico.....	76
Figura 25: Mensagem enviada para o servidor de FMP a fazer submissão de assinatura rádio.....	77
Figura 26:Arquitectura do sistema Wilmagate.....	80



## LISTA DE TABELAS

Tabela 1: Tabela comparativa de funcionalidades. ....	13
Tabela 2: Tabela places. ....	33
Tabela 3: Tabela Users. ....	34
Tabela 4: Tabela visits. ....	34
Tabela 5: Tabela publications. ....	35
Tabela 6: Tabela aps. ....	36
Tabela 7: Tabela Places/aps. ....	36





## ACRÓNIMOS

Na lista abaixo encontram-se os acrónimos. Sempre que um acrónimo é referenciado pela primeira vez durante a leitura da dissertação é apresentado o seu significado.

**AP** *Access Point*

**GPS** *Global Positioning System*

**IEEE** *Institute of Electrical and Electronics Engineers*

**LBS** *Location-Based System*

**LBSS** *Location-Based Social System*

**MAC** *Media Access Control*

**PBS** *Place- Based System*

**PDA** *Personal Digital Assistant*

**RSSI** *Received Signal Strength Indicator*

**SSID** *Service Station Identifier*

**Wi-Fi** *Wireless Fidelity*

**WLAN** *Wireless Local Area Network*



# 1 INTRODUÇÃO

Neste capítulo apresenta-se o âmbito da dissertação, o seu enquadramento na realidade actual, os objectivos que se pretendem alcançar e como o documento se encontra estruturado.

## 1.1 ENQUADRAMENTO

Actualmente, grande parte da população utiliza dispositivos móveis no seu dia-a-dia (computadores portáteis, telemóveis, PDAs, *SmartPhones*, entre outros). Estes dispositivos móveis possuem diversas interfaces de rádio, tal como a interface *Wi-Fi*.

Na última década observou-se um crescimento exponencial de infra-estruturas de suporte às redes locais sem fios WLAN (*Wireless Local Area Network*), redes “Wireless” ou ainda “*Wireless Fidelity Wi-Fi*”, em locais públicos (universidades, escolas, centros comerciais, cafés, hotéis e aeroportos). Na sequência deste crescimento surgiram áreas designadas como *hotspots Wi-Fi* (1), caracterizadas como áreas públicas onde existe cobertura de rede, permitindo aos utilizadores de dispositivos móveis, com interface *Wi-Fi*, conectarem-se à Internet. As funcionalidades desta tecnologia atendem às normas 802.11 estabelecidas pelo IEEE (*Institute of Electrical and Electronics Engineers*) (2).

As funcionalidades oferecidas pelos *hotspots Wi-Fi* podem ser exploradas no desenvolvimento de um *Place-Based System* (PBS) (3), usando-as como recurso para interligação entre o mundo físico e o virtual, ou seja, entre os utilizadores e a representação virtual de um local.

Os PBS focam-se na utilização da informação de localização de entidades físicas do mundo real (como utilizadores ou outros objectos), como dados para influenciar o comportamento do sistema e assim oferecer um conjunto variado de funcionalidades baseados na localização. Por sua vez, o conceito de *place* é utilizado como um elemento interligador entre o espaço físico e a utilização social de serviços digitais associados a esse espaço.

## 1.2 OBJECTIVOS

Esta dissertação tem como principais objectivos: estudar os conceitos fundamentais de PBSs, explorar o conjunto de funcionalidades obtidas da interacção das funcionalidades das redes Wi-Fi com os sistemas de computação ubíqua, e avaliar os modelos de integração destas funcionalidades com os restantes serviços e aplicações de suporte a um PBS, tal como, os recursos locais disponíveis do mundo real, enfatizando a virtualização do espaço físico, nomeadamente, os *places*. Por recursos locais podemos entender qualquer software ou hardware que possibilite a passagem de informação do local físico para o *place*, tais como as redes Wi-Fi, webcams ou qualquer outro dispositivo para este efeito.

As funcionalidades resultantes da integração com serviços e aplicações de suporte a PBSs incluem a autenticação de localização, detecção de presença ou redireccionamento de um browser para uma página associada a um determinado local físico.

É importante ter em conta que esta integração não deverá envolver a instalação de software nos Access Points (Ap's), nem o acesso à Internet, mas sim, poder recorrer à instalação de componentes de software específicos na infra-estrutura local ou nos dispositivos pessoais. Esta integração passa pela utilização oportunista de recursos e funcionalidades de hotspots *Wi-Fi*, por parte de utilizadores que se encontrem na área de cobertura, sem efectivamente efectuarem login de autenticação na mesma.

Esta conjuntura serviu de motivação para o objectivo principal desta dissertação que consiste em desenvolver um PBS que possa ser integrado em qualquer ambiente físico, desde que este esteja coberto por uma rede *Wi-Fi* e permita utilizar os recursos físicos existentes no local, que possam contribuir para a virtualização do espaço.

### 1.3 ESTRUTURA DO DOCUMENTO

Este documento está organizado em seis capítulos.

O primeiro capítulo é introdutório e apresenta o âmbito da dissertação, o seu enquadramento na realidade actual e os objectivos a atingir.

O segundo capítulo aborda trabalhos já existentes, onde as suas características relacionam-se com o tema desta dissertação, resumindo as principais questões de investigação.

No terceiro capítulo é feita a exploração do espaço de desenho na perspectiva de definir as plataformas de suporte aos *places*, tendo em conta condicionantes de acesso à Internet e controlo sobre a infra-estrutura local.

O capítulo quatro apresenta a análise do sistema a desenvolver, sendo o quinto capítulo relativo ao desenvolvimento e à implementação do PBS FindMePlaces (FMP) que resulta da especificação dos modelos de suporte e integração aos *places*. Ainda neste capítulo são apresentados os testes de integração de funcionalidades do sistema FindMePlaces em ambiente real, e é exposta a metodologia empregue para a divulgação do sistema FMP, assim como os resultados estatísticos obtidos.

O último capítulo é relativo a conclusões gerais e descrição de ideias para o trabalho futuro.



## 2 TRABALHO RELACIONADO

Os PBS são o foco deste trabalho relacionado. Neste capítulo é feita a introdução das principais propriedades ou características dos PBS. Adicionalmente, são apresentados alguns sistemas concretos, identificando as propriedades presentes em cada um deles, as diferentes abordagens adoptadas e as opções tecnológicas. A escolha destes sistemas foi realizada tendo em consideração os critérios relevantes para os objectivos da dissertação, tal como o estudo de conceitos fundamentais de PBS, características existentes nos sistemas que utilizam os serviços de localização, se algum possibilita a integração de recursos locais e, caso não o seja possível, como se poderá efectuar esta integração.

Posteriormente, é abordada a questão da integração de recursos locais em PBS e o papel da utilização oportunista de redes Wi-Fi. Finalizando esta secção, é efectuada a introdução de alguns aspectos técnicos relevantes da tecnologia Wi-Fi.

### 2.1 PLACE-BASED SYSTEMS VS LOCATION-BASED SYSTEMS

Para melhor compreensão do estudo desta dissertação é importante distinguir as diferenças entre *place-based systems* e *location-based systems*. Ambos os sistemas utilizam os serviços de localização, mas exploram a informação obtida utilizando diferentes técnicas.

Os sistemas baseados em localização, tais como os estudados, dão uso a técnicas:

#### Focadas em pessoas

- Utilizam informação baseada em localização para suportar notificações interpessoais;
- Activam comunicação informal;
- Identificam afinidades desconhecidas entre os utilizadores.

Os sistemas baseados em *places* utilizam técnicas:

#### Focadas em *places*

- Virtualização de locais físicos em *places*;
- Utilizam informação social para auxiliar navegação baseada em *places* e tomadas de decisão;

Em ambos os sistemas é obtida informação de localização, diferindo apenas na forma como essa informação é utilizada e como é apresentada aos utilizadores.

Mais concretamente, *place-based systems* (PBS) utilizam informações de localização para:

- Virtualização de locais físicos em *places*;
- Usar informação social para auxiliar a navegação e tomadas de decisão (p.e. verificar que tipo de público vai a um determinado restaurante que fica na proximidade);
- Apresentar a forma como os utilizadores exploram o place (comentários sobre o local físico, coordenação de tarefas, como por exemplo: “Encontra-se alguém neste local para me servir um café?” Navegação social: “Quem ocupa este local agora?” ; “Que música toca neste local agora?”)
- Apresentar quem está a utilizar o place no momento (*users online*);
- Apresentar informação sobre os utilizadores que visitaram o place; (histórico de utilizadores que visitaram o *place*);
- Utilizar *matching* de *places* (Lista de *places*; técnicas de selecção entre os places. Os critérios de selecção serão sempre a proximidade entre os *places* ou proximidade entre os *places* e o utilizador, etc);

*Location-based systems* (LBS), estão relacionados com a forma como é explorada a informação sobre a localização de utilizadores e como ela é apresentada no sistema, aos utilizadores.

Geralmente, as principais características de um LBS ou rede social com LBS são as seguintes:

- Partilha de localização entre amigos ( Exemplo: “Onde se encontra o amigo Z agora?”; “O amigo Y vai chegar tarde e até que ponto está atrasado? “);
- Histórico de visitas dos utilizadores; (vários locais; análise de histórico de visitas de amigos; “Onde o amigo Z costuma ir?”;” Onde o amigo Y já foi localizado?”);
- *Social matching* (“Quais dos meus amigos se encontra próximo de mim?”; “Está aqui alguém com quem posso ter interesses em comum?”; “Quem seria interessante eu conhecer?”);



- Partilha de informação de localização entre os utilizadores com conhecimento mútuo, ou seja, entre utilizadores com alguma ligação social (procura de amigos por localidade; chat dos amigos nas proximidades; “Quem das pessoas que anda nos mesmos locais poderia gostar de mim?”; “Onde eu conheci esta pessoa?”);
- Os históricos de visitas de utilizadores;

### 2.1.1 CONCEITOS

Quando falamos num *place* virtual referimo-nos a algo que não é apenas reconhecido como um espaço no mundo físico, mas também como um elemento agregador de funcionalidades associadas a esse espaço. Trata-se de uma virtualização de um espaço que já existe.

O conceito de *place* baseia-se em transpor um local do mundo físico para o mundo virtual, enriquecendo-o com informação complementar, tal como sugestões, recomendações de amigos e serviços de apoio virtuais, tornando-o um local mais apelativo para os seus visitantes e possíveis patrocinadores. O *place* permite assim uma visita ao local, através da sua representação no mundo virtual, mas também acesso a uma complementaridade de serviços a ele associados e a uma multiplicidade de perspectivas sobre o local, oriundas das diversas pessoas que por lá passaram e deixaram o seu testemunho. O conceito de *place* implica alguns desafios relacionados com a integração de recursos locais (4).

Na dimensão virtual um *place* tem um nome, um endereço e uma localização geográfica ou simbólica associada. Por outro lado, podemos criar um *place* virtual sem qualquer correspondência a um lugar no mundo físico. O *place* pode ser completamente virtual e existir apenas na dimensão virtual, sem corresponder a um lugar físico existente, mas sendo sempre associado a uma localização geográfica ou simbólica. ”O tipo de informação de localização fornecida pode ser uma “Posição Física” ou uma “Localização Simbólica”. Posições físicas, geralmente, são determinadas em termos de coordenadas geográficas globais, como no GPS, enquanto que localizações simbólicas estão associadas a significados abstractos ou nomes de locais, tais como “Laboratório” ou “Biblioteca” (5).

A questão relevante na criação do *place* é a nomeação deste. Na maior parte dos PBS os utilizadores estão livres de atribuir um nome simbólico ao *place* ou aceitar sugestões de nomes mais lógicos para o *place*. Com o intuito de otimizar a funcionalidade de

sugestão de nomes, aquando da criação de novos *places*, podemos explorar a implementação de modelos de espaço simbólico, em que estes são alimentados com dados provenientes do meio ambiente e, por sua vez, disponibilizam um serviço que permite obter informação adicional, geográfica ou simbólica, sobre uma área geográfica ou sobre um determinado *place*. Desta forma, torna-se possível pesquisar por um ponto simbólico, como por exemplo o nome do *place*, procurar por informação sobre este, ou então, pela referência simbólica à qual o *place* está associado (6).

No espaço físico, um local tem, frequentemente, as suas delimitações bem definidas. Nos *places*, a definição de fronteira ou do território onde começa e termina não existe de forma tão explícita como nos locais físicos. As delimitações de um *place* estão relacionadas com um ponto de referência geográfica, que pode ser definido por um par de coordenadas, ou então por um objecto. No caso de ser definido por um objecto, como por exemplo, um ponto de acesso, o *place* terá a área correspondente à área de cobertura deste. Os *places* podem ser sobrepostos no espaço, podendo existir mais que um parametrizado para a mesma referência simbólica.

Outra funcionalidade importante de um PBS é a gestão de presença. Esta funcionalidade é responsável por registar a presença do utilizador nos diferentes *places* e a mobilidade entre eles. Da mesma forma que na vida real entramos e saímos de locais e passamos por muitos locais durante o dia, no mundo virtual ou num PBS, a acção de entrada num local é designada por *check-in*. O *check-in* pode ser efectuado pelo utilizador, sendo neste caso um *check-in* explícito, onde o utilizador pretende partilhar a sua localização.

O outro tipo de *check-in* é designado por *check-in* implícito, onde ocorre a detecção de presença de utilizador sem nenhuma acção humana. No *Instant Places* (4) o mecanismo de *check-in* é suportado pela associação dos dispositivos dos utilizadores com identificadores *Bluetooth* à localização e, subsequentemente, efectuada a detecção dos dispositivos dos utilizadores, pelo *Bluetooth scanners*. No Foursquare (7), sendo este um *Location-Based Social System* (LBSS), o *check-in* é efectuado pela detecção de características da interface *Wi-Fi* dos dispositivos de utilizador.

Tal como num espaço físico, podemos executar determinadas acções nos *places*, onde estas são virtualizadas e permitem utilizar diferentes funcionalidades, tais como partilha de informação, comentários e fotografias, correspondendo à interacção de pessoas no mundo físico.

Actualmente já existem algumas implementações de conceitos anteriormente descritos, baseados em aplicações Web e em *Location Based Services* (LBS). Algumas oferecem funcionalidades interessantes na interpretação da relação entre o espaço físico e virtual, tal como o Virtual Graffiti<sup>1</sup>. Este sistema baseia-se na associação de objectos virtuais, chamados *graffitis*, que podem ser imagens multimédia, mensagens digitais ou outras anotações em locais públicos, paredes de edifícios ou estações de comboio e metro (8).

No projecto QRpedia<sup>2</sup>, os utilizadores podem imprimir etiquetas ou Tags com códigos QR e fixá-los nos objectos relacionados do mundo real, criando assim uma hiperligação do mundo físico para o mundo virtual, neste caso para o serviço Wikipedia. Os códigos QR contêm informações sobre o URL de artigos da Wikipedia. Posteriormente, qualquer utilizador equipado com um dispositivo móvel pode tirar uma fotografia à Tag e obter informações contidas no código QR (9).

#### 2.1.2 FOURSQUARE

Entre muitos *place-based systems* que têm surgido na última década, destaca-se o Foursquare (10). O Foursquare permite associar as actividades de utilizadores à sua localização, explorar novos locais e partilhar conteúdos, fotos, ideias, eventos e tudo o que está relacionado com os locais visitados, na sua rede de amigos. Este também possibilita coordenar a interacção entre os utilizadores, baseando-se na presença dos seus amigos em locais e nos eventos que ocorrem nestes locais.

Os utilizadores podem escolher um *place* através de uma lista, procurar um *place*, ou criar um novo *place*. Para cada *check-in*, os utilizadores podem escolher o estado de não o partilhar com ninguém, partilhá-lo com os seus amigos do Foursquare ou transpor os *check-ins* para o facebook ou twitter.

Ao efectuar o *check-in*, os utilizadores vêem que outros utilizadores já fizeram *check-in* naquele *place* nas últimas 3 horas e o total de *check-ins* naquele *place* desde a sua criação. No Foursquare, os nomes são gerados na comunidade de utilizadores, o que dá logo uma imagem de maior interactividade e bidireccionalidade no fluxo de informação. Os *check-in's* podem ser privados, tanto nos *places* públicos, que são acessíveis a todos, como nos *places* privados.

---

<sup>1</sup> <http://tagging.us/html/pr.html>

<sup>2</sup> <http://qrpedia.org/>

Só os *places* nomeados como “home” são considerados privados, não sendo visíveis, a menos que os utilizadores os procurem explicitamente pelo seu nome. O Foursquare emprega elementos como “*points*”, “*badges*” e “*mayorships*” para aumentar o comprometimento com o serviço e motivar os *check-in*'s com mais frequência. Por exemplo, os utilizadores obtêm pontos por fazer *check-in*, *badges* por certas combinações de *check-in* e *mayorships* de *places* pelo maior número de *check-ins*.

Os utilizadores também podem obter vantagens comerciais, como a oferta de descontos, baseados em *check-ins* e *mayorships*. Os utilizadores podem procurar por *places* e fazer *check-in* quando, efectivamente, não estão lá. Nesse caso, não recebem pontos ou *mayorships* quando as suas coordenadas GPS ou a sua identificação de rede e respectiva localização não coincidirem com a localização geográfica do *place*.

Os utilizadores desta aplicação têm uma página de perfil publicamente acessível. Eles podem escolher usar o seu nome real e fornecer um link para os seus perfis do Facebook e do Twitter.

O posicionamento dos utilizadores no Foursquare é efectuado de modo automático, logo que estes se autenticam e acedem ao site. Aquando do seu acesso, a posição do utilizador é visível no mapa conforme os dados do último *check-in* efectuado, sendo solicitado ao utilizador a autorização de partilha da sua localização actual, o que fará com que a sua posição seja actualizada no mapa, caso ele autorize.

Para determinar a posição do utilizador, o Foursquare utiliza funcionalidades de geolocalização incluídas na API do browser, definidas pela W3C (11).

*The Geolocation API defines a high-level interface to location information associated only with the device hosting the implementation, such as latitude and longitude. The API itself is agnostic of the underlying location information sources. Common sources of location information include Global Positioning System (GPS) and location inferred from network signals such as IP address, RFID, WiFi and Bluetooth MAC addresses, and GSM/CDMA cell IDs, as well as user input (W3C2010).*

Apenas as versões mais recentes dos browsers suportam a API de geolocalização conforme as normas W3C, nomeadamente o Firefox versão 3.5+, Opera versão 10.6, Google Chrome versão 5 +, Internet Explorer versão 9 +.

Através da aplicação móvel nativa ou acedendo à aplicação Web do Foursquare é possível ao utilizador realizar o *check-in* no *place* pretendido, recebendo pontos apenas

se estiver fisicamente no local. Como meio de verificar e comprovar a posição actual do utilizador e confrontá-la com aquela em que ele pretende realizar o *check-in*, existem diversos mecanismos que permitem obter e comparar os dados de localização do utilizador: IP address , GSM or CDMA cell phone ID.

Após tentar efectuar um *check-in* através de aplicação móvel ou no Web site [m.foursquare](http://m.foursquare.com)<sup>3</sup>, o utilizador pode criar um *place*, quando este não existe na lista de *places* disponíveis. Para tal, é necessário inserir o nome do *place* e indicar a sua localização física e, caso os dados introduzidos sejam válidos, é criado um novo *place*.

Na criação de um *place* terá de haver uma correspondência entre a sua localização virtual e a sua localização no mundo real. No entanto, no momento da criação, não é obrigatório que a localização física fornecida seja a verdadeira ou a mais específica, sendo permitido ao utilizador criar um *place* com uma localização diferente do local físico onde ele se encontra.

Os *places* são sempre associados a um ponto de referência geográfica, não implicando que exista correspondência no mundo físico para o *place* criado no mundo virtual.

Outras funcionalidades são disponibilizadas nesta abordagem, tais como a submissão de opiniões sobre lugares físicos, criação de listas de lugares a visitar e outras sugestões.

### 2.1.3 FACEBOOK PLACES

Os serviços de localização trazem funcionalidades importantes para o Facebook. Com o Facebook Places <sup>4</sup>os utilizadores podem adicionar a sua localização a tudo o que partilham.

Os Facebook Places estão disponíveis através de uma aplicação que pode ser instalada nos dispositivos móveis, como também na página Web. Depois de se autenticar na página, o utilizador é questionado se deseja partilhar a sua localização. Se autorizar, o Facebook Places apresentar-lhe-á a lista de *places* e eventos nas proximidades. O utilizador poderá fazer o *check-in* em qualquer um dos *places* da lista ou pesquisar por um outro e efectuar o *check-in*, mesmo que não se encontre fisicamente no local. Também é possível ao utilizador fazer o *check-in* sem partilhar a sua localização.

---

<sup>3</sup> <https://foursquare.com/mobile/login?continue=%2Fmobile%2F>

<sup>4</sup> <https://touch.facebook.com/>

Depois de efectuar o *check-in* o utilizador pode navegar para a página do *place*. O utilizador pode criar um *place* e atribuir-lhe qualquer nome sem que este tenha correspondência no mundo físico. Os novos *places* ficam associados a um ponto de referência geográfica correspondente à localização do utilizador.

#### 2.1.4 LOCAMODA

O *place-based media system* Locamoda<sup>5</sup> é uma evolução do Wiffiti<sup>6</sup>. O Locamoda permite mostrar em ecrãs públicos os comentários publicados no Twitter<sup>7</sup> de um determinado utilizador e que correspondem a uma determinada tag. No Twitter, as tags ou palavras-chave são utilizadas para categorizar os comentários por assuntos, facilitando a recuperação de dados no sistema quando é efectuada a procura.

