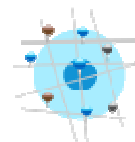


# Relatório de Trabalho Final de Curso

---

*Licenciatura em Engenharia Informática*



Projecto MeOnMap

*Julho de 2011*

# FICHA TÉCNICA

**Universidade:** Universidade Lusófona de Humanidades e Tecnologias

**Ano lectivo:** 2010 / 2011

**Curso:** Licenciatura em Engenharia Informática

**Cadeira:** Trabalho de Final de Curso

**Orientador:** Professor Sérgio Guerreiro

**Projecto:** MeOnMap

**Alunos:** Bruno Manuel David António - Nº 20063141

E-Mail: [brunomdantonio@gmail.com](mailto:brunomdantonio@gmail.com)

Telefone: 962705338

Hugo Miguel Assis Augusto – Nº 20063138

E-Mail: [hugo.assis@gmail.com](mailto:hugo.assis@gmail.com)

Telefone: 962975680

# ÍNDICE

<b>1</b>	<b>INTRODUÇÃO .....</b>	<b>6</b>
<b>2</b>	<b>ENQUADRAMENTO TEÓRICO .....</b>	<b>7</b>
2.1	ARQUITECTURA DA SOLUÇÃO .....	7
2.2	ÂMBITO DO PROJECTO .....	8
2.2.1	<i>Aplicação MeOnMap .....</i>	<i>9</i>
2.2.2	<i>Servidor MeOnMapServer.....</i>	<i>9</i>
2.3	CONCEITOS APLICACIONAIS .....	10
2.3.1	<i>Frequência de actualização .....</i>	<i>10</i>
2.3.2	<i>Raio de visualização.....</i>	<i>11</i>
2.3.3	<i>Estado dos amigos .....</i>	<i>11</i>
2.3.4	<i>Pedido de localização.....</i>	<i>12</i>
2.3.5	<i>Blacklist.....</i>	<i>12</i>
2.3.6	<i>Notificações .....</i>	<i>12</i>
2.3.7	<i>Modo Background.....</i>	<i>13</i>
<b>3</b>	<b>PROJECTO.....</b>	<b>14</b>
3.1	TECNOLOGIAS UTILIZADAS .....	14
3.2	INTEGRAÇÃO ENTRE COMPONENTES .....	15
3.3	MEONMAPSERVER.....	15
3.3.1	<i>End point friends/get .....</i>	<i>16</i>
3.3.2	<i>End point friends/positions .....</i>	<i>17</i>
3.3.3	<i>End point blacklist/update.....</i>	<i>20</i>
3.3.4	<i>Base de Dados.....</i>	<i>21</i>
3.3.5	<i>Volume das comunicações.....</i>	<i>22</i>
3.4	APLICAÇÃO MEONMAP .....	22
3.4.1	<i>Arquitectura de componentes .....</i>	<i>23</i>
3.4.2	<i>Descrição técnica .....</i>	<i>24</i>
3.4.3	<i>Workflow utilizado.....</i>	<i>28</i>
<b>4</b>	<b>CONCLUSÕES E TRABALHO FUTURO .....</b>	<b>29</b>
<b>5</b>	<b>BIBLIOGRAFIA.....</b>	<b>33</b>

# ÍNDICE DE IMAGENS

IMAGEM 1 – ARQUITECTURA DO SISTEMA .....	8
IMAGEM 2 – ENDPOINTS DO WEB SERVICE .....	16
IMAGEM 3 - TABELA <i>USERS</i> COM ALGUNS DADOS .....	21
IMAGEM 4 – INTERFACE DE MAPA DA APLICAÇÃO MEONMAP .....	23
IMAGEM 5 – ARQUITECTURA LÓGICA.....	24
IMAGEM 6 – MECANISMO DE AUTENTICAÇÃO API FACEBOOK.....	25
IMAGEM 7 – CONFIGURAÇÃO DE MECANISMO DE LOCALIZAÇÃO DO ANDROID .....	26
IMAGEM 8 – REPRESENTAÇÃO DO WORKFLOW EM ESQUEMA .....	28

# Resumo

Este documento pretende apresentar e descrever as componentes do sistema implementado no âmbito da realização da cadeira de Trabalho Final de Curso, bem como identificar as optimizações e evoluções ao presente trabalho.