As aplicações do Locamoda permitem aproveitar o conteúdo das redes sociais que se tornaram o centro de partilha de informação ou geradores de informação, como também, filtrar os comentários de utilizadores e mostrá-los no ecrã, em espaços públicos, obtendo assim o conteúdo dinâmico, sendo esta forma de partilha de informação ideal para os inventos direccionados a um tema específico. O Locamoda também aproveita os sistemas baseados em localização, como o Facebook Places e o Foursquare, para apresentar os *check-ins* em ecrãs públicos.

#### 2.1.5 PINPOINT

O PinPoint<sup>8</sup> é um PBS que permite marcar os *places* e eventos com pins no mapa e agrupá-los em conjuntos. Permite ainda enviar aos utilizadores os convites para os eventos e *places*, como também procurar os pins criados por outros utilizadores.

O pin é um *place* ou evento marcado no mapa, pelo utilizador. Todos os pins possuem nome, subtítulo, morada e descrição. Para criar um pin, o utilizador deve seleccionar a opção “Create a pin” no menu principal, procurar no mapa o *place* pretendido, escolher a opção “Pinpoint this location” e introduzir as suas propriedades.

---

<sup>5</sup> <http://locamoda.com/>

<sup>6</sup> <http://wiffiti.com/>

<sup>7</sup> <https://twitter.com/>

<sup>8</sup> <http://pin.pn/dashboard>

### 2.1.6 ANÁLISE

Nesta secção é apresentada uma análise comparativa das funcionalidades e características dos vários sistemas que utilizam os serviços baseados em localização (LBS), sendo usado um conjunto de métricas adequadas:

Métrica 1 (M1): Principais objectivos do sistema;

Métrica 2 (M2): Como é que se cria um *place*;

Métrica 3 (M3): Como é efectuada a nomeação do *place*;

Métrica 4 (M4): Tipos de check-in realizados no sistema;

Métrica 5 (M5): É efectuada detecção de presença de utilizadores;

Métrica 6 (M6): É possível fazer check-in num local sem estar lá fisicamente;

Métrica 7 (M7): Um *place* precisa de ter correspondência a um local no mundo físico (real);

Métrica 8 (M8): Os *places* estão sempre associados a um ponto de referência geográfica;

	M1	M2	M3	M4	M5	M6	M7	M8
Foursquare	location-based social network	Manual	Manual	Explicito / implícito	Sim	Sim	Não	Sim
Facebook Places	social network with LBS	Manual	Manual	Explicito	Não	Sim	-	Sim
Locamoda	social media network	-	-	-	Não	-	-	-
Pinpoint	place-based social media	Manual	Manual	Explicito	Não	Sim	Não	Sim

Tabela 1: Tabela comparativa de funcionalidades.

Foi identificado que a maioria dos sistemas é orientado a redes sociais de amigos, como o Facebook Places, onde a localização é apenas uma funcionalidade adicional às

publicações de utilizadores ou apenas a identificação de um *place* no mapa, como no caso do Pinpoint.

O sistema mais próximo de um sistema orientado a *places* é o Foursquare, onde é possível associar conteúdos à localização. Mesmo assim, identificou-se que não é possível explorar todos os recursos, serviços e aplicações associados a um local físico, nomeadamente os recursos locais.

## 2.2 TECNOLOGIA WI-FI

Nesta secção aborda-se a tecnologia *WI-FI* como sendo um recurso local que pode ser integrado nos PBS. Na secção 2.2.1 é apresentado o exemplo de funcionamento da tecnologia captive portal, que pode ser integrada em PBS como mecanismo de redireccionamento de utilizadores para diferentes páginas no browser.

O surgimento da tecnologia Wi-Fi criou um novo conjunto de oportunidades no que concerne à virtualização do mundo físico. Antes do surgimento desta tecnologia, o acesso ao mundo virtual restringia-se aos locais físicos com infra-estrutura de rede de acesso por cabo, o que impossibilitava explorar todas as funcionalidades que um PBS disponibiliza.

Com o aparecimento dos hotspots Wi-Fi, estas limitações foram ultrapassadas, sendo estes um recurso de maior mobilidade e que não necessita da introdução de componentes na infra-estrutura para a integração entre as funcionalidades Wi-Fi e as dos PBS. Por outro lado, um novo desafio nasceu, visto que a estrutura da rede Wi-Fi de um espaço é, na maior parte das vezes, desconhecida. Verifica-se também que os recursos tecnológicos API, que permitem tirar partido da tecnologia Wi-Fi, são muito escassos. A tecnologia Wi-Fi cria oportunidades para a ligação entre o mundo físico e virtual, podendo ser entendida por ela mesma como um recurso local, no momento que actua como sensor de detecção de presença de dispositivos existentes no local.

As redes *Wi-Fi* têm dois modos de funcionamento:

- Em modo de infra-estrutura, onde cada dispositivo *Wi-Fi* (Station) comunica com um ponto de acesso ou “*Access Point (AP)*” central para poder ter acesso a outros dispositivos na rede;



- Em modo *ad-hoc*, onde os dispositivos trocam informações directamente uns com os outros dentro de uma determinada área, sem a necessidade de uma infraestrutura previamente instalada.

### 2.2.1 CAPTIVE PORTAL

O diagrama apresentado na figura 1 exemplifica um cenário de autenticação de utilizadores através do mecanismo de captive portal. A troca de mensagens é realizada entre o Cliente, os componentes de sistema WilmaGate (12), o Gatekeeper e o Gateway, Servidor DNS, responsável por resolver os nomes de domínio em endereços IP, Proxy Server, Authenticator e Web site onde o cliente pretende navegar.

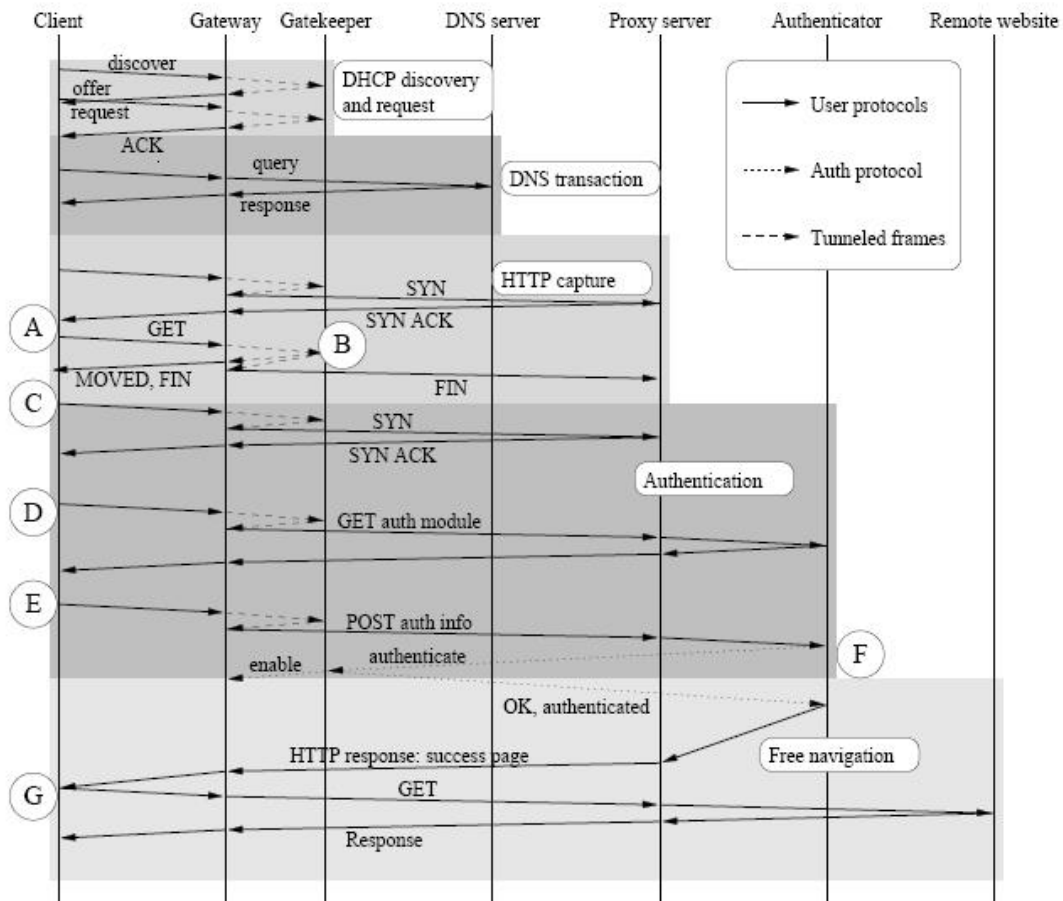


FIGURA 1: DIAGRAMA DE SEQUÊNCIA PROCESSO DE AUTENTICAÇÃO DE CLIENTE COM MECANISMO DE CAPTIVE PORTAL , FONTE (13).

Quando o cliente obtém o endereço IP, ainda não lhe é permitida a sua navegação na internet. Quando o browser aponta para um URL, primeiro, realiza uma consulta ao DNS, para o qual o gateway é transparente. Para a consulta ao DNS interessa o nome de domínio do website pedido ou, se o proxy server estiver correctamente configurado, o

IP do Proxy. (Necessita do IP do proxy para eventualmente questionar recursivamente outros DNS Servers). Se o servidor proxy estiver devidamente configurado, o nome de domínio é resolvido em endereço IP.

Depois de obter a resposta do DNS, o browser estabelece a conexão com o servidor proxy. Depois da conexão aberta, um pedido HTTP GET é enviado através do proxy server. (label A). O pacote contendo o pedido GET é interceptado. O pacote que contém o pedido HTTP GET é interceptado pelo GateKeeper (label B) que forja um pacote de resposta simulando ser o servidor proxy. A resposta contém um código HTTP 302 que redirecciona o cliente para a página de autenticação. Para além disso, o package contém o FIN, que também é forjado pelo GateKeeper, de modo a que o cliente feche a conexão com o servidor proxy.

O pacote FIN é forjado e enviado para o servidor Proxy pelo GateKeeper, simulando que o FIN foi enviado pelo utilizador (label C). A conexão é fechada e o utilizador é redireccionado para a página de autenticação. O fecho da conexão é necessário para que, depois de forjar o pacote, o número de sequência de resposta no cliente e no servidor proxy não correspondam. De seguida, o processo de autenticação inicia-se com o cliente a abrir uma conexão no proxy Web.

O próximo pedido GET (actualmente pedido CONNECT destinado para o servidor seguro (label D)) é direccionado para a página de login do servidor de autenticação de forma a ser admitido pelo Gatekeeper. Após obter a página, o utilizador introduz os dados de login como credenciais de autenticação (label E) que serão recolhidas por um script em php, que as enviará para o servidor de autenticação (label F), o qual as verificará e, depois de aprovadas, contactará o módulo de autorização do Gatekeeper de modo a enviar um comando de autorização. O Gatekeeper mantém a sua própria tabela, envia um comando de autorização ao gateway, desbloqueia os pacotes do utilizador e responde ao servidor de autenticação. O servidor de autenticação, por seu lado, envia uma página de sucesso ao cliente, passando este a ser livre para navegar na Web.

Todas as informações de autenticação são trocadas directamente entre o utilizador e o servidor de autenticação via canal seguro HTTPS. Desta forma, o processo de autenticação não pode ser encriptado pelo operador de acesso, que nunca terá capacidade de descobrir as credenciais de autenticação. O resultado final do procedimento de autenticação é o comando de autorização enviado pelo provider de

autenticação para o WilmaGate (12). De qualquer forma, este comando não contém dados de autenticação. O Gatekeeper apenas necessita de saber se o cliente foi autenticado com sucesso e pelo provider de confiança. Para garantir a autenticidade do cliente durante o processo de autenticação, um token aleatório de 16-bytes é gerado pelo Gatekeeper durante o processo de redirecionamento (label B) e é inserido em todas as subsequentes autenticações nas páginas Web. O mesmo token é fornecido em label F.



### 3 ANÁLISE

Neste capítulo é feita a análise do potencial de integração de recursos locais num PBS, onde as principais características são a autenticação de localização, detecção de presença, gestão de places e integração de recursos locais.

Com esse intuito, é feita a conceptualização do espaço de desenho das plataformas de suporte aos places, evidenciando o papel dos recursos, actores, características de acesso à rede e o controlo sobre a infra-estrutura local.

#### 3.1 ESPAÇO DE DESENHO

Com esta secção pretende-se definir o espaço de desenho e os elementos que dele fazem parte, assim como as suas condicionantes, sendo esta definição essencial para a especificação de modelos de suporte aos *places* e posterior elaboração de modelos de integração dos *places*. A possibilidade de acesso à internet e o controlo sobre a infra-estrutura são os principais factores que influenciam o desenho da solução.

##### 3.1.1 DESCRIÇÃO DE ACTORES

Num espaço físico identificam-se os seguintes participantes nos cenários de utilização de um *Place-based System*: o utilizador, o criador dos *places* e o proprietário de infra-estrutura. Todos estes actores podem criar *places* associados ao espaço físico, tendo privilégios diferentes.

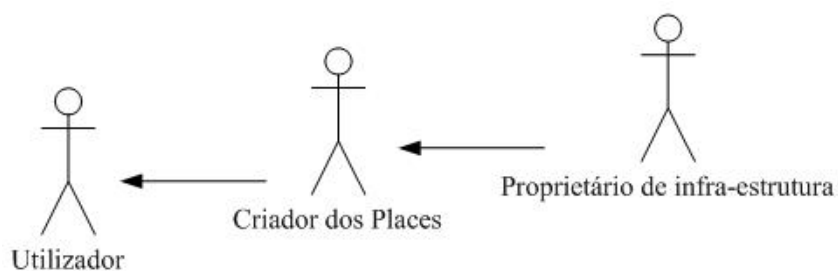


Figura 2: Representação de diferentes privilégios de actores no sistema FindMePlaces.

### Utilizador

Em ambiente real, este actor representa uma pessoa que se encontra num espaço físico e possui um PC ou outro dispositivo com interface wireless devidamente configurada.

É possível ao Utilizador usufruir dos *places* associados a um espaço físico onde se encontra, efectuar *check-ins* e utilizar outras funcionalidades da aplicação, como submeter os comentários ou imagens relativas ao *place*. O Utilizador pode ou não ter acesso à internet, dependendo dos privilégios que lhe foram atribuídos.

### Criador dos *places*

Este actor representa um utilizador que cria um *place* associado a um espaço físico, possuindo todos os privilégios de Utilizador, com a diferença que este tem menos limitações em utilização de recursos locais que, por sua vez, dependem do modelo de suporte onde se encontra.

### Proprietário da infra-estrutura

Em ambiente real, este actor representa uma pessoa ou empresa proprietária de um espaço físico. O Proprietário da infra-estrutura assume a responsabilidade de gerir os recursos físicos do mesmo, cuja utilização contribuirá para a representação virtual do seu espaço.

Outra das suas funções passa por definir os privilégios dos Criadores dos *places* quanto aos recursos locais que podem ser por eles utilizados na acção de criação de um novo *place*.

O proprietário da infra-estrutura pode ceder privilégios de gestão da infra-estrutura a um Criador de *places*, se assim se justificar, assim como assumir ele o papel de Criador de *places*.

### 3.1.2 CENÁRIOS DE CRIAÇÃO E UTILIZAÇÃO DOS PLACES

Num cenário geral, os intervenientes acedem à página de *places* no seu browser e podem efectuar *check-ins* em *places* existentes no sistema, criar um novo *place*, partilhar conteúdos nos *places*, gerir recursos ou explorar outras funcionalidades. As funcionalidades do sistema FindMePlaces, com vista à criação de *places*, variam em função dos recursos existentes em cada espaço, assim como os privilégios de cada interveniente.

Com o objectivo de demonstrar as possíveis utilizações dos diferentes privilégios dos três actores, em cenários reais, definiram-se quatro tipos de cenário de integração de um PBS.

Existem duas principais variáveis que marcam as vantagens da integração de um PBS em cenários reais e o impacto que têm na interacção dos actores com os mesmos, tendo em conta os seus privilégios: o acesso à internet e a possibilidade de controlo sobre a infra-estrutura. Combinando estas duas variáveis, obtemos quatro cenários possíveis, os quais estão ilustrados na figura 3.

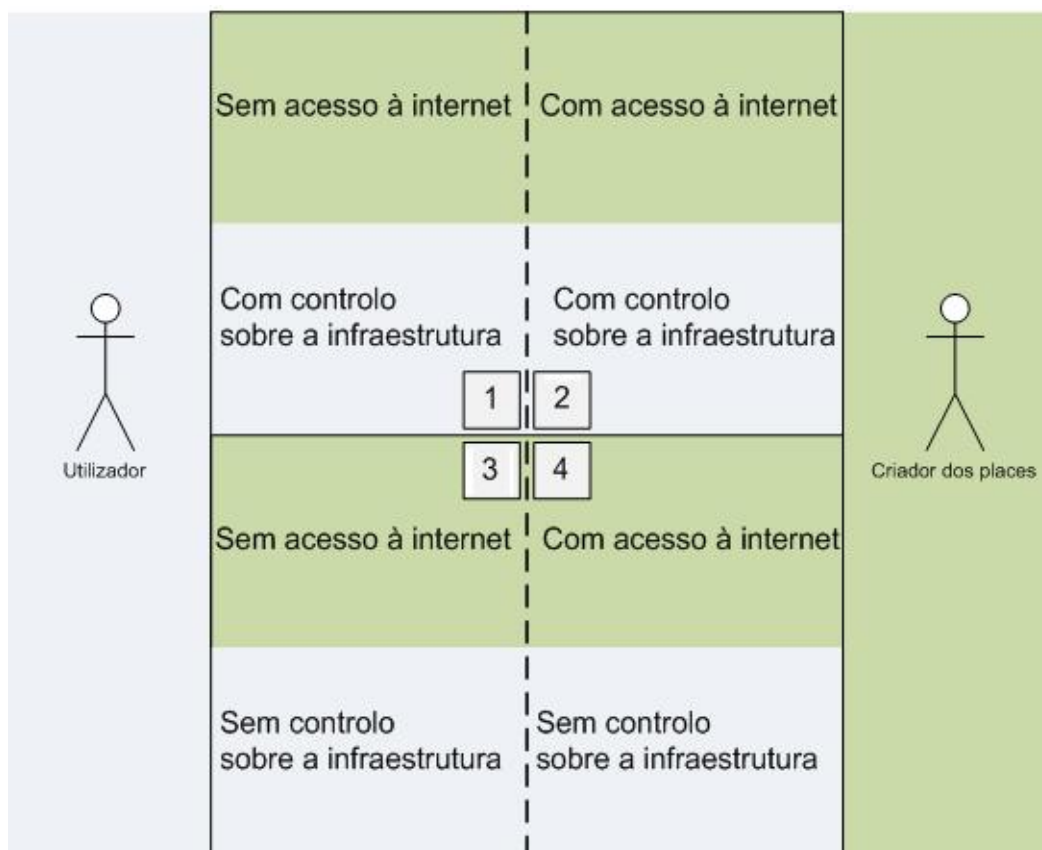


Figura 3: Cenários de utilização do Sistema FindMePlaces.

O recurso relevante para o Utilizador é a possibilidade de acesso à Internet sendo este uma possível condicionante à exploração das funcionalidades do *place*. O controlo sobre a infra-estrutura do espaço físico é um factor relevante para o Criador dos *Places*. Tendo o acesso à infra-estrutura, poderão adoptar-se soluções que envolvem a instalação de componentes de software/hardware específicos na infra-estrutura local. O Proprietário da infra-estrutura terá o acesso à internet e o controlo sobre a infra-estrutura em todos os casos possíveis. Os quatro cenários de utilização correspondem, em termos

de recursos disponíveis, aos ambientes reais onde a PBS poderá ser integrada.

De seguida, serão apresentados os cenários na perspectiva de exemplificar os exemplos da vida real onde se podem aplicar os PBS e abrangendo todas as possibilidades que resultaram das condicionantes.

### CENÁRIO 1 BAR

Neste primeiro cenário retratamos a possibilidade de utilização de um PBS em ambiente **sem** acesso à internet mas **com** o controlo sobre a infra-estrutura.

*Quando queremos frequentar um estabelecimento de lazer, como um bar ou uma discoteca, temos interesse em saber o ambiente que caracteriza o estabelecimento, não só em questões visuais, como também o tipo de música que é reproduzida. Muitas vezes, quando nos encontramos perto do local, a única maneira de ter conhecimento das características do estabelecimento é entrando, o que, em muitos sítios, implica pagar entrada.*

*Um PBS integrado na infra-estrutura da rede local do estabelecimento pode contribuir para a virtualização do espaço em questão. O funcionário do estabelecimento assume o papel de criador de places, em que tem acesso à infra-estrutura da rede local. Ele cria um place associado ao estabelecimento, onde, não só disponibiliza informação de menus, eventos, horários e outras informações relevantes, como também disponibiliza fotografias e integra um software de música, onde as playlists reproduzidas, estão a ser disponibilizadas em tempo real no place criado.*

*O utilizador, estando ao alcance da AP do estabelecimento, pode-se associar-se à rede, sendo redireccionado para a página Web do estabelecimento. Apesar do cliente não ter acesso à internet, este consegue ter uma experiência virtual, muito semelhante à realidade encontrada no local, podendo assim decidir se quer conhecer a realidade do estabelecimento.*

Neste cenário é possível fornecer as funcionalidades que fazem uso de serviços ou recursos locais sobre o ambiente dentro de um *place*, menus, playlists, fotografias, etc.

Também deve ser utilizada a técnica de captive portal, que encaminha o utilizador para a página do sistema, logo que este tenta aceder a um browser.



## CENÁRIO 2 HOTEL

Neste segundo cenário retratamos a possibilidade de utilização de um PBS em ambiente **com** acesso à internet e **com** o controlo sobre a infra-estrutura.

*Hoje em dia, a maioria dos hotéis disponibilizam acesso grátis à internet, via Wi-Fi, aos seus clientes, de maneira a que estes tenham todas as comodidades possíveis, aproximando-os do conforto existentes nas suas próprias casas.*

*A existência de um PBS integrado na infra-estrutura de rede do hotel possibilita aprofundar este conceito de conforto.*

*O gestor da rede do hotel cria places associados a diversas áreas do hotel e, tendo acesso à infra-estrutura, decide adicionar câmaras Web nos locais de acesso público do hotel e associá-las aos respectivos places.*

*Imaginemos que um cliente do hotel não gosta de estar na piscina com muito movimento ou que um casal deseja ter um jantar num local calmo. Facilmente estes clientes poderão aceder à rede do hotel, associando-se a um AP disponível, ou através da página Web do hotel, e consultar os places desejados, tendo acesso a imagens dos places sem terem de sair do quarto, podendo assim decidir antecipadamente se não vão até à piscina porque está muita gente, ou se usufruem do serviço de quarto em vez de se deslocarem ao restaurante, devido a este estar demasiado lotado.*

O cenário descrito apresenta uma maior possibilidade, relativamente ao cenário anterior, de explorar as funcionalidades ligadas aos serviços locais, tais como, câmaras Web, Ap's, ou Internet.

## CENÁRIO 3 AEROPORTO

Neste terceiro cenário retratamos a possibilidade de utilização de um PBS em ambiente **sem** acesso à internet e **sem** o controlo sobre a infra-estrutura.

*Existem momentos em que, quando viajamos de avião, passamos muito tempo nos aeroportos, seja por atrasos de voos ou escalas de longa duração. Durante os períodos de espera, gostamos de fazer alguma coisa para passar o tempo, mas também sentimos a necessidade de estarmos sempre informados da situação dos voos, assim como gostamos de ter informação útil, acessível sem grande esforço.*

*Uma das distrações possíveis existentes nos aeroportos é a zona comercial, onde podemos visitar várias lojas.*

*Imaginemos um aeroporto em que um PBS se encontra integrado na infra-estrutura de rede, efectuando a localização baseada nos APs disponíveis. Os gerentes de cada loja, apesar de apenas terem acesso aos recursos disponibilizados pelos gestores da rede do aeroporto, podem criar um place associado à sua loja e disponibilizar fotografias.*

*O utilizador, enquanto passeia pelo aeroporto, pode usar o seu dispositivo móvel (por exemplo um smartphone) para aceder à rede local do aeroporto, disponível através dos vários APs. Ao aceder à rede é automaticamente redireccionado para a página do aeroporto, onde é constantemente actualizada a informação dos voos, é disponibilizado o mapa do aeroporto e também uma lista de places perto da sua localização. Acedendo aos places disponíveis, o utilizador pode descobrir mais facilmente o que deseja, ou apenas ir a uma determinada loja, apenas pelo interesse despertado pelo conteúdo disponível no place visitado.*

No panorama geral, este cenário é mais limitado relativamente aos outros, tanto na exploração das funcionalidades relacionados com recursos físicos de local como funcionalidades *Wi-Fi*.

#### CENÁRIO 4 UNIVERSIDADE

Neste quarto cenário retratamos a possibilidade de utilização de um PBS em ambiente **com** acesso à internet mas **sem** o controlo sobre a infra-estrutura.

*Assumimos que este cenário se passa numa universidade, onde um PBS se encontra integrado na infra-estrutura de rede local.*

*Um professor, como utilizador, acede à página Web da universidade e efectua o devido login na sua área. Assumindo o papel de criador de places, decide criar um place associado ao departamento de sistemas de informação.*

Este ultimo cenário permite a integração das funcionalidades *Wi-Fi* mas não permite a integração de recursos locais.

O espaço de desenho definido acima, resulta em quatro modelos de suporte aos places ou protótipos:

1. Sem acesso à internet/ com controlo sobre a infra-estrutura;
2. Com acesso à internet/ com controlo sobre a infra-estrutura;

3. Sem acesso à internet / sem controlo sobre a infra-estrutura;
4. Com acesso à internet / sem controlo sobre a infra-estrutura.

Podemos averiguar em qual dos quatro modelos de suporte aos places podemos aplicar as soluções tecnológicas e abordagens às funcionalidades de PBS existentes e apresentadas no Capítulo 2. Este estudo é feito com o intuito de identificar quais são os recursos tecnológicos viáveis para cada um dos modelos de suporte.



## 4 CONCEPÇÃO DO SISTEMA

Neste capítulo será descrita a abordagem adaptada para implementação dos modelos de suporte aos *places* e à experimentação de dois dos quadrantes do espaço de desenho, com vista à avaliação dos problemas que advém da integração. Para tal, foram implementados dois protótipos distintos que encaixam nos primeiros dois cenários. Os protótipos baseados em APIs de localização distintas correspondem aos modelos de integração 1 e 2.

### 4.1 REQUISITOS DO SISTEMA

O FindMePlaces (FMP) é um *place-based system* cujo objectivo é proporcionar ao utilizador a transcrição de um espaço físico em dimensão virtual, aproveitando toda a informação existente e os seus recursos locais.

Os principais requisitos do sistema FMP são essencialmente os seguintes:

- Variedade e abrangência funcional – Permite a utilização de serviços e recursos disponíveis no espaço virtual, com o objectivo de promover a interacção com locais físicos existentes nas proximidades, tais como serviços de meteorologia, mapas, câmaras e menus de restaurantes.
- Funcionamento Personalizado – ter em consideração a identidade do utilizador. O sistema deve ter uma página de boas-vindas personalizada e ter em conta o histórico de visitas e preferências do utilizador.
- Adaptabilidade ao âmbito do utilizador – De acordo com a localização do utilizador, o sistema deve ser capaz de se adaptar às características do ambiente físico, possíveis de serem determinadas, de forma a o caracterizar. Tais características podem passar pela informação de outros utilizadores situados no local, as condições meteorológicas ou tráfego rodoviário.
- Pro-actividade - Capacidade de agir automaticamente sem a intervenção do utilizador, desencadeando acções, tais como a sugestão de nomes no momento de criação de *places* ou a detecção de presença do utilizador, no caso do *check-in* implícito.

## 4.2 ARQUITECTURA DO SISTEMA

Para pôr os modelos de integração em prática, foi necessário desenhar uma arquitectura de suporte, funcional para todos os modelos. O PBS FindMePlaces deverá apresentar uma arquitectura semelhante à figura 4.

A arquitectura do sistema é dividida em 2 áreas, sendo a primeira representativa do espaço virtual e composta pelos componentes do sistema a desenvolver. A segunda área diz respeito ao espaço físico, que representa o ambiente físico no qual o sistema está aplicado.

Dada a importância da área do espaço físico, esta foi incluída no desenho da arquitectura. Esta inclusão foi efectuada com o objectivo de realçar que os recursos locais disponíveis também podem ser utilizados pelo sistema a desenvolver.

Dentro de espaço físico existe a componente de recursos locais que diz respeito a todos os recursos de um determinado espaço físico. Estes recursos variam com a alteração da localização física do utilizador.

O diagrama de componentes ilustrado na figura 4 captura a estrutura física do sistema a desenvolver, contendo os componentes e ligações entre eles.

Na área do espaço virtual deverão estar presentes os três componentes: a Interface Web, Servidor Web e o Servidor de BD.

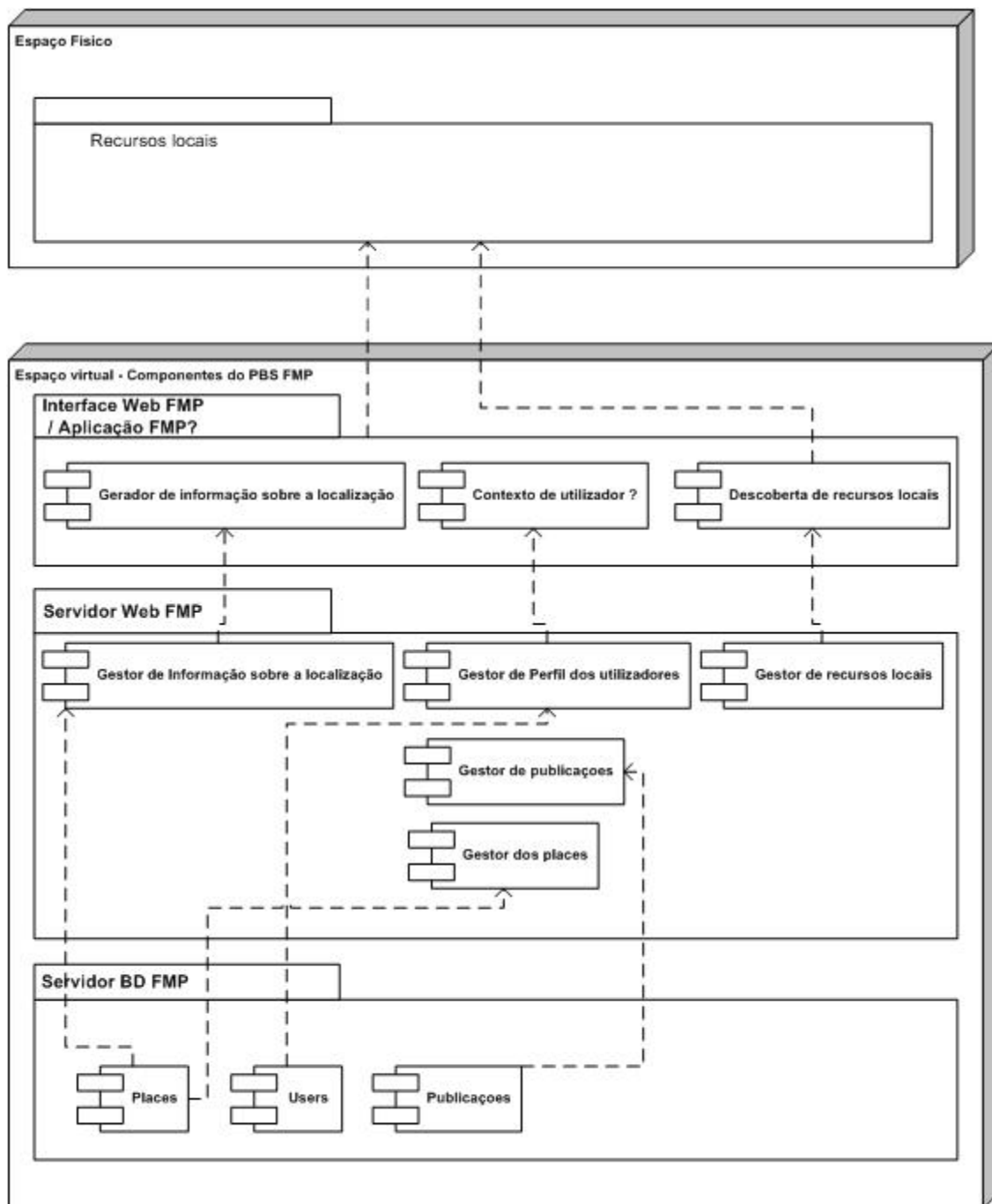


Figura 4: Arquitectura funcional do sistema FindMePlaces.

#### 4.2.1 FMP- INTERFACE WEB

A interface Web é constituída por um grupo de páginas Web. Esta disponibiliza as várias funcionalidades que correspondem a um grupo de procedimentos, cujo objectivo é satisfazer os pedidos dos utilizadores.

Gerador de informação sobre a localização – este componente é responsável por interagir com as API's externas que permitem determinar a localização do utilizador, tais como a API do Google Latitude e a API da Gecko (13); A informação de localização obtida através da API é de posteriormente transferida para o Servidor Web.

Âmbito de utilizador – são as várias interfaces Web que permitem aos utilizadores interagir com o sistema, permitindo-lhes efectuar operações de autenticação, gestão de presença e publicação de conteúdos.

Localizador de recursos locais – este componente é responsável por descobrir recursos físicos disponíveis na vizinhança física do utilizador, os quais poderão ser integrados no sistema. Conforme a localização do utilizador se altera, também se alteram os recursos disponíveis no espaço físico. Foram definidos dois tipos de recursos genéricos que podem ser detectados e posteriormente utilizados no processo de criação do *place*. O primeiro tipo diz respeito a recursos de hardware, tais como Webcams ou microfones. O segundo tipo refere-se a recursos de software, como AP's *Wi-Fi*.

#### 4.2.2 SERVIDOR FMP

O servidor FMP é o componente fundamental do sistema, sendo este responsável pela coordenação entre todos os componentes do mesmo. O servidor FMP, para além da conexão com a Interface Web, também comunica com o Servidor de Base de Dados. O servidor FMP tanto pode residir numa máquina na rede global (Internet), como também pode ser configurado na rede local do espaço físico, como demonstrado nos cenários em que não existe o acesso à Internet, nem existe controlo sobre a infra-estrutura. Este Servidor engloba cinco componentes, sendo estes:

Gestor de informação sobre a localização – é o componente responsável por processar a informação sobre a localização actual do utilizador, recebida via interface Web. Esta informação pode ser utilizada para pesquisa de *places* e para registar o *check-in* do utilizador, enviando a informação para o servidor da Base de Dados.

Gestor de perfil dos utilizadores – componente responsável por gerir toda a informação relativa ao perfil de utilizador: histórico, registo de visitas e registo de *places* criados.



Gestor de recursos locais – recebe e processa a informação sobre os recursos detectados pelo localizador de recursos.

Gestor de publicações - interage com a base de dados, adicionando e removendo os comentários e fotografias e organizando-as, de forma a que possam ser apresentados ao utilizador através da interface Web;

Gestor de *places* - componente responsável pela organização de *places* existentes na base de dados, possibilitando a apresentação destes ao utilizador, via interface Web. O gestor de *places* também controla o processo de envio de informação para o servidor de base de dados, necessária à inserção e remoção de *places*.

#### 4.2.3 SERVIDOR BD FMP

No servidor BD FMP encontra-se a Base de Dados, onde é armazenada toda a informação necessária para o funcionamento do sistema FMP. A base de dados tem como principal função armazenar os dados de utilizadores, de *places* e de publicações.

#### 4.2.4 MODELO DE DADOS

Para o sistema FMP foi definida uma base de dados designada por FindMePlaces. Nesta base de dados existem 6 tabelas:

- *places*;
- *users*;
- *visits*;
- *publications*;
- *aps*;
- *places\_aps*;

## Modelo Entidade – Relacionamento

Apresentamos de seguida a descrição das diferentes tabelas do modelo de dados.

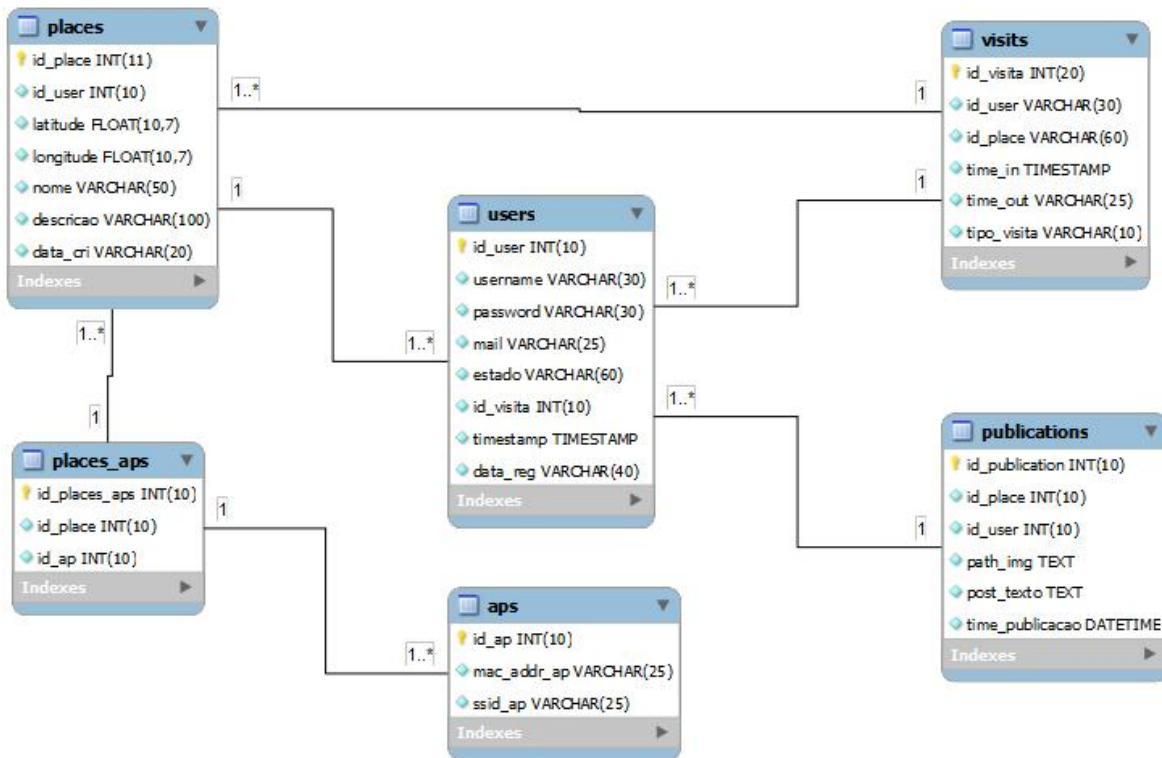


Figura 5: O modelo Entidade – Relacionamento da base de dados FindMePlaces.

#### 4.2.5 DESCRIÇÃO DAS TABELAS

##### Tabela *places*

Contém a lista de todos os *places*, com a indicação do identificador do utilizador (*id\_user*), as coordenadas geográficas (latitude, longitude), nome do *place* (nome), a sua descrição (descrição) e a data de criação.

Campo	Tipo de dados	Características
<i>id_places</i>	int(10)	primary key
<i>id_user</i>	int(10)	foreign key
latitude	varchar(25)	-
longitude	varchar(25)	-
nome	varchar(25)	-
descrição	varchar(25)	-
data	date	-

Tabela 2: Tabela *places*.

A descrição dos seus atributos é a seguinte:

- *id\_places*: dataIdentificador do place;
- *id\_user*: Identificador de utilizador;
- latitude: coordenada geográfica: latitude;
- longitude: Coordenada geográfica: longitude;
- nome: nome do Place;
- descrição: descrição do place;
- *data\_cri*: data de criação do place.

##### Tabela *users*

Contém a lista de utilizadores registados no FMP, como também o registo sobre o identificador de visita actual (*id\_user*), e alguns dados sobre a identidade de utilizador tais como o nick do utilizador (username), a palavra-passe (password), email (email), a data de registo no sistema FindMePlaces (*data\_reg*) e o estado.

<b>Campo</b>	<b>Tipo de dados</b>	<b>Características</b>
id_user	int(10)	primary key
id_visita	int(10)	foreign key
username	varchar(25)	-
password	varchar(25)	-
mail	varchar(25)	-
estado	varchar(25)	-
data_reg	data	-

**Tabela 3: Tabela Users.**

A descrição dos seus atributos é a seguinte:

- id\_user: identificador do utilizador;
- id\_visita: identificador da visita;
- username: login do utilizador;
- password: ppassword do utilizador;
- mail: email do utilizador;
- estado: estado da conta do utilizador;
- data\_reg: data de registo no sistema FindMePlaces.

#### Tabela *visits*

Na tabela Visits são armazenados todos os registos referentes aos check-ins.

<b>Campo</b>	<b>Tipo de dados</b>	<b>Características</b>
id_visita	int(10)	primary key
id_user	int(10)	foreign key
id_place	int(10)	foreign key
time_in	data	-
time_out	data	-
tipo_visita	data	-

**Tabela 4: Tabela visits.**

A descrição dos seus atributos é a seguinte:

- id\_visita - Identificador de visita;

- id\_user - identificador do utilizador;
- id\_place - identificador do *place*
- time\_in - tempo de *check-in*/entrada no *place*
- time\_out - tempo de check-out/saída do *place*.
- tipo\_visita - tipo de visita que poderá ser o *check-in* implícito, *check-in* explícito ou check-out.

### Tabela *publications*

A tabela Publications armazena os dados sobre os conteúdos submetidos ao sistema FMP pelo utilizador.

Campo	Tipo de dados	de Características
id_publicacao	int(10)	primary key
id_user	int(10)	foreign key
id_place	int(10)	foreign key
path_img	varchar(25)	-
post_text	varchar(25)	-
time_publication	data	-

**Tabela 5: Tabela *publications*.**

A descrição dos seus atributos é a seguinte:

- id\_publicacao: identificador da publicação;
- id\_user: identificador do utilizador;
- id\_place: Identificador do *place* onde foi feita a publicação;
- path\_img: path completo para a imagem do qual o utilizador fez upload;
- post\_text: comentário/post feito pelo utilizador;
- time\_publication: data e hora da publicação de comentário/post ou imagem/foto;

### Tabela *aps*

Esta tabela armazena os dados relativos a Pontos de Acessos. A tabela *aps* destina-se a relacionar um id do AP com o endereço mac (*mac\_addr\_ap*) e o nome da rede (*ssid\_ap*). Desta forma, podem-se adicionar ou remover os AP's de forma dinâmica, tornando o sistema mais flexível a novas adaptações.