Desta forma foi proposto o desenvolvimento do projecto MeOnMap. Este tem como objectivo fazer a integração de redes sociais com a componente de geolocalização, utilizando como meio as funcionalidades e mobilidade disponibilizada nos equipamentos *smartphone* com tecnologia Android. A arquitectura do sistema divide-se em duas componentes:

- MeOnMap - Aplicação móvel para equipamentos Android, que disponibiliza o interface de funcionalidades com o utilizador;
- MeOnMapServer - Sistema central, com capacidade para processamento e arquivo da informação disponibilizada pela aplicação móvel;

O projecto aqui apresentado implementa um conjunto de funcionalidades que demonstram de uma forma genérica os objectivos a que nos propusemos. No entanto estamos cientes de que este, ainda não atinge o estado de maturidade necessário para que possa ser considerado um produto final. Como tal, deverá ser considerado como um protótipo inteiramente funcional com potencial de crescimento em funcionalidades e algumas necessidades de optimização quanto à performance.

Genericamente, este documento encontra-se dividido em 5 capítulos e um anexo:

- Introdução – É feita uma introdução ao projecto e ao trabalho de final de curso;
- Enquadramento teórico - Apresentação do projecto e conceitos aplicacionais;
- Projecto – Descrição técnica das componentes do projecto;
- Conclusões e trabalho futuro – Identificação das optimizações e evoluções ao projecto;
- Bibliografia – Indicação das referências bibliográficas utilizadas;
- Anexo 1 – Manual do utilizador da aplicação MeOnMap;

Será entregue, em suporte digital (CD-ROM), todo o código fonte produzido no âmbito deste projecto.

# 1 Introdução

Este documento pretende apresentar e descrever as componentes do sistema implementado no âmbito da realização da cadeira de Trabalho Final de Curso, bem como identificar as optimizações e evoluções ao presente trabalho.

Desta forma foi proposto o desenvolvimento do projecto MeOnMap. Este tem como objectivo fazer a integração de redes sociais com a componente de geolocalização, utilizando como meio as funcionalidades e mobilidade disponibilizada nos equipamentos *smartphone* com tecnologia Android.

Tendo como objectivo as premissas anteriormente descritas o projecto MeOnMap, é uma aplicação para equipamentos móveis em tecnologia Android, onde o utilizador após autenticação na rede social Facebook, partilha a sua localização com os restantes utilizadores do sistema. Essa partilha é possível através da utilização de um repositório central (servidor MeOnMapServer), onde são armazenadas as últimas localizações dos utilizadores do sistema, partilhando-as através da aplicação móvel MeOnMap. Estas localizações são visualizadas sobre o mapa, em tempo real, utilizando a API Google Maps.

A nossa motivação para a realização deste trabalho está relacionada com a cada vez maior massificação da utilização das redes sociais, e tecnologias onde a palavra-chave é a mobilidade. Sendo actualmente a área das aplicações móveis e da geolocalização, algumas das áreas com maior crescimento e interesse no sector das TI, o desafio e motivação não poderia ser maior.

O trabalho aqui apresentado reflecte não só a experiência e conhecimentos técnicos adquiridos ao longo do curso, mas também um conjunto de experiencias profissionais adquiridos ao longo das nossas carreiras. Todo o trabalho foi realizado sob a orientação do professor Sergio Guerreiro.

## 2 Enquadramento teórico

Conforme referido, o projecto MeOnMap tem como objectivo disponibilizar aos utilizadores de equipamentos Android, uma aplicação móvel que permite a captura da sua localização actual, comunicando-a a um sistema central. O acesso à aplicação implica fazer a autenticação sobre a rede social Facebook, possibilitando assim que a sua localização, seja conhecida e partilhada pelos amigos membros da sua rede social e utilizadores do sistema MeOnMap. Ao utilizador, é disponibilizado um interface composto por um mapa, que apresenta a sua localização actual e a ultima localização conhecida para cada um dos seus amigos.