<b>Campo</b>	<b>Tipo de dados</b>	<b>Características</b>
id_ap	int(10)	primary key
mac_addr_ap	varchar(25)	-
ssid_ap	varchar(25)	-

Tabela 6: Tabela aps.

A descrição dos seus atributos é a seguinte:

- id\_ap: identificador do ponto de acesso;
- mac\_addr\_ap: endereço mac do ponto de acesso;
- ssid\_ap: nome da rede.

#### Tabela *places/aps*

A tabela *places/aps* faz correspondência entre AP's e *places*.

<b>Campo</b>	<b>Tipo de dados</b>	<b>Características</b>
id_places_ap	int(10)	primary key
id_ap	int(10)	foreign key
id_place	int(10)	foreign key

Tabela 7: Tabela *Places/aps*.

A descrição dos seus atributos é a seguinte:

- id\_places\_ap - identificador que associa o um *place* ao ponto de acesso;
- id\_ap - identificador do ponto de acesso;
- id\_place - identificador do *place*.

### 4.3 PROTÓTIPO 1

A tecnologia utilizada para estimar a localização é baseada em recolha de informação sobre os pontos de acesso *Wi-Fi* próximos e o endereço IP do dispositivo de utilizador. A especificação W3C 07/2010 define uma API (11) que permite obter a informação de localização associada ao dispositivo de utilizador. Empregando esta tecnologia é possível embeber as funcionalidades de localização numa página Web.

Apenas os browsers mais recentes suportam esta API de localização acima referida: Firefox versão 3.5 ou superior, Opera versão 10.6, Google Chrome versão 5 ou

superior, Internet Explorer versão 9 superior;

A figura 6 descreve o processo de localização através de uma interface Web, recorrendo à API de um browser. Quando o utilizador abre uma página Web que invoca o serviço de localização, o browser pergunta se o utilizador deseja partilhar a sua localização. Se o utilizador permitir a utilização da informação de localização, o browser recolhe a informação sobre os pontos de acesso nas proximidades do utilizador, o endereço IP da sua máquina/seu dispositivo, e envia estes dados para o fornecedor de serviço de localização.

O serviço de localização estima a localização do utilizador na forma de uma par de coordenadas geográficas (latitude, longitude), a qual é devolvida ao cliente Web que iniciou o pedido.

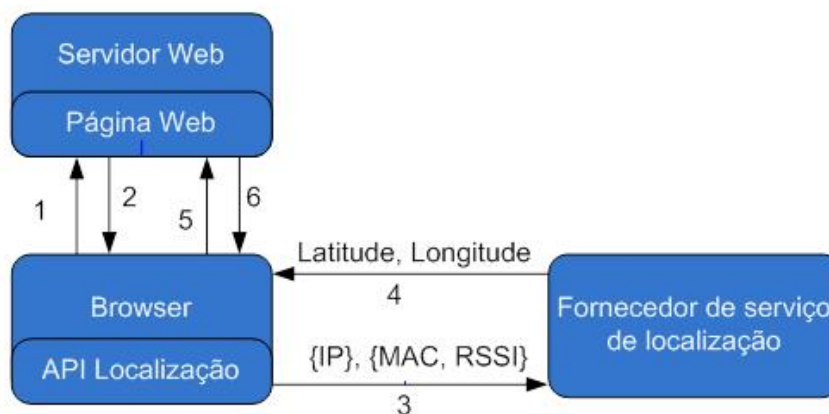


Figura 6: Funcionamento da API de localização de browser ou W3C.

A descrição detalhada do processo de autenticação de localização recorrendo à API do Browser resume-se nos seguintes passos:

Passo 1: O browser Web envia o pedido da página Web ao servidor Web.

Passo 2: Servidor Web disponibiliza a página Web inicial.

Passo 3: Com auxílio da API, o browser recolhe a informação sobre o endereço IP do dispositivo de utilizador, endereço MAC e a força do sinal dos pontos de acesso nas

proximidades de dispositivo de utilizador, e envia esta informação ao fornecedor de serviço de localização.

Passo 4: O fornecedor de serviço de localização estima a posição do utilizador através da informação proveniente do browser. A estimativa de localização, representada em coordenadas geográficas, latitude e longitude, é enviada para o browser, formando a posição do utilizador.

Passo 5: O browser pede ao servidor a actualização da página Web.

Passo 6: O servidor Web carrega a página Web.

No protótipo 1 o mapeamento das posições em *places* é feito como mostra esta figura, onde, apesar de na dimensão virtual conseguirmos referenciar um *place* com um par de coordenadas único, na dimensão real isso não é tão simples.

Na dimensão real as fronteiras de um local não correspondem a um ponto, mas sim a uma área, podendo um local englobar outros de menor dimensão ou sobrepor-se.

A estimativa de localização do utilizador é referenciada com coordenadas geográficas, latitude e longitude. Quando o utilizador que se encontra num espaço físico e cria um *place*, este é associado às coordenadas geográficas da sua localização actual. Ao efectuar o *check-in* implícito, a presença de utilizador ficará associada ao *place* com as coordenadas geográficas correspondentes à estimativa da sua localização, que pode já existir na base de dados ou ser criada pelo utilizador.

A lista dos *places* onde o utilizador poderá fazer o *check-in* explícito é constituída pelos *places* situados a uma determinada distância relativamente à sua localização actual. O

utilizador não poderá estar em mais do que um *place* ao mesmo tempo. Assume-se que cada *place* é referenciado por um par de coordenadas único. A figura 7 apresenta o esquema de mapeamento das posições em *places*.



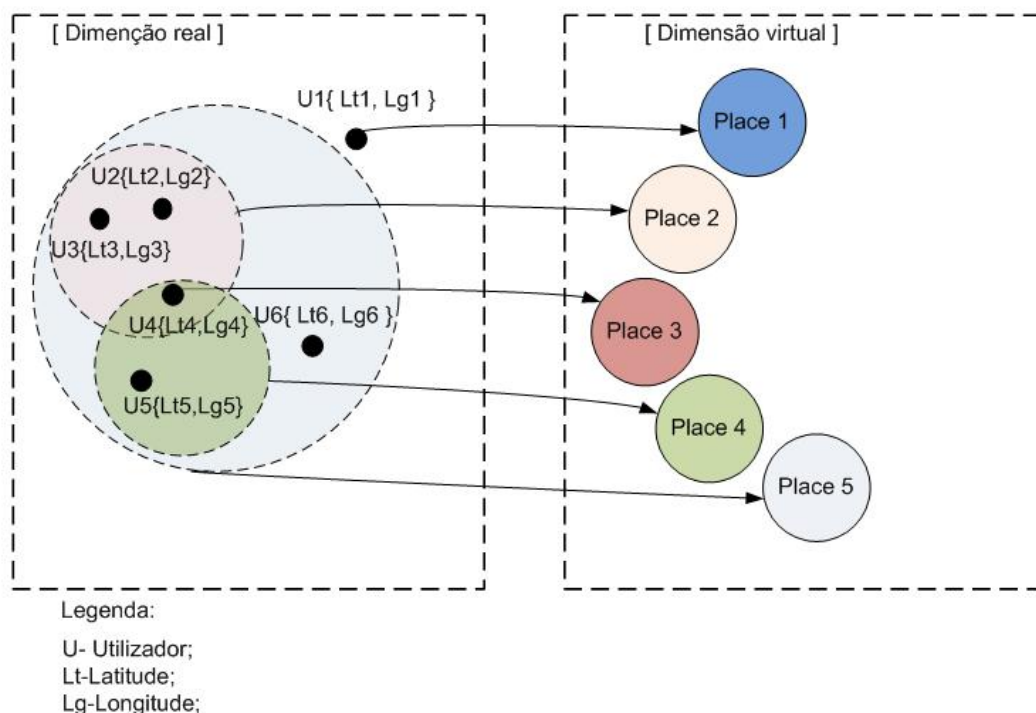


Figura 7: Protótipo Geográfico.

#### 4.4 PROTÓTIPO 2

A diferença do protótipo 2 relativamente ao protótipo 1, consiste na utilização de APIs de localização diferentes. Neste protótipo recorre-se a localização simbólica, podendo esta ser determinada com a informação relativa aos pontos de acesso *Wi-Fi* que difundem as redes ao alcance do dispositivo de utilizador.

A tecnologia *UniversalXPConnect* (14) permite aceder, a partir de um browser, à informação que descreve os AP *Wi-Fi* visíveis em cada instante, como o endereço Mac (*SSID*) e o nível de sinal (*RSSI*).

Mais especificamente, é necessário integrar na interface Web de FMP o script com a API da Gecko (13), que necessita de privilégios para ser executado do lado do cliente o *UniversalXPConnect*. Posteriormente, é possível aceder à API de localização do browser, que permite obter uma lista de pontos de acesso e as informações sobre eles, tais como endereço MAC (*SSID*) e o nível do sinal (*RSSI*).

A figura 8 apresenta o processo de localização efectuado através de uma interface Web recorrendo à API do browser. Quando o utilizador abre uma página Web que embebe a tecnologia *UniversalXPConnect*, deverá conceder autorização para que a API de localização seja executada no browser. No caso em que a autorização é dada, o browser

recolhe a informação sobre os pontos de acesso na vizinhança, enviando de seguida essas informações para o servidor de *places*. O servidor de *places* utiliza esta informação para determinar a lista de *places* associados aos pontos de acesso ou ainda para realizar o check-in, ou criar um novo *place*. Esta é a diferença fundamental entre o protótipo 1 e 2.

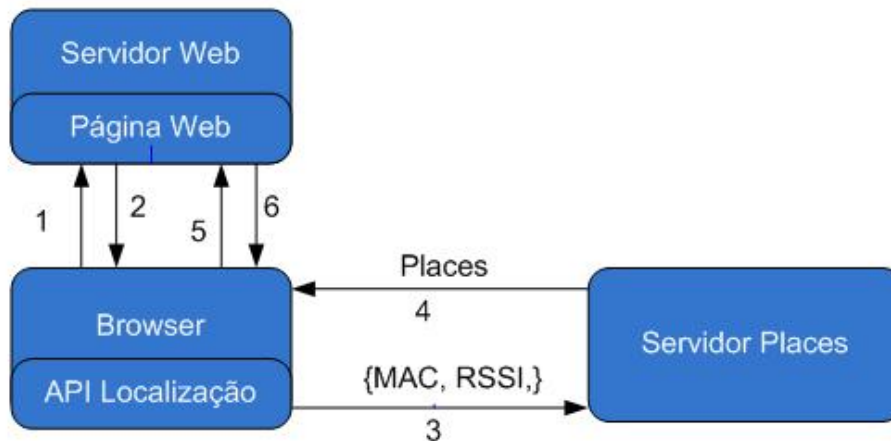


Figura 8: Funcionamento da API de localização no protótipo geográfico.

A descrição detalhada do processo de autenticação de localização recorrendo à API do Browser resume-se nos seguintes passos:

Passo 1: O browser Web envia o pedido da página Web ao servidor Web.

Passo 2: Servidor Web disponibiliza a página Web inicial.

Passo 3: Com auxílio da API, o browser recolha a informação sobre o endereço MAC e a força do sinal dos pontos de acesso nas proximidades de dispositivo de utilizador e envia esta informação para o servidor de *places*.

Passo 4: O servidor de *places* processa a informação proveniente do browser e envia o resultado para o browser.

Passo 5: O browser pede ao servidor actualização da página Web.

Passo 6: O servidor Web recarrega a página Web.

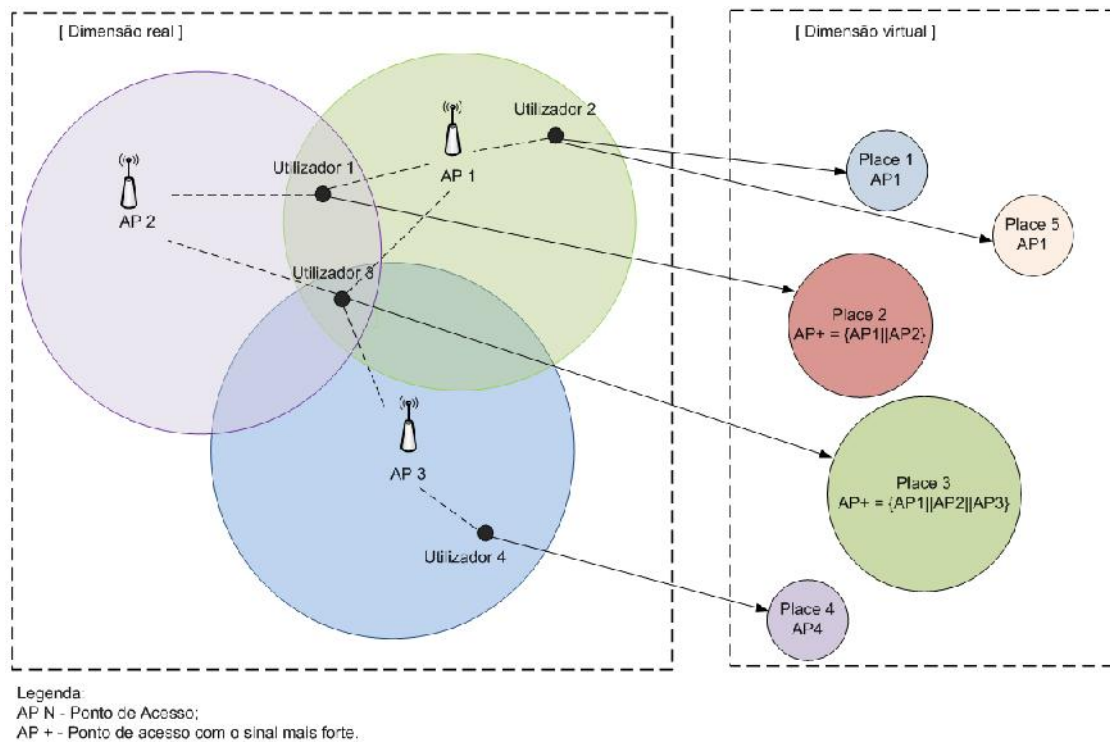


Figura 9: Mapeamento dos Places no protótipo 2.

Neste caso o utilizador poderá ter ao seu alcance uma ou mais redes locais sem fios difundidas pelos respectivos pontos de acesso. A figura 9 ilustra o mapeamento dos *places* numa área geográfica, para o caso protótipo 2. O utilizador 2 tem ao seu alcance a rede difundida pelo AP1. Se criar um *place* na dimensão virtual, este ficará associado ao AP1, o único AP na sua vizinhança. No caso de utilizador 3 e utilizador 1, existe mais do que uma rede disponível nas vizinhanças respectivas. Na criação de um *place* este ficará associado, na dimensão virtual, ao AP com o sinal mais forte.

#### 4.5 LIMITAÇÕES E DESAFIOS

Inicialmente, optou-se por implementar dois protótipos distintos que encaixam nos primeiros dois cenários.

A tecnologia utilizada para estimar a localização no Protótipo 1 é baseada em API W3C e é suportada pela maior parte dos browsers, nas suas versões mais recentes. A estimativa de localização é referenciada na forma de um par de coordenadas geográficas. No protótipo 2 é utilizada uma API de localização que recorre a localização

simbólica. A tecnologia *UniversalXPConnect* utilizada pela API de localização do protótipo 2 baseia-se na execução de um script do lado do cliente. Esta situação pode criar problemas de segurança, uma vez que o script fica com nível de privilégios elevado e pode comprometer a máquina do utilizador. Deste modo é necessário que o utilizador autorize explicitamente a execução do script e requer a configuração de certificado do lado do Servidor FMP. A API utilizada no protótipo 2 apenas pode ser executada no browser Mozilla Firefox e se existir o certificado válido configurado do lado do servidor. Devido a esta limitação técnica encontrada na utilização da API que recorre a localização simbólica, foi feita a junção dos dois protótipos. Desta forma, foi possível determinar a posição do utilizador de duas formas distintas através de mapeamento simbólico ou geográfico.

#### 4.6 INTEGRAÇÃO DO FMP COM UM MODELO DE ESPAÇO SIMBÓLICO.

A abordagem adaptada no FindMePlaces para sugestão dos nomes mais lógicos para os *places* criados pelo utilizador passa pela integração no FMP do modelo de espaços simbólico definido em (6).

O modelo de Espaço Simbólico propõe a representação de um modelo de espaço inspirando-se nos modelos mentais humanos produzidos a partir das suas experiências com a realidade. Trata-se de um modelo semântico que consiste em objectos, atributos de objectos, relações e atributos de relações.

A integração do sistema FindMePlaces com o modelo de espaço simbólico pretende:

- a) Objectivo 1: permitir que os utilizadores de sistema FindMePlaces contribuam para a construção de um modelo simbólico do espaço;
- b) Objectivo 2: utilizar o modelo simbólico na construção de um mecanismo de recomendação que auxilie os utilizadores de PBS FindMePlaces na criação de *Places* (contextualização do local em que estão a criar o *Place*, e ajuda à escolha de um nome adequado).
- c) Objectivo 3: Permitir uma maior facilidade de interacção entre utilizadores que acedam à informação dos mesmos *places*.

O Modelo de espaço simbólico é dividido em três módulos (6): SCBrowserWEB – é uma aplicação Web que permite navegar pelo modelo criado, usando a funcionalidade de pesquisa para encontrar um objecto e depois navegar pelo modelo através dos links embebidos nas descrições dos objectos.

Disponível através de URL<sup>9</sup>:

- O *Contextualizer Simbólico* é o serviço que permite a interacção com o modelo de espaço simbólico via interface HTTP. Através do Contextualizer é possível inserir novos objectos, inserir atributos de objectos, inserir relações, inserir atributos de relações, obter a descrição de um objecto dado um valor de um objecto ou o valor de um atributo, obter a descrição de um objecto dado o seu ID, etc.
- *Space Editor WEB*- é um editor que permite a inserção manual de objectos no modelo. Está disponível através de URL<sup>10</sup>. Para usar esta aplicação, o utilizador deve registar-se primeiro. Depois é só seguir as instruções para criar objectos, atributos de objectos, relações com outros objectos e atributos de relações.

Existe o fluxo de dados bidireccional entre o sistema FMP, o Servidor FMP, e o Contextualizer Simbólico. No sentido do Servidor FMP para o Contextualizer Simbólico, a ligação serve para criar os objectos que representem os *places* com os respectivos atributos, os objectos que representam os APs e as relações entre estes objectos.

No sentido do Contextualizer Simbólico para o Servidor FMP, serve para pesquisar os objectos (*places*) a partir do AP mais forte, derivando daí os *places* e outros objectos que estejam por perto.

#### 4.6.1 CRIAR O PLACE COM O AUXÍLIO DO MODELO DE ESPAÇO SIMBÓLICO E RESPECTIVA INTERFACE DE UTILIZADOR.

A integração com o *Modelo de Espaço Simbólico* permite ao FindMePlaces auxiliar o utilizador na escolha de um nome adequado para o *place* que está a criar.

Quando o utilizador selecciona a opção para criar um novo *place*, é redireccionado para a página “create new *place*”. O utilizador deverá preencher os campos *name* e *description*, tendo em conta o panorama geral dos *places* no local “Near You Places”. Esta lista é constituída pelos diversos *places* pertencentes à base de dados onde são armazenados os dados referentes aos *places* e objectos vindos do *Modelo de Espaço*

---

<sup>9</sup> <http://local.dsi.uminho.pt:8080/scb2/index.html>

<sup>10</sup> <http://local.dsi.uminho.pt:8080/scse2/index.html>

*Simbólico*. Cada círculo contém o nome do *place* já existente no local onde o utilizador se encontra, proporcionando, assim, ao utilizador, uma visão geral de espaço e do que já existe nele.

Após o preenchimento dos campos obrigatórios, *name* e *description*, e de seleccionar a opção “criar”, os dados são armazenados na base de dados FindMePlaces e é inserido um novo *place* com os seus atributos e relações no modelo de espaço simbólico.

#### 4.6.2 REPRESENTAÇÃO DE UM ESPAÇO FÍSICO EM FUNÇÃO DO MODELO DE ESPAÇO SIMBÓLICO.

Modelo de espaço simbólico - consiste numa base de dados relacional, contendo o modelo de espaço, e alguns algoritmos de raciocínio. O modelo simbólico é constituído por vários objectos que possuem atributos e interligações entre eles.

A Figura 10 ilustra a representação do espaço físico em função do modelo de espaço simbólico. Para tal, existem dois tipos de objectos:

- **Place**
  - O objecto **Place** possui dois atributos: *name* igual a nome do *place* e *type* igual a “*place*”;
- **AP**
  - O objecto **AP** tem dois atributos: *name* igual a MAC\_AP e *type* igual a “AP”;

Os objectos no Modelo de espaço simbólico são interligados através das relações que poderão ser de dois tipos:

- **Is\_Served\_by**
  - O sentido de relação é do objecto **Place** para o objecto **AP**;
  - Os atributos de relação são *name* e é igual a nome relação Is\_Served\_by e o *type* igual a *simple* ou em alternativa ao *symmetric*;
- **It\_Serves\_to**
  - O sentido de relação é do objecto **AP** para o **Place**;

- Os atributos de relação são *name* e é igual a nome de relação *It\_Serves\_to* e o *type* igual a *simple*;) )

Conforme os objectivos definidos na secção anterior, no que diz respeito ao objetivo1: no momento de inserção de objectos no modelo de espaço simbólico, em primeiro lugar é inserido o objecto AP e depois o objecto *Place*. Cada objecto *Place* é ancorado a um

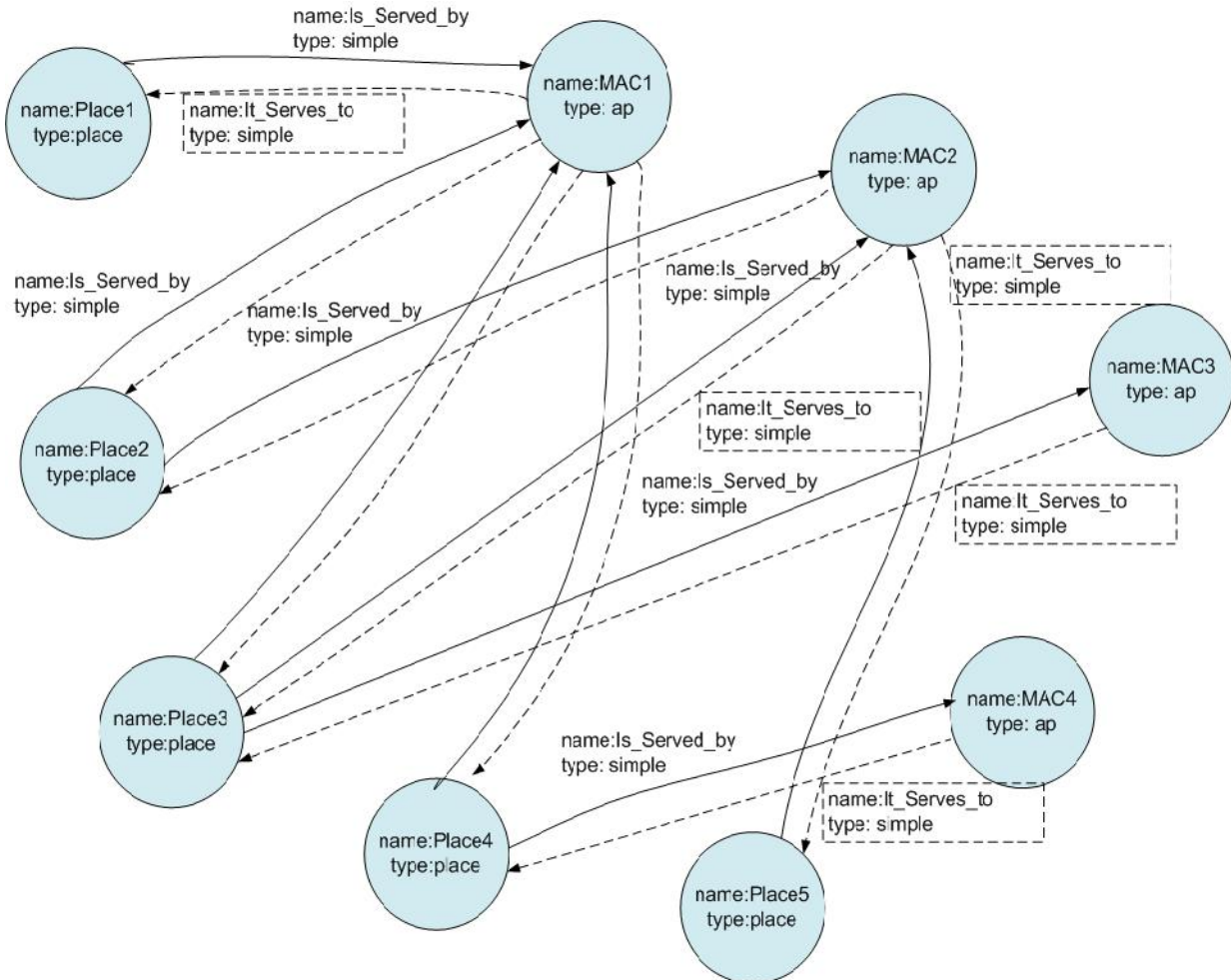


Figura 10: Representação de um espaço físico em dimensão de modelo simbólico.

ou vários objectos AP através de relação *Is\_Served\_by*. Cada objecto AP serve a um ou mais objectos do tipo *Place*. O *Place 1* é ancorado ao AP1 e o AP1, por sua vez, serve os *Place1*, *Place2*, *Place3* e *Place4*, e assim sucessivamente.

Relativamente ao objectivo 2: no momento de pesquisa de objectos do tipo *Place*, o único dado disponível é um dos endereços MAC de pontos de acessos no local onde se encontra o utilizador.

Esta pesquisa torna-se mais eficiente e rápida, visto que existe uma interligação bidireccional representada pelas relações *Is\_Served\_by* e *It\_Servers\_to*, apenas é necessária uma interacção para identificar os objectos do tipo *Place* relacionados com o objecto de tipo AP através da relação *It\_Servers\_to*.

O processo de procura torna-se mais rápido, visto que é possível identificar os *Places* relacionados com o AP através de relação *It\_Servers\_to*, enquanto o processo de inserção de novos *places* e os objectos relacionados torna-se mais lento, visto ser necessário inserir duas relações: *It\_Servers\_to* e *Is\_Served\_by* entre cada objecto *Place* e o objecto AP.

O caso mais complexo, onde se tem  $n$  *Places* ancorados a  $n$  AP, é de difícil implementação devido ao elevado número de interligações entre os *Places* e os APs existentes no modelo de espaço simbólico e à necessidade de pesquisa recorrente entre objectos. Este caso provoca uma certa ambiguidade na obtenção da lista dos *places* sugerida ao utilizador no momento de criação do novo *place*, uma vez que irão sugerir *places* de vários APs em que uns estarão mais próximos do que outros.

#### 4.7 INTEGRAÇÃO DO SISTEMA FINDMEPLACES COM O SISTEMA EPI

O sistema EPI (15) tem como objectivo proporcionar aos utilizadores de redes Wi-Fi a troca de mensagens de texto entre utilizadores, desde que se encontrem na mesma vizinhança física, mesmo sem conexão à Internet. As mensagens trocadas num determinado local, são armazenadas e novamente difundidas em outros locais, para onde o utilizador do sistema EPI se desloque.

A integração do sistema FindMePlaces com o sistema EPI pretende transformar a interface Web FindMePlaces num sistema de recolha de assinaturas rádio *Wi-Fi*.



O protótipo 2 do sistema FindMePlaces recorre à recolha da assinatura rádio *Wi-Fi* para efeitos de localização. Esta assinatura é utilizada directamente para detectar *Places* próximos ou para ancorar novos *Places*.

Para a integração com o sistema Epi, a sistema FindMePlaces (protótipo 2) deve submeter cada uma das assinaturas rádio que recolhe ao sistema Epi. Para isso devem usar a interface HTTP descrita no Anexo 1.

Estão previstos dois tipos de interacção que serão referidos como função f1. A descrição desta pode ser consultada em Anexo 1. A função f1 destina-se a permitir ao sistema FindMePlaces enviar ao Servidor Central do Epi os dados relativos a cada uma das assinaturas rádio recolhidas. A ligação unidireccional entre o Servidor e o Sistema Epi serve para submeter as assinaturas rádio sempre que forem recebidos pelo sistema FindMePlaces.



## 5 DESENVOLVIMENTO E IMPLEMENTAÇÃO DO SISTEMA FMP

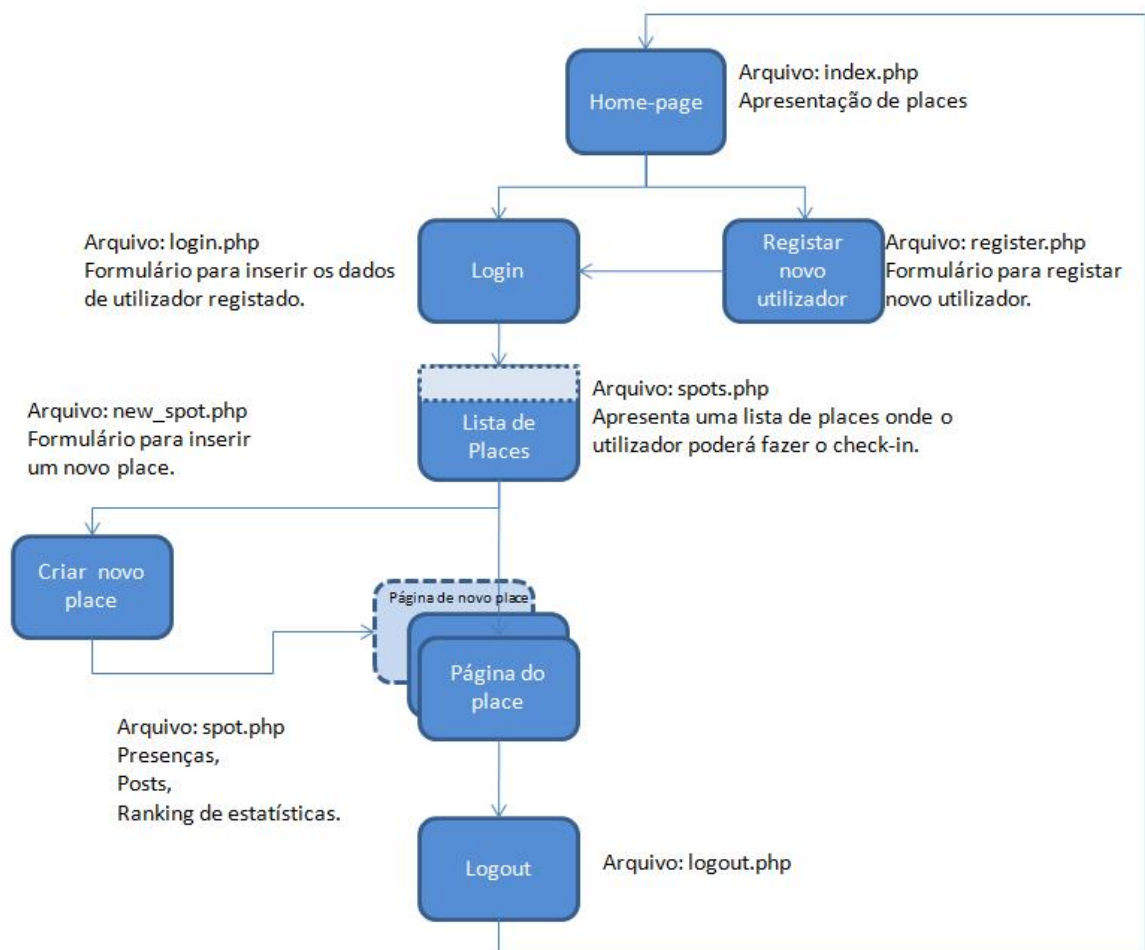
Após a apresentação da especificação de plataformas de suporte aos places, neste capítulo descreve-se todo o processo de desenvolvimento e implementação do PBS FindMePlaces.

### 5.1 INTERFACE WEB FMP

A interface Web FMP possui um *layout* e um esquema de navegação bastante simples. Esta interface é responsável por transferir o fluxo de informação recebido do utilizador para o servidor FMP e vice-versa. A interface Web é constituída por um conjunto de páginas Web descritas nesta secção.

#### 5.1.1 MAPA DE NAVEGAÇÃO

Inicialmente, o utilizador encontrar-se-á na homepage do site, onde lhe é possível visualizar a informação geral sobre os *places*, assim como juntar-se a estes e registar-se ou efectuar o login. Para o utilizador efectuar a acção de login, este terá de ter a priori uma conta criada. No caso de o utilizador estar autenticado, este poderá navegar para a sua página pessoal ou ainda visualizar a lista dos *places* na sua vizinhança. O utilizador poderá efectuar *check-in* em qualquer um dos *places* da lista. No caso de a lista não conter o *place* pretendido, é possível criar um novo *place* navegando para formulário “Novo Place”. Depois de criar um novo *place*, o utilizador é redireccionado de volta para a página que contém a lista dos *places* actualizada, onde poderá efectuar o *check-in*. Após o *check-in*, a presença de utilizador é registada no *place*.



Legenda:






-  Página normal
-  Mais de uma página do mesmo tipo.
-  Link de ida
-  Link de ida e volta
-  Janela pop-up (abre-se uma nova janela no nível superior da página).

Figura 11: Mapa de navegação e a respectiva legenda.

A interface Web do PBS FMP é estruturada segundo o mapa de navegação apresentado na figura 11. Este diagrama mostra as várias páginas que compõe a interface Web FMP, detalha as ligações entre as páginas, o seu conteúdo e as várias formas de navegar entre elas.

## 5.1.2 INTERFACE COM O UTILIZADOR

### HOME PAGE

A página inicial é bastante simples e contém informação geral sobre o FMP. A barra de navegação horizontal permite navegar entre os separadores: “About” relativo ao projecto FMP, “FAQ” com questões frequentes e respostas e os “Contact us”. A página inicial apresentada ao utilizador é ilustrada na figura 12.

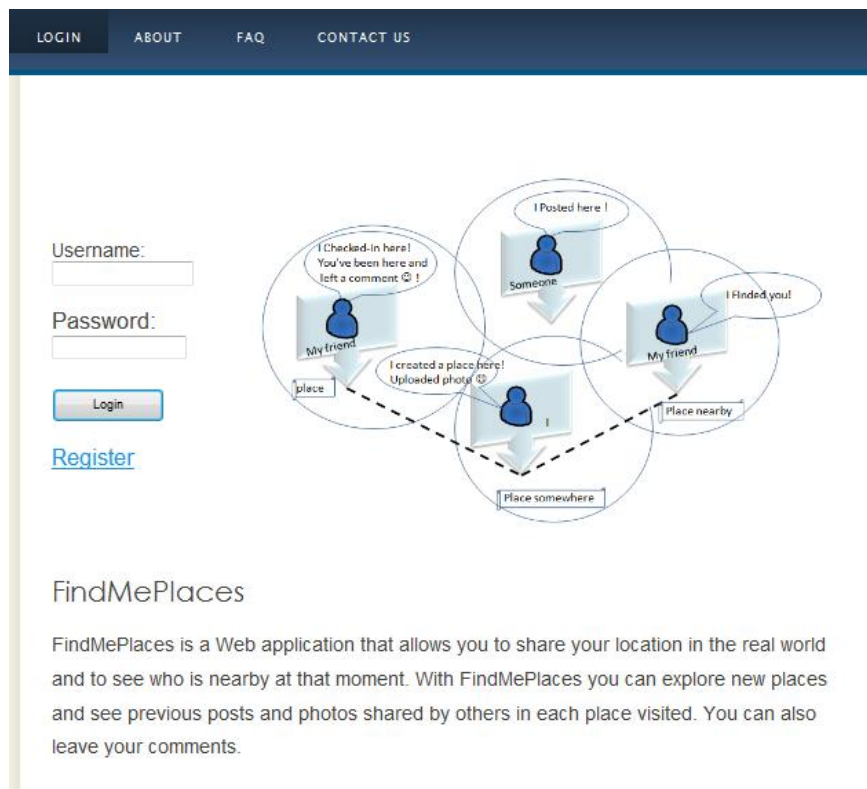


Figura 12: Interface de login no sistema FindMePlaces.

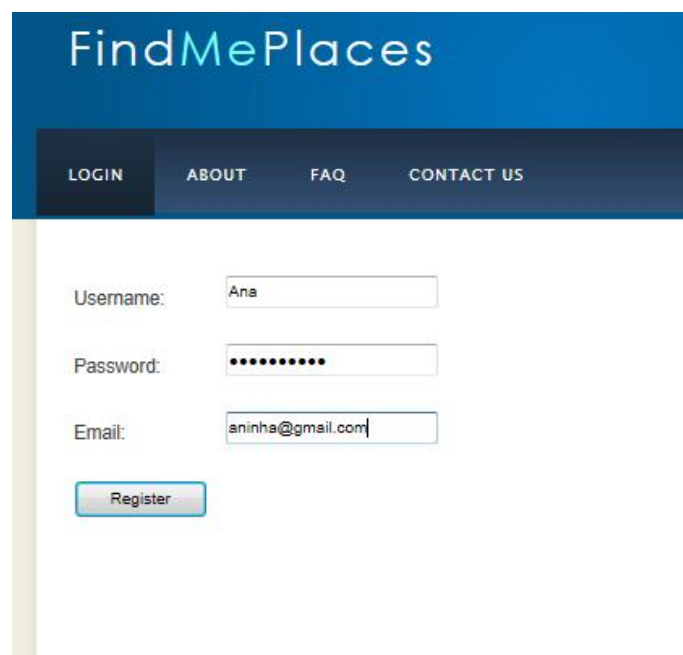
### LOGIN

Na Home page, o utilizador pode efectuar o login, introduzindo as suas credenciais de autenticação. Se o utilizador for autenticado correctamente, passa a ter acesso a outras funcionalidades do sistema FindMePlaces. Se não o for, é lançada uma mensagem de alerta a indicar eventuais erros na introdução de dados ou a falta dos mesmos. Se for este o caso, o utilizador poderá introduzir novamente as credenciais ou, em alternativa, registar-se na aplicação.

## REGISTO DE UM NOVO UTILIZADOR

---

Na página de registo, o utilizador deverá escolher o username, password e inserir um endereço de email válido. A validação dos dados introduzidos no formulário pelo utilizador é feita do lado do cliente/browser, no código da página web. Após validação, os dados são transferidos para o Servidor FMP. Os métodos que permitem efectuar o registo de um novo utilizador são executados no ficheiro register.php. A figura 13 ilustra o formulário de registo de um novo utilizador.



The image shows a web browser window displaying the registration page for 'FindMePlaces'. The page has a blue header with the site name 'FindMePlaces' in white. Below the header is a dark blue navigation bar with links for 'LOGIN', 'ABOUT', 'FAQ', and 'CONTACT US'. The main content area is white and contains a registration form with three input fields: 'Username' (containing 'Ana'), 'Password' (masked with dots), and 'Email' (containing 'aninha@gmail.com'). Below the fields is a blue 'Register' button.

Figura 13: Registo de um novo utilizador.

## LISTA DE PLACES

---

Após ter efectuado o login com sucesso, o utilizador é encaminhado para a página dos *places*, sendo esta constituída por 3 áreas. A barra de navegação horizontal que permite efectuar o logout ou criar novo *place*; Abaixo da barra de navegação horizontal encontra-se uma área personalizada conforme o perfil de cada utilizador e a seguir, inicia-se a área com a lista de *places*.

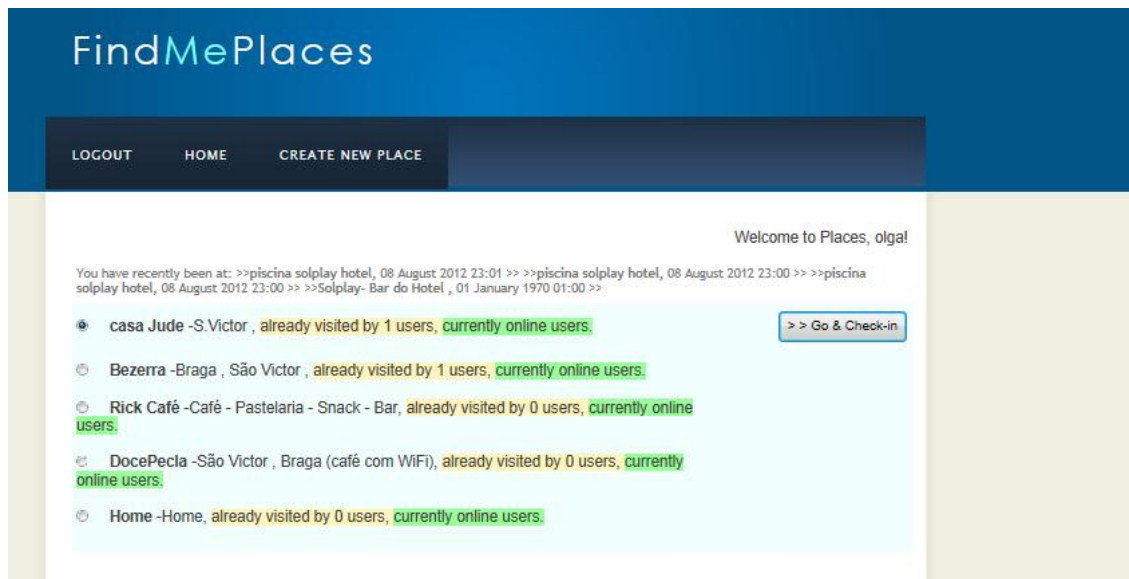


Figura 14: Apresentação da lista dos places.

- Detecção de localização

Quando o utilizador é redireccionado para a página dos *places* é feita a detecção da sua localização e é lançada uma janela pop up a pedir permissão para partilhar a localização.

- Construção/Ordenação da lista dos *places*

A informação sobre a localização física do utilizador é enviada para o Servidor Web. O servidor Web constrói a lista dos *places* com base no resultado extraído da pesquisa sobre a localização, na base de dados. Os *places* são apresentados ao utilizador em lista ordenada de forma a que os *places* mais próximos fiquem em primeiro lugar.

- Expandir a lista de *places*

A lista de *places*, inicialmente, apresenta os *places* correspondentes a espaços físicos a uma distância até 1 km da localização do utilizador. Seleccionando a opção “*more*”, são acrescentados à lista os *places* à distância de até 2 km.

- *Check-in* implícito

De forma transparente para o utilizador, o sistema regista o *check-in* implícito no primeiro *place* da lista, ou seja, o que se situa mais perto da localização física do

utilizador.

- *Check-in* explícito

Escolhendo o *place* da lista dos *places* e carregando no botão “Go&Check-in”, o utilizador faz o *check-in* explícito, entrando dentro de um *place* virtual.