A implementação desta solução pressupõe a utilização e integração com outras tecnologias já disponíveis, nomeadamente a utilização de equipamentos móveis com tecnologia Android. Estes disponibilizam um conjunto de funcionalidades, tais como a utilização de GPS (Global Position System), capacidades de processamento e de acesso à Internet. Para este tipo de equipamentos, é promovida a utilização da API do Google Maps, adicionando assim uma componente geográfica à solução.

O Facebook é actualmente a rede social mais utilizada a nível mundial, estando anunciado no site que tem cerca de 750 milhões de utilizadores activos. Existem diversas formas de acesso ao Facebook, sendo o Facebook Mobile uma das formas privilegiadas de acesso, contando com cerca de 250 milhões de utilizadores activos. Este volume de utilizadores potencia a criação de aplicações específicas para esta plataforma e a disponibilização de API's para integração com outras ferramentas.

### 2.1 Arquitectura da solução

O projecto MeOnMap está dividido em duas componentes aplicacionais, onde foram implementadas as componentes tecnológicas anteriormente referidas, dando origem às funcionalidades específicas do sistema.

As duas componentes aplicacionais do MeOnMap são:

- **Aplicação MeOnMap** – Aplicação desenvolvida para dispositivos móveis com tecnologia Android, que tira proveito das funcionalidades disponibilizadas no equipamento tais como o GPS, capacidades de processamento e de ligação à

Internet. O utilizador interage com esta aplicação, para visualizar em mapa a sua localização e a localização dos seus amigos.

- **Servidor MeOnMapServer** – Aplicação desenvolvida para ambiente de servidor. Esta é composta por um Web service e respectiva base de dados centralizada, e possibilita a recepção e processamento da informação enviada por cada um dos equipamentos móveis.

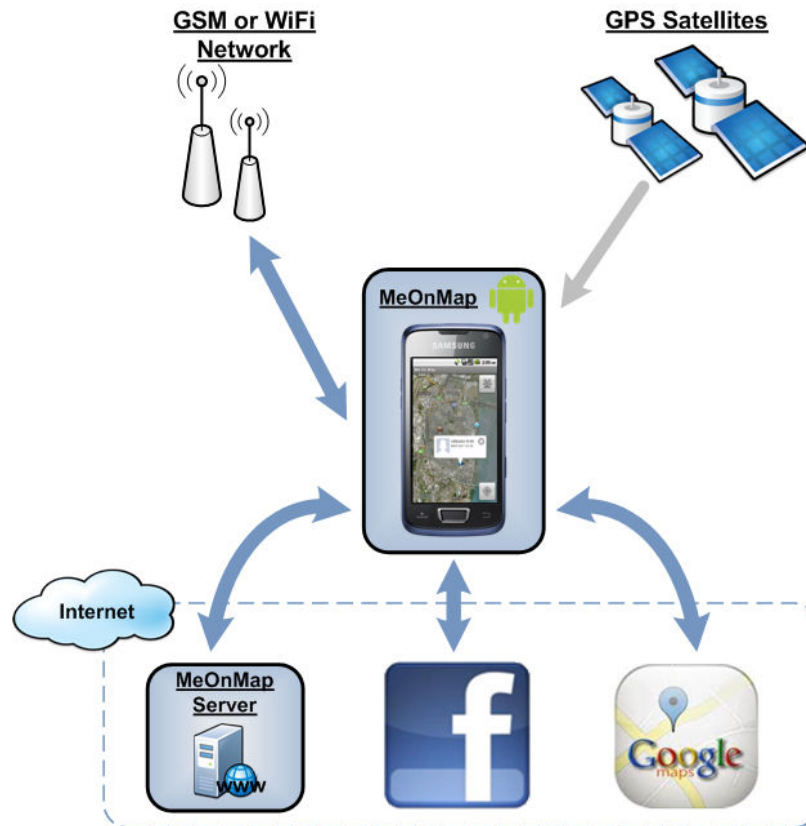


Imagem 1 – Arquitectura do sistema

## 2.2 Âmbito do projecto

O projecto MeOnMap surge com base num conjunto de ideias, onde se pretende reunir os conceitos da mobilidade, de utilização das redes sociais e dos sistemas de informação geográficos. Tendo estas ideias em mente, rapidamente foram identificadas um conjunto alargado de funcionalidades para implementar. No entanto, cientes das dificuldades tecnológicas que iríamos encontrar e com as restrições de tempo, foram identificados os requisitos mínimos para funcionamento do projecto, tendo focado os



nossos esforços de desenvolvimento na pesquisa de soluções para cada um desses requisitos. Após a implementação dessas ideias base, rapidamente começaram a surgir novas funcionalidades, e o desafio da implementação das mesmas, sobre a componente de mobilidade e a componente de servidor.