- Apresentação de lista dos *places*

A lista dos *places* é construída por vários itens - *places*. Cada item da lista apresenta a informação sobre o *place*, tal como o nome do *place* e a sua descrição, o número total de utilizadores que já visitaram o *place*, (*check-in* explícito) e o número de utilizadores online no *place* naquele momento.

- Mensagem personalizada

Na parte esquerda da área personalizada ao perfil de utilizador é apresentada uma mensagem de Boas-vindas.

- Histórico de *check-ins*

Uma outra funcionalidade na área personalizada ao perfil de utilizador é o histórico de *check-ins*. É um tipo de lista com os últimos 4 *check-ins* explícitos realizados pelo utilizador. O *check-in* é descrito pelo nome do *place*, a sua descrição e a data e hora do *check-in*.

### NOVO PLACE

---

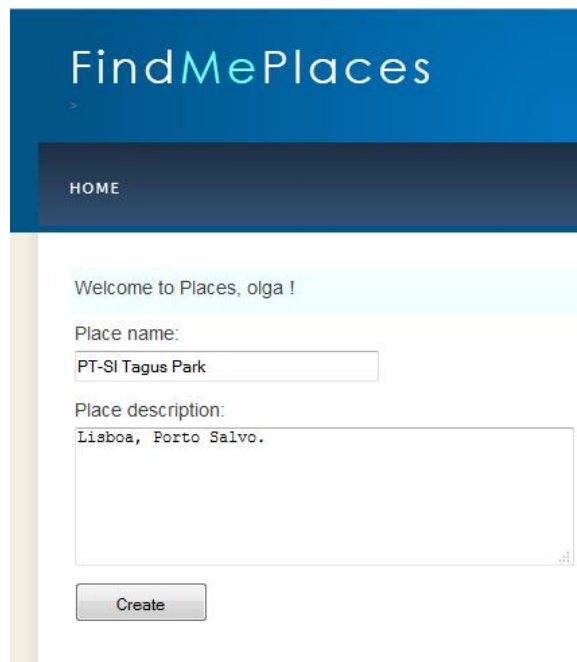
Esta página, com a mesma estrutura de outras, é dividida em 3 áreas. A barra de navegação horizontal permite voltar à página inicial (home), ou seja navegar até a lista dos *places*. Se, por algum motivo, o utilizador não concluir o processo de criação do novo *place*, a área personalizada ao perfil de utilizador apresenta a mensagem de boas vindas personalizada e o formulário para a criação do *place*.

- Criar novo *place*

Nesta página o utilizador deverá introduzir na primeira *text box* o nome do *place* que pretende criar e na outra *text box* a sua descrição. Ambas as áreas são limitadas a um determinado número de caracteres. O processo de validação neste formulário passa apenas por não deixar criar o *place* se o campo descrição ou nome do *place* se encontrarem por preencher. Carregando na opção “create” é lançada uma mensagem



que indica se a operação de criação de um novo *place* correu com sucesso, sendo o utilizador de seguida encaminhado para a Home (lista de *places*).



The screenshot shows the 'FindMePlaces' web application. At the top, there is a blue header with the text 'FindMePlaces' in white. Below the header is a dark blue navigation bar with the word 'HOME' in white. The main content area has a light green background and contains a welcome message: 'Welcome to Places, olga !'. Below this, there are two input fields: 'Place name:' with the text 'PT-SI Tagus Park' and 'Place description:' with the text 'Lisboa, Porto Salvo.'. At the bottom of the form is a 'Create' button.

Figura 15: Interface para criação de um novo place.

### PLACE

A página do *place* é constituída por uma barra de navegação horizontal, na qual apenas existe a opção de “*check-out*” que permite ao utilizador voltar para a lista de *places*. É também constituída por uma área personalizada de utilizador com a mensagem personalizada de boas vindas, uma área de conteúdos e, do lado direito, uma área de âmbito do utilizador.

### PUBLICAÇÕES

Na área de conteúdos, o utilizador pode publicar as fotografias ou comentários sobre o *place* onde fez o *check-in* e a sua actividade, deixar sugestões para os outros utilizadores que provavelmente por lá passarão, ou qualquer tipo de conteúdo que possa ser transmitido por texto e/ou fotografia.

- Apresentação e listagem das publicações

Na área de conteúdos é apresentada uma lista de fotografias em miniatura, relativas ao *place*, e duas colunas: uma de comentários e outra de fotografias, respectivamente. Os conteúdos das colunas são organizados de forma decrescente, de publicação mais recente para a mais antiga, sendo inicialmente apresentadas as últimas 20. Carregando na opção “*more img/comments*” é possível expandir as duas colunas e visualizar as fotografias e comentários mais antigos.

- Divulgação de publicações nas redes sociais

As publicações podem ser divulgadas através da sua partilha nas redes sociais. Para tal, o utilizador deverá escolher um dos ícons apresentados no widget, na área de âmbito de utilizador.

- Carregamento de fotografia

O utilizador pode escolher uma imagem e fazer o seu carregamento para o sistema. Depois do utilizador carregar a imagem, esta é adicionada à fila de imagens na parte superior da área de conteúdos. No carregamento de fotografias é efectuada a validação sobre o tipo de ficheiro que será carregado.

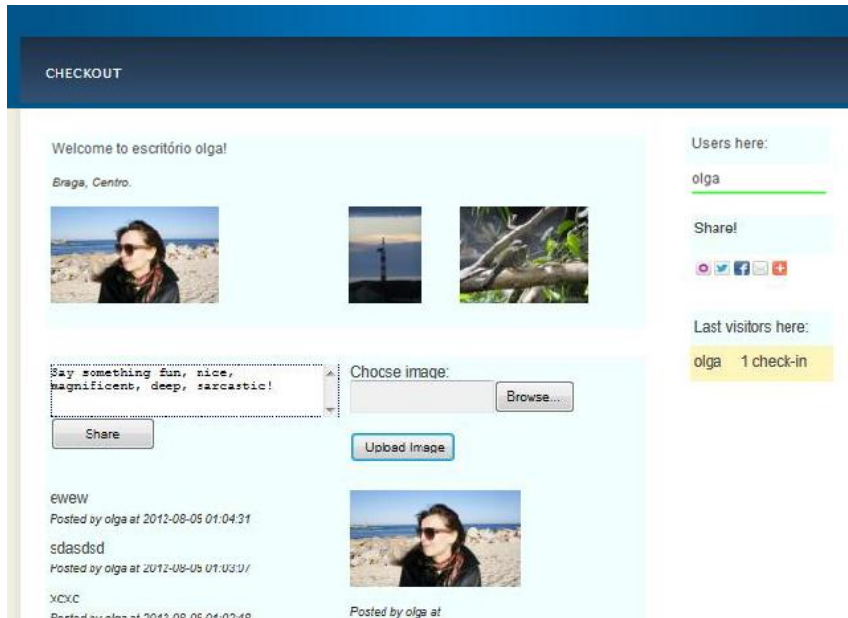


Figura 16: Página do place.

- Publicação de comentários

Através da *text box*, na área de conteúdos, é possível ao utilizador deixar um comentário, e partilhá-lo. Na área de texto não é possível partilhar um comentário vazio, sendo esta a única validação efectuada na submissão do mesmo.

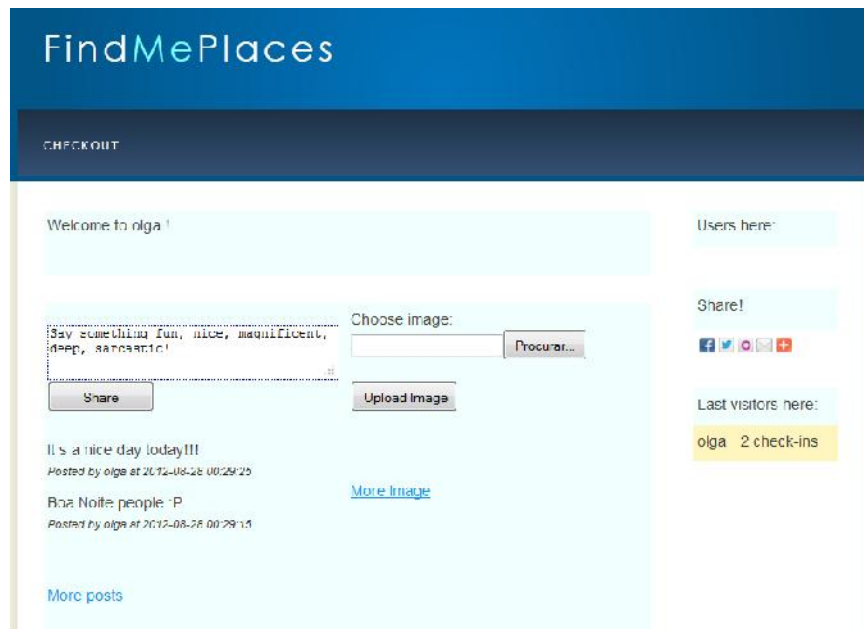


Figura 17: Interface para partilha de conteúdos e lista de presença.

### LISTA DE PRESENÇA

A lista de presença encontra-se na área de âmbito de utilizador. Esta lista apresenta os utilizadores que se encontram online (linha verde) ou ausentes (linha laranja) dentro de um *place*, num determinado momento.

### HISTÓRICO DE VISITAS

O histórico de visitas é uma lista que apresenta os nomes dos últimos utilizadores que efectuaram o *check-in* no *place*, como também o número de *check-ins* efectuados por cada um deles.

## 5.2 SERVIDOR FMP

No servidor FMP podemos encontrar as configurações necessárias ao servidor Web Apache, assim com as configurações de certificados essenciais à autenticação de identidade.

### 5.3 SERVIDOR BD FMP

O servidor BD FMP foi desenvolvido em WAMP, onde se encontra incluído o servidor web Apache e o sistema de gestão de bases de dados relacionais MySQL. A Base de Dados foi implementada com o auxílio da linguagem SQL (Structured Query Language – Linguagem de Consulta Estruturada), sendo utilizada tanto para consultas como para inserção de novos dados.

### 5.4 INTERACÇÃO ENTRE O SISTEMA FINDMEPLACES E MODELO DE ESPAÇO SIMBÓLICO.

A comunicação existente entre o sistema FMP e o modelo de espaço simbólico é realizada sempre entre o servidor FMP e o Contextualizer simbólico via pedidos *HTTP*. A iniciativa de comunicação parte sempre do servidor FMP, onde este envia as mensagens *HTTP Request* descritas no Anexo 1. No caso de insucesso, o *HTTP Response* devolvido pelo Contextualizer simbólico para o Servidor FMP será um dos códigos de erros respectivo a cada uma das funções (Anexo 1).

O servidor FMP e o Contextualizer simbólico interagem em dois casos, sendo estes:

- a) Criar um novo *place* com auxílio do modelo de espaço simbólico.
- b) Inserir objectos no modelo simbólico sempre que o utilizador acaba de criar um *place*.

#### 5.4.1 CRIAR UM NOVO PLACE COM AUXÍLIO DO MODELO DE ESPAÇO SIMBÓLICO.

Neste caso, quando o utilizador pretende criar um novo *place*, o servidor FMP recorre ao modelo de espaço simbólico para construir um panorama geral de *places* existentes no local onde o utilizador se encontra. Após autenticar-se na página de FMP, é apresentada uma lista de locais ao utilizador. Caso esta lista não contenha o *place* onde o utilizador pretende fazer o *check-in*, este poderá criar um novo *place* seleccionando a opção *create new place*. De seguida, o utilizador é redireccionado para o formulário correspondente. Durante este redireccionamento o servidor FMP deverá verificar na base de dados FMP e no Modelo de espaço simbólico se existem *places* ancorados aos APs, visíveis no local, onde o utilizador se encontra. Na página *create new place* apresenta-se o panorama geral de *places* próximos do local onde o utilizador pretende

criar o novo *place*. Com auxílio desta visão, o utilizador deverá preencher os campos obrigatórios *name* e *description* e seleccionar opção *create* para finalizar a operação.

A troca de mensagens entre o Utilizador que acede à Interface Web FindMePlaces, Servidor FindMePlaces e o contextualizar simbólico está retratada na figura 18.

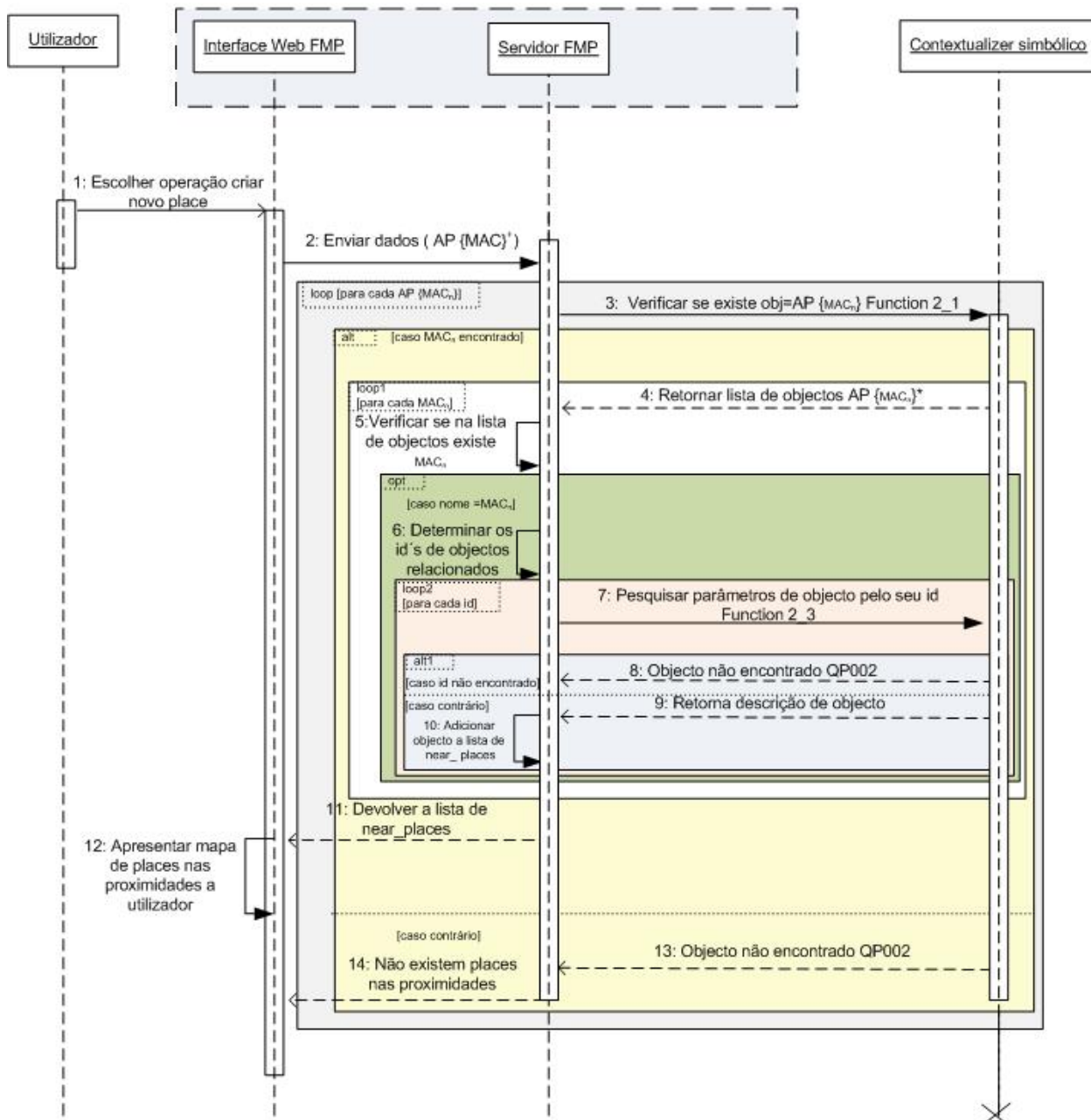


Figura 18: Diagrama de seqüência: criar um novo place com auxílio de modelo de espaço simbólico.

Mensagens:

- 1.O utilizador escolhe a opção criar novo *place*.
2. A interface Web FMP envia uma lista de endereços MAC, de pontos de acesso para o ServidorFMP.
- 3.O ServidorFMP determina o AP com o sinal mais forte (RSSI mais alto).
- 4.O Servidor FMP envia um pedido ao Contextualizer simbólico para verificar se já existe algum objecto com campo “nome” igual ao endereço MAC do ponto de acesso n. (Function 2\_1: verifica a existência de um objecto pelo seu nome (String).
- 5.alt[caso MACn encontrado], O modelo de espaço simbólico retorna a lista de objectos que contem o endereço MAC no campo “nome” ao Servidor FMP.
- 6.O servidor FMP procura na lista o objecto cujo nome corresponde a endereço MAC.
7. Opt[caso objname=MACn], O servidor FMP determina os id’s dos objectos relacionados.
- 8.Loop2: [para cada id] O servidor FMP envia pedido ao Modelo de espaço simbólico (function 2\_3 devolve os parâmetros de objecto pelo seu id.).
- 9.Alt1 [caso id não encontrado] é devolvida a mensagem [QP002- objecto não encontrado] para o Servidor FMP.
- 10.[caso id encontrado] retorna a descrição de objecto ao ServidorFMP.
- 11.O ServidorFMP adiciona o objecto à lista de objectos *near\_places*.
- 12.Devolver lista de *near\_places*.
13. Apresentar ao utilizador o mapa dos *places* nas proximidades.
- 14.Caso o objecto não seja encontrado no modelo de espaço simbólico, é enviada mensagem [QP002- objecto não encontrado] para o Servidor FMP.
- 15.O servidor FMP envia uma mensagem de retorno para a interface Web FMP com informação que não existem *places* nas proximidades [FMP001-não existem sugestões de nomes para *places*]; opcional.
- 16.Em todas estas operações, existe a possibilidade de ocorrer um time-out ou latência ou seja, sempre que há uma comunicação entre os servidores em ambos os sentidos. Se isso ocorrer, é importante que o utilizador não fique pendurado no meio desta comunicação.

#### 5.4.2 INSERIR OBJECTOS NO MODELO DE ESPAÇO SIMBÓLICO SEMPRE QUE O UTILIZADOR ACABE DE CRIAR UM PLACE.

Após o utilizador finalizar a operação *create new place*, os dados do *place* são armazenados na base de dados, sendo estes: o nome do *place*, a sua descrição e o endereço MAC de AP com RSSI mais forte. O servidor FMP envia um pedido ao contextualizer simbólico para inserir um novo objecto de tipo AP, enviando de seguida um pedido para inserir o objecto de tipo *Place*, verificando sempre se os objectos que pretende inserir já existem no modelo de espaço simbólico. O passo seguinte consiste em criar duas relações *Is\_server\_by* entre o *Place* e o AP e *It\_server\_to* entre o AP e o *Place*.

Quando é concluída a operação *create new place* o servidor FMP efectua uma troca de mensagens necessárias com o Contextualizer simbólico para acrescentar os objectos no modelo de espaço simbólico. Em primeiro lugar, o Servidor FMP envia pedidos ao Contextualizer Simbólico para que sejam inseridos os objectos do tipo AP e do tipo *Place*. De seguida são enviados pedidos para inserção das ligações *Is\_Served\_by* e *It\_serves\_to*. A interacção pormenorizada entre o utilizador que acede à Interface Web FindMePlaces, Servidor FindMePlaces e o Contextualizer Simbólico está retratada na figura 19.

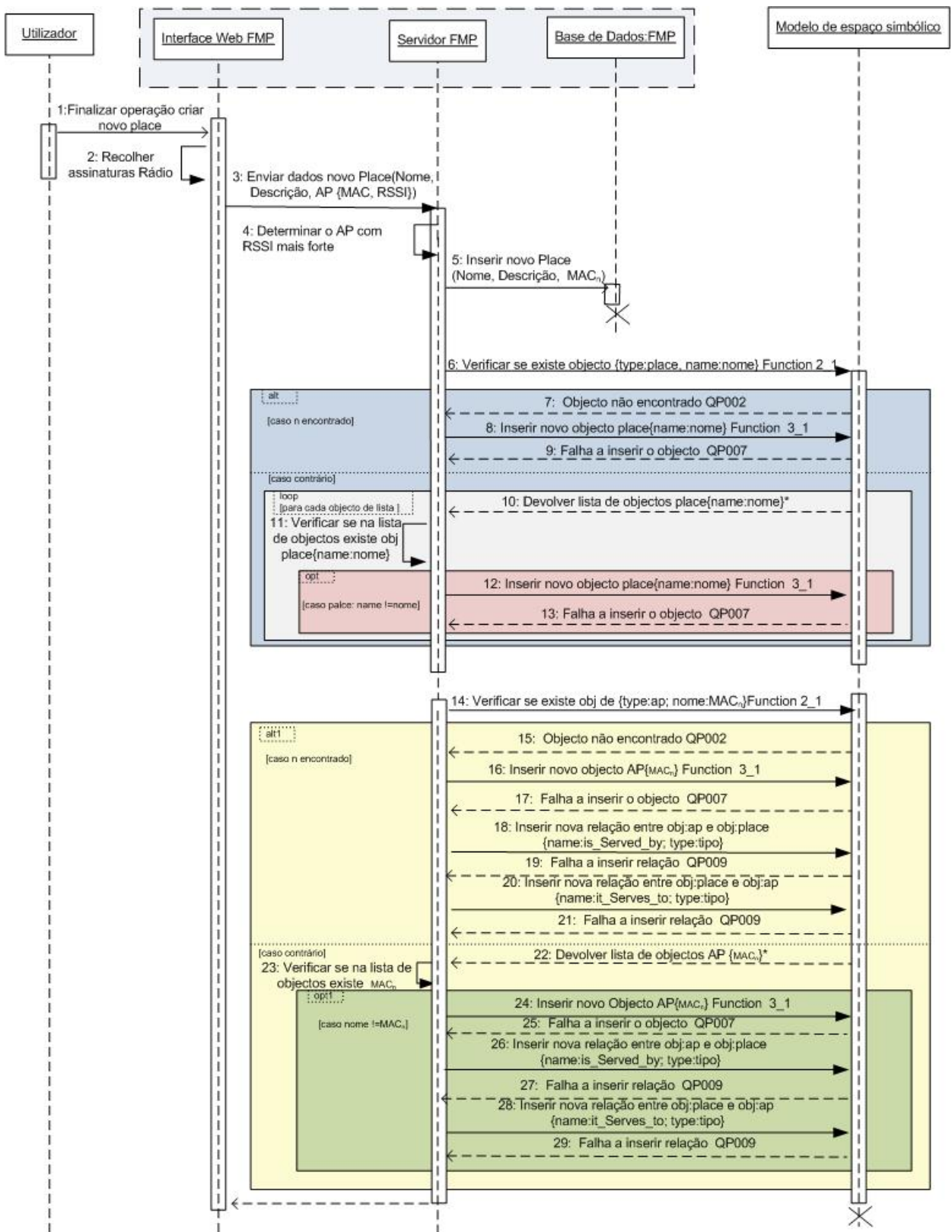


Figura 19: Diagrama de sequência: inserção de objectos no modelo de espaço simbólico sempre que o utilizador acabe de criar um place.



## Mensagens:

- 1-O utilizador finaliza a operação de criar um novo *place*.
- 2-A InterfaceWebFMP recolhe a assinatura rádio.
- 3- A InterfaceWebFMP envia os dados relativos ao novo *place*: o nome, a descrição, a lista de endereços MAC dos APs e valores de RSSI correspondentes.
- 4-O Servidor FMP determina o AP com o sinal mais forte.
- 5-O ServidorFMP insere o novo *place* na base de dados FMP.
- 6-O Servidor FMP envia para à base de dados o pedido de inserção de um novo *place*, representado pelo nome de *place*, descrição, endereço MAC do Ponto de Acesso com o RSSI mais forte.
- 7-O Servidor FMP envia um pedido ao WSContextualizer para verificar se já existem os objectos com campo nome que contem o endereço MAC do ponto de acesso n. (Function 2\_1: verifica existência de um objecto pelo seu nome (String)).
- 8-Alt[caso não encontrado] caso o objecto não exista, é devolvida a mensagem [QP002-objecto não encontrado] para o Servidor FMP.
- 9-O Servidor FMP envia o pedido de inserção de novo objecto place com parâmetros: name:nome e type:palce. (Function 3\_1).
- 10-Caso a inserção não tenha sucesso, é devolvida a mensagem [QP007-falha a inserir o objecto] ao servidorFMP.
- 11-Alt [caso contrario] Se já existirem objectos com este nome, devolve a lista dos places.
- 12-O servidorFMP verifica se na lista dos places já existe o *place* com o name=nome.
- 13-Se não existir, envia pedido de inserção de novo objecto *place* com parâmetros: name:nome e type:place. (Function 3\_1) para o modelo de espaço simbólico.
- No caso de falha na inserção, é devolvida a mensagem OP007- falha a inserir o objecto.
- 14-Loop1 [para cada AP {MACn}] O Servidor FMP envia um pedido ao Contextualizer simbólico para verificar se já existem objectos com campo nome que contenham o endereço Mac do ponto de acesso n. (Function 2\_1: verifica existência de um objecto pelo seu nome (String)).
- 15.Alt1 [caso não encontrado] caso o objecto não exista, é devolvida a mensagem [QP002- objecto não encontrado] para o Servidor FMP.

- 16-Caso não exista, envia pedido de inserção de novo objecto AP com parâmetros: name: MAC<sub>n</sub>; type:ap (Function 3\_1) para o modelo de espaço simbólico.
- 17.No caso de falha na inserção, é devolvida a mensagem OP007- falha a inserir o objecto.
- 18.O Servidor FMP envia pedido de inserção de uma nova relação com parâmetros: {name:is\_Served\_by; type:tipo} obj= objc=*place*{name:nome} -> AP{MAC<sub>n</sub>}
- 19.No caso de falha na inserção, é devolvida a mensagem QP009- falha a inserir relação.
- 20.Alt1[caso contrario],loop2[para cada MAC<sub>n</sub>] O Contextualizer simbólico devolve a lista de objectos com campo nome que contém o endereço MAC do ponto de acesso n.
- 21.O servidorFMP verifica se na lista dos places já existe o AP com o endereço nome=MAC<sub>n</sub>.
- 22.Opt1[caso nome!=MAC<sub>n</sub>] Caso não exista, envia pedido de inserção de novo objecto AP com parâmetros: name: MAC<sub>n</sub>; type:ap (Function 3\_1) para o modelo de espaço simbólico.
- 23.No caso de falha na inserção, é devolvida a mensagem OP007- falha a inserir o objecto.
- 24.O Servidor FMP envia pedido de inserção de uma nova relação com parâmetros: {name:is\_Served\_by; type:tipo} obj= objc=*place*{name:nome} -> AP{MAC<sub>n</sub>}
- 25.No caso de falha na inserção é devolvida a mensagem QP009- falha a inserir relação.

## 5.5 MODO DE INTEGRAÇÃO DO SISTEMA FINDMEPLACES COM O SISTEMA EPI.

### 5.5.1 SUBMISSÃO DE ASSINATURAS DE RÁDIO AO SISTEMA EPI.

A comunicação existente entre o sistema FMP e o servidor EPI é realizada sempre entre o servidor FMP e o Servidor EPI, via pedidos *HTTP*. A iniciativa de comunicação parte sempre do servidor FMP, e este envia mensagens *HTTP Request* descritas no Anexo1. No caso de insucesso, o *HTTP Response* devolvido pelo Servidor EPI para o Servidor FMP será um dos códigos de erros relativos a cada uma das situações.

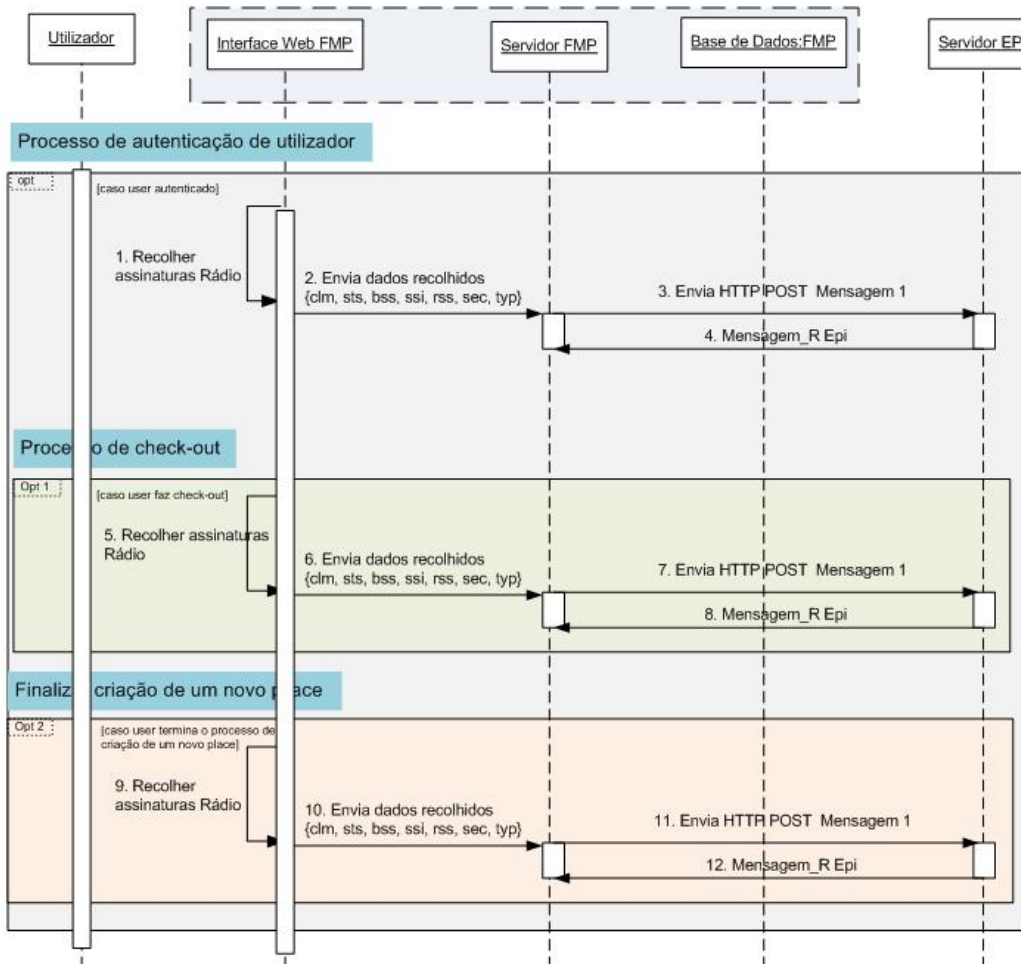


Figura 20: Diagrama de sequência: Submissão de assinaturas de rádio ao sistema EPI.

## 5.6 TESTES DE INTEGRAÇÃO DE FUNCIONALIDADES DO SISTEMA

### FINDMEPLACES EM AMBIENTE REAL

Este secção descreve a metodologia empregue para a divulgação do sistema FMP, bem como os resultados estatísticos obtidos da monitorização da integração do sistema FindMePlaces em ambiente real.

#### 5.6.1 METODOLOGIA DE DIVULGAÇÃO DO SISTEMA FMP

Com finalidade de promover a utilização de sistema FindMePlaces, foi feita a sua divulgação através de diferentes meios:

- Criou-se a página de FindMePlaces <sup>11</sup> no Facebook para dar a conhecer o sistema FindMePlaces e para que seja mais fácil aos utilizadores partilharem as suas experiências e colocarem questões.
- Na interface web da aplicação <sup>12</sup> FMP é possível ler alguma informação sobre o sistema FindMePlaces, consultar a FAQ e conhecer um pouco mais sobre este projecto.
- Foi enviado um e-mail para a comunidade académica, com o intuito de divulgar e incentivar o uso do FindMePlaces.

#### 5.6.2 ESTATÍSTICAS DE MONITORIZAÇÃO DO SISTEMA FMP

Durante um período de aproximadamente 3 meses, foram efectuadas algumas monitorizações ao sistema FMP. Os resultados obtidos são apresentados nas secções seguintes. Os gráficos das monitorizações foram obtidos com auxílio da ferramenta *jpgraph*<sup>13</sup>

---

<sup>11</sup> <https://www.facebook.com/pages/FindMePlaces/117005225051445>

<sup>12</sup> <http://findmeplaces.dsi.uminho.pt/>

<sup>13</sup> <http://jpgraph.net/>

## REGISTO DE NOVOS UTILIZADORES POR SEMANA

O gráfico da figura 21 refere-se ao número de novos utilizadores, por semana, que se registaram no sistema FindMePlaces.

Pode-se verificar a existência de dois picos na adesão à utilização do sistema FMP. Um na semana onde ocorreu a primeira divulgação, sendo o outro na semana da segunda divulgação. Nestas duas semanas foi a altura em qual se angariaram maior número de utilizadores.

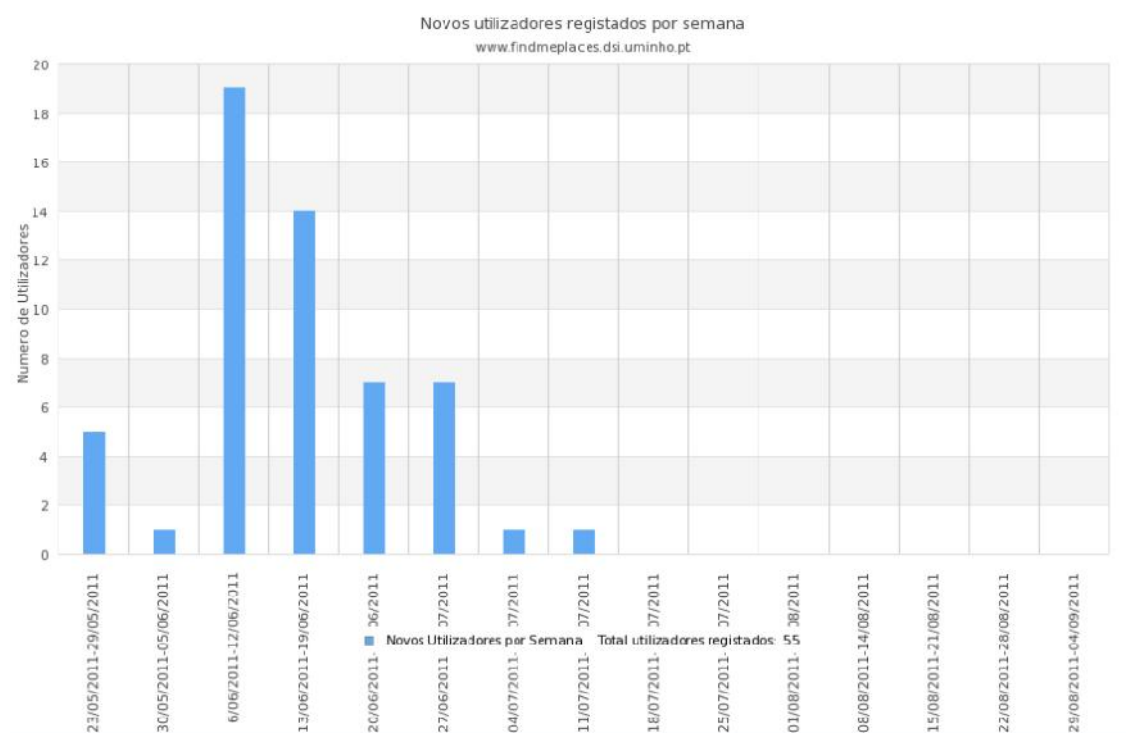


FIGURA 21: NOVOS UTILIZADORES REGISTADOS POR SEMANA.

## CRIAÇÃO DE NOVOS PLACES POR SEMANA

O gráfico da figura 22 apresenta os novos places criados por semana nos protótipos 1 (a azul) e protótipo 2 (a rosa). Nas semanas em que foi feita a divulgação verificou-se um maior número de criações de novos *places*.

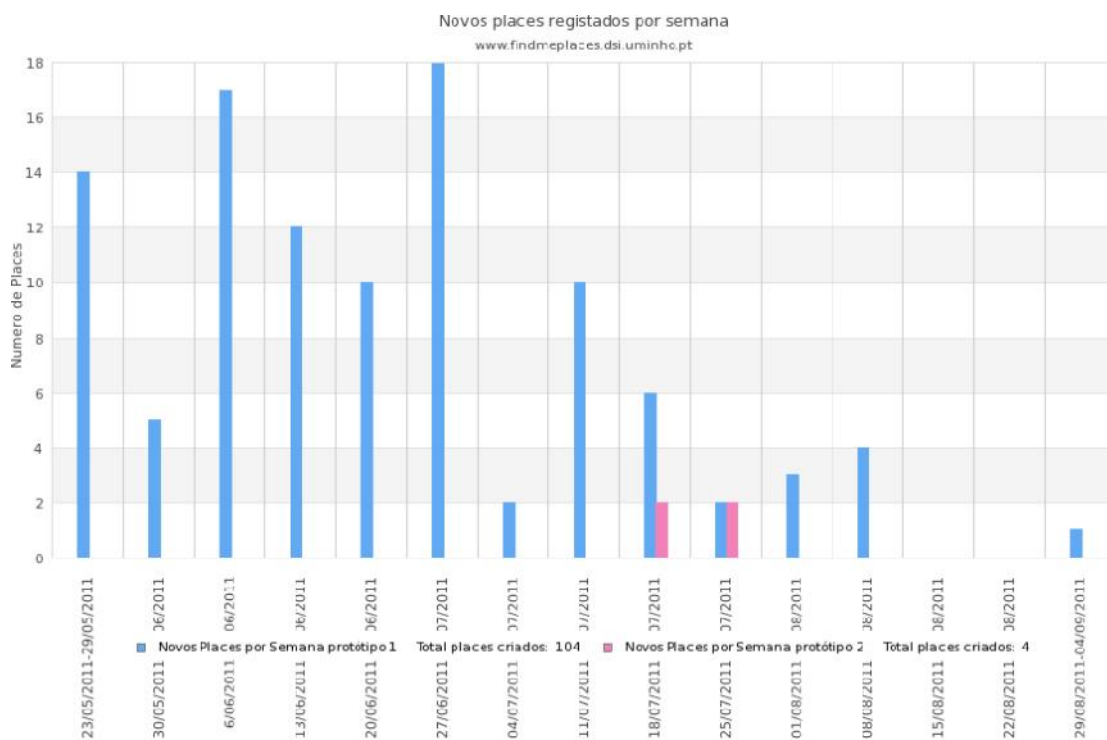


FIGURA 22: NOVOS PLACES REGISTRADOS POR SEMANA.

## NOVAS PUBLICAÇÕES POR SEMANA

Na figura 23 é apresentado o gráfico que resulta de número das publicações feitas pelos utilizadores em ambos os protótipos. O número total de publicações efectuadas em todo o período de monitorização do sistema é de 195.

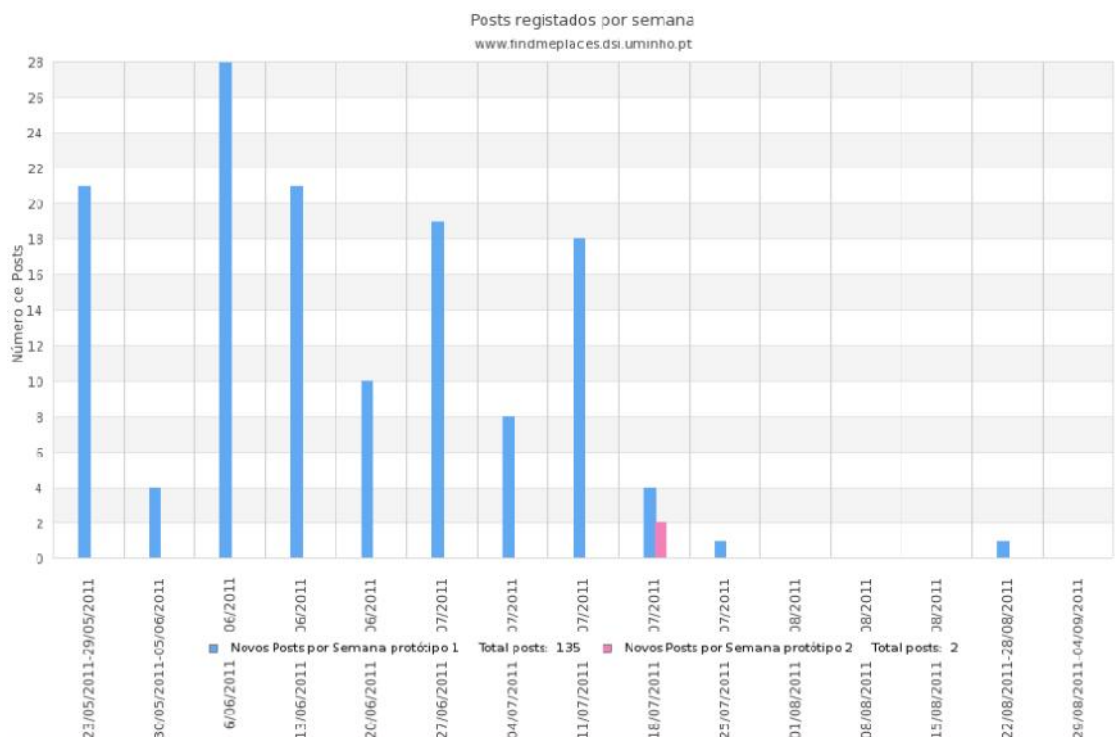


FIGURA 23: PUBLICAÇÕES REGISTADAS POR SEMANA.

### 5.6.3 ANÁLISE DOS RESULTADOS

Com o objectivo de testar os protótipos no mundo real, estes foram colocados em uso, disponibilizando ao público a aplicação FMP. Cada protótipo foi testado em fases diferentes.

Durante o período experimental, foram monitorizados 3 parâmetros: O registo de novos utilizadores, novas publicações e a criação de novos *places*.

Dessa monitorização obtivemos as estatísticas apresentadas na secção anterior.

O protótipo 1 (representado a azul), foi o primeiro a ser posto em prática. O protótipo 2, representado a rosa, foi disponibilizado posteriormente.

Podemos verificar que o sucesso de utilização da aplicação diminuiu drasticamente com a introdução do protótipo 2. Este facto deveu-se a limitações de existência de APIs para detecção de localização simbólica através de endereços Mac de APs.

Enquanto que através do protótipo 1, todos os browsers utilizam a API de localização recorrendo a coordenadas geográficas, no protótipo 2, apenas através do browser firefox é possível utilizar a localização por *macaddress*, não evitando mesmo assim a necessidade de reconfiguração do browser.

Foi tentado implementar um certificado no servidor do FMP como indicado pelas políticas de certificação da mozilla, para impedir a necessidade de reconfiguração, mas este nunca funcionou da forma esperada.

Os restantes browsers eram automaticamente direccionados para a utilização do protótipo 1. Isto revelou-se uma lacuna funcional.