Apresentamos de seguida as funcionalidades mais relevantes de cada uma das componentes aplicacionais.

### 2.2.1 Aplicação MeOnMap

- Capacidade de *login* e *logout* no Facebook;
- Navegação em mapa;
- Apresentação de mapa com a localização actual, e informação acerca dos amigos (que pertencem à rede de amigos do Facebook e são utilizadores do MeOnMap);
- Consulta em mapa de informação acerca dos amigos, apresentando diferente simbologia em função do estado (Online, Offline, Fora do raio);
- Sistema de notificações de proximidade de amigo;
- Listagem de amigos com visualização da localização;
- Envio periódico da actual localização, e recepção da localização dos amigos;
- Configuração do tempo de actualização automática;
- Configuração da área (raio) para visualização de amigos;
- Definição da lista de amigos sem privilégios para aceder à sua localização (blacklist).
- Verificação do estado da rede, estado do GPS e disponibilidade do servidor MeOnMapServer;
- Possibilidade de execução da aplicação em modo de *background*;

### 2.2.2 Servidor MeOnMapServer

- Centralização de informação;
- Recepção das localizações dos utilizadores do sistema;

- Processamento da informação recebida (cálculo da distância, avaliação do estado dos amigos, e aplicação do filtro à localização - blacklist);
- Manutenção das listas de blacklist, por utilizador;
- Envio de informação acerca da localização dos utilizadores do sistema;

## 2.3 Conceitos aplicacionais

Embora o custo de acesso às comunicações móveis tenha vindo a sofrer um decréscimo, e a utilização de redes Wireless gratuitas seja cada vez mais frequente, ainda se está perante uma realidade diferente daquela que seria desejada. Assim, uma das preocupações que esteve sempre presente foi a de capacitar o utilizador em controlar e ajustar o volume de dados transmitido, em função da ligação disponível para acesso à Internet (WiFi ou 3G), e dos custos associados à utilização da mesma.

Desta forma, durante o processo de concepção e implementação, surgiu a necessidade de criação de alguns conceitos aplicacionais, que deram origem às funcionalidades, que passamos a descrever.

### 2.3.1 Frequência de actualização

O sistema permite visualizar em tempo real a localização dos amigos que utilizam a aplicação MeOnMap. O princípio base de funcionamento do sistema, assenta na capacidade da aplicação móvel comunicar numa determinada frequência ou ciclo de tempo com o servidor, transmitindo-lhe a localização do utilizador através da ligação Internet disponível e de um Web service específico. Sempre que ocorre uma comunicação de localização é retornado à aplicação uma lista de amigos com as respectivas localizações.

Naturalmente que estas comunicações e a frequência de actualização dos dados, tem impacto no volume de dados transmitido. Assim, foi disponibilizado ao utilizador uma forma de ajustar esta frequência, em função das suas necessidades ou restrições na utilização da ligação à Internet.

### 2.3.2 Raio de visualização

Sendo o volume de dados transmitido uma das preocupações na concepção e implementação do projecto, foi criado o conceito de raio de visualização. O utilizador ao definir este raio ou área de visualização, está a indicar ao sistema que pretende receber informação, apenas dos amigos (utilizadores do MeOnMap), que estejam dentro desse raio definido.

A título de exemplo de utilização, provavelmente não será de grande interesse o utilizador estar a receber de minuto a minuto a posição exacta de um amigo que se encontra em outro país ou continente. Caso pretenda saber especificamente a sua localização pode utilizar a funcionalidade de “pedido de localização” para tal. No entanto, já terá maior interesse saber de minuto a minuto, a posição exacta dos utilizadores do MeOnMap que se encontrem numa área de 3 km.