## 6 CONCLUSÕES

Apesar da existência de várias abordagens nesta área, ainda são escassos os trabalhos que realizam o estudo de espaço de desenho, tendo em conta as limitações e diferenças de cada um deles, e a análise de características de APIs de localização existentes para aplicações Web, ponderando as suas limitações e oportunidades com vista a desenvolver os modelos de suporte e integração das funcionalidades *Wi-Fi* nas funcionalidades de PBS.

Com a realização deste trabalho, foi feito um estudo sobre a forma de explorar as funcionalidades obtidas da interacção das redes *Wi-Fi* com as funcionalidades das PBS, assim como as possibilidades de integração de recursos locais. Numa primeira fase, foi feita a investigação sobre os *place-based systems* e sobre alguns mecanismos de redes *Wi-Fi* que permitem a sua utilização de forma oportunista.

Inicialmente, foi feita a conceptualização do espaço de desenho baseando-se nos cenários reais, evidenciando o papel dos actores e dos recursos e como estes podem contribuir para uma melhor virtualização do espaço físico.

Foram definidos os modelos de suporte aos places e os modelos de integração, que permitiram mapear as soluções técnicas disponíveis, para cada um dos modelos.

Baseada nesta investigação e análise, procedeu-se à implementação do sistema FindMePlaces. Decidiu-se implementar um protótipo baseado na tecnologia *Wi-Fi*, como o principal recurso que permite suportar as funcionalidades de *check-in* e descoberta de utilizadores.

Ao desenvolver o segundo protótipo, implementou-se e experimentou-se a utilização oportunista de recursos *Wi-Fi*. Concluiu-se que existem grandes limitações tecnológicas a nível das APIs de localização para aplicações *Web based*, existindo apenas uma API, e somente para firefox, que permite a leitura da informação sobre os endereços mac e força do sinal dos AP's. Apesar destas limitações, implementou-se o protótipo 2, sendo este testado com alguns utilizadores.

A integração do FindMePlaces com outros sistemas foi realizada, explorando a integração de um PBS com um Modelo de Espaço simbólico e com o sistema de

*Collaborative Sensing Epi.* Ao efectuar esta integração e avaliando o resultado, ficou evidente que ainda existe muito por explorar na área de integração de modelos de espaço simbólico com outros sistemas, visto que este é somente utilizado no momento de criação de places.

## REFERÊNCIAS

1. **Anand Balachandran, Geoffrey M. Voelker Paramvir Bahl.** *Wireless Hotspots: Current Challenges and Future Directions.*
2. **B. O'Hara, A. Petrick.** *IEEE 802.11 Handbook- A Designer's Companion.*
3. **Jamshid Heidari, João Rodrigues, Luís Carvalho.** *CoolMe: A Wifi-Enabled Cool Places Recommendation System.*
4. **Rui José, Hélder Pinto, Bruno Silva, Ana Melro, Helena Rodrigues.** *Beyond interaction: Tools and practices for situated publication in display networks.*
5. **Nascimento, Fernando Ney da Costa.** *Um serviço para inferência de localização de dispositivos móveis baseado em redes ieee 802.11.*
6. **Karolina Baras, Adriano Moreira.** *Groups and Frequent Visitors Shaping the Space Dynamics.* in proceedings of the Smart Spaces and Next Generation Wired/Wireless Networking 11th International Conference, NEW2AN 2011, and 4th Conference on Smart Spaces, ruSMART 2011, St. Petersburg, Russia, Springer-Verlag, Lecture Notes in Computer Science, Vol. : s.n., ruSMART 2011.
7. **Sebastian Büttner, Henriette Cramer, Mattias Rost, Nicolas Belloni, Lars Erik Holmquist.**  *$\varphi^2$ : Exploring physical Check-Ins for Location-Based Services.*
8. **Aylett, M. Y. Lim and R.** *MY Virtual Graffiti System.* 2004.
9. **Eden, Terence.** *Introducing QRpedia.* 2011-04-03.
10. **Janne Lindqvist, Justin Cranshaw, Jason Wiese, Jason Hong, and John Zimmerman.** *I'm the Mayor of My House: Examining Why People Use foursquare - a Social-Driven Location Sharing Application.*
11. **Specification, W3C Geolocation API.** <http://www.w3.org/TR/geolocation-API/>. June 2011.
12. **Severina, Mauro Brunato and Danilo.** *WilmaGate: a New Open Access Gateway for Hotspot.*
13. **API, Gecko.** [https://developer.mozilla.org/en-US/docs/Monitoring\\_WiFi\\_access\\_points](https://developer.mozilla.org/en-US/docs/Monitoring_WiFi_access_points).
14. <http://www.mozilla.org/projects/security/components/signed-scripts.html>.
15. **Helena Rodrigues, Maria João Nicolau, Rui João José and Adriano Moreira.** *Engaging participants for collaborative sensing of human mobility.* 2012.

16. *ISO/IEC Standard for Information Technology- Telecommunications and Information Exchange Between Systems- Local and Metropolitan Area Networks- Specific Requirements- Part 11: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications Am.*

17. **Por Jochen H. Schiller, Agnès Voisard.** *Location-Based Services.* 2004.

18. Wiffiti. [Online] June 2011. <http://wiffiti.com/>.

19. **Sousa, Carlos.** *Difusão Epidémica de Mensagens em Hotspots WiFi.*

20. <http://www.grouplens.org/system/files/Terveen-Geographical.pdf>

## ANEXOS

### ANEXO 1- FUNÇÕES EPI (FONTE (16))

**Função f2\_1** - Verificar a existência de um objecto pelo seu nome.

HTTP Request :

[http://local.dsi.uminho.pt:8080/sc2/SCTX?function=f2\\_1&objname=name&infer=1](http://local.dsi.uminho.pt:8080/sc2/SCTX?function=f2_1&objname=name&infer=1)

- function- tipo de função;
- objname- nome de objecto;
- infer

Código de erro: *QP002 - Requested object not found*

**Função f2\_2** - Obtêm objecto pelo valor dos seus atributos.

- HTTP Request :

[http://host:port/sc/SCTX?function=f2\\_2&attvalue=value&infer=2](http://host:port/sc/SCTX?function=f2_2&attvalue=value&infer=2)

- function- tipo de função;
- attvalue
- infer

- Código de erro: *QP003 - Requested object not found*

**Função f2\_3** - Verificar a existência de um objecto pelo seu nome.

- HTTP Request :

[http://local.dsi.uminho.pt:8080/sc2/SCTX?function=f2\\_1&objname=na  
me&infer=1](http://local.dsi.uminho.pt:8080/sc2/SCTX?function=f2_1&objname=name&infer=1)

- function- tipo de função;
- objname- nome de objecto;
- infer

- Código de erro: *QP002 - Requested object not found*

- **Função f3\_1**- Obtêm objecto pelo valor dos seus atributos.

- HTTP Request :

[http://host:port/sc/SCTX?function=f2\\_2&attvalue=value&infer=2](http://host:port/sc/SCTX?function=f2_2&attvalue=value&infer=2)

- function- tipo de função;
- attvalue
- infer

- Código de erro: *QP003 - Requested object not found*

- **Função f3\_3** - Verificar a existência de um objecto pelo seu nome.

- HTTP Request :
  - [http://local.dsi.uminho.pt:8080/sc2/SCTX?function=f2\\_1&objname=name&infer=1](http://local.dsi.uminho.pt:8080/sc2/SCTX?function=f2_1&objname=name&infer=1)
    - function- tipo de função;
    - objname- nome de objecto;
    - infer
- Código de erro: *QP002 - Requested object not found*

Nome da função	Descrição	HTTP Request	Código erro
Função f2_1	Verifica a existência de um objecto pelo seu nome.	<u><a href="http://local.dsi.uminho.pt:8080/sc2/SCTX?function=f2_1&amp;objname=name&amp;infer=1">http://local.dsi.uminho.pt:8080/sc2/SCTX?function=f2_1&amp;objname=name&amp;infer=1</a></u>	QP002 - Requested object not found
Função f2_2	Obtem objecto pelo valor dos seus atributos.	<u><a href="http://host:port/sc/SCTX?function=f2_2&amp;attvalue=value&amp;infer=2">http://host:port/sc/SCTX?function=f2_2&amp;attvalue=value&amp;infer=2</a></u>	QP003 - Requested object not found
Função f2_3	Obtem os paramentos de objecto pelo seu id.	<u><a href="http://host:port/sc/SCTX?function=f2_3&amp;objid=26&amp;infer=3">http://host:port/sc/SCTX?function=f2_3&amp;objid=26&amp;infer=3</a></u>	QP004 - Requested object not found
Função f3_1	Inserção de um novo objecto.	<u><a href="http://local.dsi.uminho.pt:8080/sc2/SCTX?function=f3_1&amp;objname=name&amp;objtype=type&amp;objauthor=author&amp;objcdate=YYYY-MM-DD hh:mm:ss">http://local.dsi.uminho.pt:8080/sc2/SCTX?function=f3_1&amp;objname=name&amp;objtype=type&amp;objauthor=author&amp;objcdate=YYYY-MM-DD hh:mm:ss</a></u>	QP007 - Insert object failed
Função f3_3	Inserção de uma nova relação.	<u><a href="http://local.dsi.uminho.pt:8080/sc2/SCTX?function=f3_3&amp;relname=name&amp;reltype=type&amp;objdom=123&amp;objcdom=321&amp;hostcdom=host&amp;portcdom=8080">http://local.dsi.uminho.pt:8080/sc2/SCTX?function=f3_3&amp;relname=name&amp;reltype=type&amp;objdom=123&amp;objcdom=321&amp;hostcdom=host&amp;portcdom=8080</a></u>	QP009 - Insert relation failed

## ANEXO 2 - INTERFACE ENTRE FMP E SISTEMA EPI FUNÇÃO F1 (FONTE (16))

A Mensagem 1 corresponde a um pedido HTTP POST com a seguinte forma:

<http://epi.dsi.uminho.pt:8080/epiServerPlaces/epiSP?f1=f1&clm=clm&has=has&mts=mts&sts=sts&bss=bss&ssi=ssi&cha=cha&rss=rss&sec=sec&typ=typ>

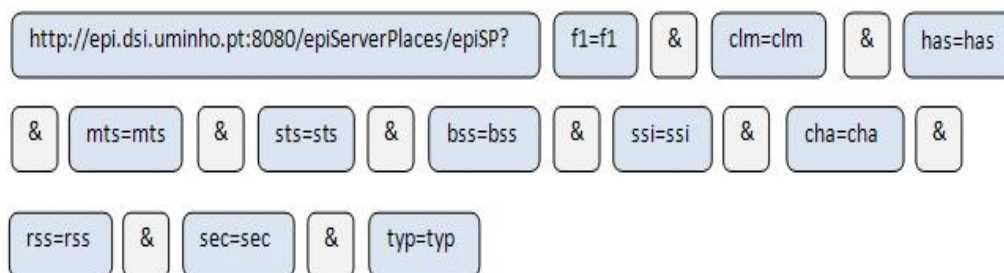


Figura 25: Mensagem enviada para o servidor de FMP a fazer submissão de assinatura rádio.

Este pedido inclui, no corpo do pedido POST, os seguintes campos:

**f** – identifica a função que se pretende invocar. Neste caso deve tomar o valor “f1”.

**clm** – identifica a máquina de onde a mensagem foi enviada, através do seu endereço MAC na forma “XX:XX:XX:XX:XX:XX”, onde XX representa um byte em formato hexadecimal (00 a FF). Se não for possível obter o endereço MAC da máquina onde o Brower está a correr, substituir por “00:00:00:00:00:0”.

**has** – transporta o resumo da mensagem enviada ou recebida. É representado por uma string com 32 caracteres hexadecimais. No caso do pedido não se referir a uma mensagem, este campo deve conter a string “nomsg”. No caso do FindMePlaces, é usada a string “nomsg”.

**mts** – transporta o instante de tempo em que a mensagem foi criada, na forma de uma string com 26 caracteres com o formato “yyyy:MM:dd:HH:mm:ss;±xx:zz”, em que yyyy representa o ano, MM representa o mês, dd representa o dia, mm representa os minutos, ss representa os segundos e ±xx:zz representa o desfaseamento horário existente em várias zonas do mundo. No caso do FindMePlaces é “2011:01:01:00:00:00;+00:00”.

**sts** – transporta o instante de tempo em que a assinatura rádio foi recolhida, na forma de uma string com 19 caracteres com o formato “yyyy:MM:dd:HH:mm:ss”, em que yyyy representa o ano, MM representa o mês, dd representa o dia, mm representa os minutos, ss representa os segundos.

**bss** – transporta uma lista com os BSSIDs dos APs detectados. Este campo será representado por uma string com a lista de endereços MAC, separados pelo carácter “;”. Cada endereço MAC deve ser transmitido na forma “XX:XX:XX:XX:XX:XX”, onde XX representa um byte em formato hexadecimal (00 a FF).

**ssi** – transporta uma lista com os nomes das redes detectadas (um registo por AP detectado). Este campo será representado por uma string. Cada nome da rede anunciada terá um limite de 15 caracteres, os nomes são separados pelo carácter “;” (ex: “eduroam;e-UM;e-U”). Se não for possível obter os nomes das redes, substituir pela string “unknown” para cada um dos APs detetados.

**cha** – transporta uma lista com a identificação dos canais rádio que estão a ser usados em cada um dos APs detectados, separadas pelo carácter “;”. Cada canal é representado por uma string com mínimo 1 carácter para canais iguais ou inferiores a 9 e máximo 2 caracteres para canais superiores a 9. Se não for possível obter os canais rádio, substituir pelo valor 0 para cada um dos APs detetados.

**rss** – transporta uma lista com o valor da força do sinal recebido de cada um dos APs, separados pelo carácter “;”. Cada valor é representado por uma string com o mínimo de 3 caracteres para forças de sinal superiores a “-100”, e 4 caracteres para forças de sinal iguais ou inferiores a “-100”.

**sec** – transporta uma lista com o tipo de segurança usado na rede que está a ser anunciada por cada AP, por exemplo WEP, WPA, WPA-TKIP, separados pelo carácter “;”. Cada valor é representado por uma string com um máximo de 8 caracteres. Se não for possível obter o protocolo de segurança, substituir pela string “unknown” para cada um dos APs detetados.

**typ** – transporta uma lista com a identificação do tipo de rede, se é uma rede ad-hoc ou se é anunciada por um AP, separados pelo carácter “;”. Cada valor é representado por uma string que pode tomar um de dois valores: “a” no caso de uma rede ad-hoc, ou “i”



no caso de uma rede com infraestrutura. Se não for possível obter o tipo de rede, substituir pelo carácter “?” para cada um dos APs detetados.

A um pedido do tipo f1, o Servidor Central deve responder com uma das seguintes strings:

### **Mensagem\_R Epi**

**001;record\_id** – significa que o pedido foi aceite e que os respetivos dados foram armazenados na base de dados com o número do registo representado por record\_id.

**100;description** – significa que o pedido não foi aceite devido a algum erro interno do Servidor Central, erro esse descrito por description.

**101** – significa que o pedido não respeita o formato definido e que, por isso, os dados não foram armazenados na base de dados.

**102;description** – significa que o pedido respeita o formato definido mas que não foi possível armazenar estes dados na base de dados, sendo o erro descrito por description.

### ANEXO 3 - WILMAGATE

WilmaGate (12) é um sistema de gestão de *hotspots Wi-Fi* baseado no paradigma de OAN (*Open Access Network*). Este permite a interação entre diferentes fornecedores de acesso à Internet (ISP) e operadores de autenticação, com objectivo de garantir o acesso à Internet aos utilizadores através de qualquer *hotspot* independente do ISP. Uma das funcionalidades deste sistema é a possibilidade de autenticação de utilizadores via *captive portal*. O *Captive portal* é uma tecnologia utilizada para o controlo de acesso na maioria dos *hotspots Wi-Fi* e em redes locais por cabo. Para alcançar este objectivo o WilmaGate proporciona funcionalidades de autenticação e autorização distribuída; além de mecanismos de segurança, e *firewalling*, entre outras, que por norma são fornecidas por componentes standard de um sistema de acesso *Gateway*.

A figura 23 ilustra a arquitectura modular do sistema WilmaGate, que é constituído por dois módulos principais: o *Gatekeeper* e o *Gateway*.

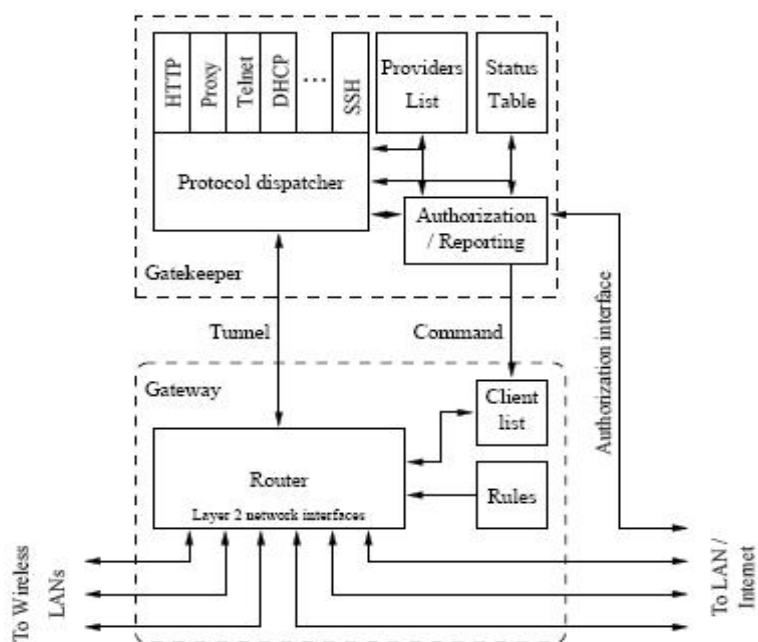


Figura 26:Arquitectura do sistema Wilmagate.

O componente *Gateway* é uma máquina que contém diversas interfaces de rede (pelo menos duas), algumas das quais são destinadas a redes sem fios LAN (wireless LANs) e algumas para redes com fio (wired ou LAN/ Internet, conforme arquitectura). O seu principal objectivo é desempenhar todas as funcionalidades de rede, ou seja, é basicamente um switch ao nível da camada 3 com algumas funcionalidades adicionais.

O Gateway contém 3 módulos principais:

**Router** - este módulo efectua a troca de tramas *Ethernet* entre interfaces de rede e o *Gatekeeper*. Este módulo identifica o destino e a origem de pacotes de acordo com os seus endereços MAC/ IP.

**Client list** - esta lista armazena a correspondência entre o endereço MAC e IP dos clientes autorizados. A Lista de clientes é acedida pelo router somente via interface de leitura. O router toma decisões sobre os pacotes com base no estado de autenticação do cliente, armazenados na *client list*, e no protocolo de pacotes de rede e nível de transporte;

**Rules** – é uma tabela que armazena a política de encaminhamento praticada pelo router. As regras de encaminhamento executadas pelo router são armazenadas na tabela Rules, criada na inicialização e acedida pelo Router através de leitura de uma interface.

Todos os pacotes provenientes de clientes wireless, cujo estado é não autorizado são reencaminhados para o *Gatekeeper* para o processamento adicional, enquanto todos os pacotes recebidos do *Gatekeeper* são encaminhados para a interface apropriada e sem interferência das regras aplicadas pelo *Gateway*.

As *Rules* e *Client List* são mantidas em forma mais simplificada possível, a fim de evitar soluções complexas na implementação de hardware dedicado e sobrecarga de tráfego de rede. A consulta à *Client List* é efectuada através de interface de comandos de texto simples.

O módulo *Gatekeeper* executa todos os outros pedidos que requerem mais processamento do que apenas consultar o cabeçalho de pacote IP. Ele recebe todos os pacotes IP não autorizados do Gateway via **Tunnel** socket UDP.

O pacote é processado com objectivo de autenticar o utilizador, e eventualmente este é enviado novamente para o Gateway para ser encaminhado. Os principais módulos do Gatekeeper são:

- *Status Table* – Mantém a tabela de estados de cliente;
- *Provider List* - Mantém a lista de fornecedores de serviço internet ou ISP's ;
- *Authorization/ Reporting* - Autenticação e autorização do cliente;
- *The Dispatcher*;

O Gatekeeper actua de acordo com os pacotes recebidos do Gateway, actualiza a tabela de estados e executa os comandos de volta para o Gateway.

Os comandos podem ser os seguintes:

- Enviar o pacote para uma interface específica;
- Autorizar o cliente identificado pelo MAC/IP;
- Revogar autorização atribuída ao cliente;

Adicionalmente ao túnel do Gateway, o Gatekeeper interage com entidades remotas via sockets TCP:

- O Interface Authorization é usada para receber informação de servidores de autenticações confiáveis sobre autorização de utilizador.

- O Interface Report é utilizada para obter os dados sobre o estado de utilizador como também para manter actualizado o Gateway sobre as alterações no estado de utilizadores.

### **Autenticação de clientes no WilmaGate**

O cliente deve ter um IP válido e uma autorização para utilização de serviços internet.

Para obter a autorização, o cliente deve contactar o seu servidor de autenticação e enviar-lhe as informações solicitadas sobre sua própria identidade. Se o servidor de autenticação reconhecer os direitos de utilizador, ele contacta o WilmaGate e informa que o cliente tem permissão para aceder aos serviços internet.

Quando o cliente envia informação para o servidor de autenticação ele ainda não está autorizado a navegar na internet pelo WilmaGate, portanto o Wilmagate necessita de uma política de encaminhamento e autenticação de pacotes.