Desta forma, a cada ciclo de tempo, serão actualizadas apenas as localizações dos utilizadores que se encontrem dentro do raio, reduzindo assim o volume de dados transmitido. Existem opções de configuração que permitem definir este raio de visualização e uma opção onde o mesmo não é considerado.

### 2.3.3 Estado dos amigos

Ao utilizar a aplicação MeOnMap, o utilizador vai sendo informado acerca da última localização conhecida de cada um dos seus amigos bem como da data e hora da mesma. Assim, o interface em mapa apresenta a localização de cada amigo, utilizando diferentes simbologias para cada um dos três possíveis estados:

- **Estado Online** – Indica que o utilizador se encontra dentro do raio de visualização e que a sua última localização conhecida, foi registada há menos de 15 minutos. A sua representação em mapa é feita através de um símbolo azul.
- **Estado Offline** - Indica que o utilizador se encontra dentro do raio de visualização mas a sua última localização conhecida, foi registada há mais de 15 minutos. A sua representação em mapa é feita através de um símbolo castanho.
- **Estado Fora do Raio** - Indica que o utilizador se encontra fora do raio de visualização. Esta situação pode ocorrer, caso o utilizador tenha pré-definido um raio de visualização. Em determinado momento, o utilizador é notificado da proximidade de um amigo, mas pelo facto de um deles se encontrar em movimento, em determinada altura um pode sair do raio de influência do outro.

Nesta circunstância ficará registado em mapa, a ultima localização conhecida do seu amigo, sendo a sua representação em mapa feita através de um símbolo cinzento.

#### 2.3.4 Pedido de localização

Caso o utilizador pretenda otimizar o volume de dados transmitidos, poderá configurar o raio de visualização. No entanto, em determinadas circunstâncias poderá solicitar especificamente a localização de um amigo, independentemente da sua localização actual (dentro ou fora do raio de visualização). A localização solicitada será apresentada em mapa.

#### 2.3.5 Blacklist

Uma das questões mais pertinentes acerca dos sistemas de localização está relacionada com a invasão de privacidade das pessoas. Assim, o sistema contempla um mecanismo que permite ao utilizador configurar quais os amigos que tem privilégios para acederem à sua localização. Quando um utilizador define uma *blacklist*, selecciona um conjunto de amigos utilizadores do sistema, sendo essa lista comunicada ao servidor MeOnMapServer. Este processo não é recíproco, ou seja caso o utilizador “UserA” coloque o utilizador “UserB” em *blacklist*, significa apenas que o “UserB” vai deixar de visualizar a localização de “UserA”, continuando “UserA” a visualizar “UserB”.

#### 2.3.6 Notificações

O sistema operativo Android utiliza um mecanismo de notificações, que permite informar o utilizador acerca da ocorrência de eventos, através de um pequeno *beep* do equipamento e de um pequeno icon apresentado na barra do topo do ecrã. Com este mecanismo é permitido estar a interagir sobre uma aplicação, enquanto outra aplicação despoleta esses eventos e notificações, mantendo o interface do utilizador livre de mensagens e interrupções.

A presença de amigos dentro do raio de visualização do utilizador da aplicação MeOnMap vai accionar este mecanismo de notificações. Esta notificação ocorre quando é detectada a presença de um novo amigo no raio. Caso o amigo permaneça dentro do

raio por mais do que um ciclo de tempo, a notificação ocorrerá apenas a primeira vez. Caso ele se desloque para fora do raio de visualização, e volte a entrar dentro do raio, será novamente notificado.

Caso o utilizador seleccione a notificação, o mapa será ajustado para que apresente o enquadramento geográfico de todos os amigos alvo da notificação.

### 2.3.7 Modo Background

Ao executar a aplicação MeOnMap, é apresentado ao utilizador um interface de mapa com a sua localização. Este mapa vai sendo actualizado com informação acerca da localização dos amigos.

O Android permite a execução de multitarefas, ou seja, enquanto a aplicação está a ser executada ocupando o interface disponível pode por exemplo, receber uma chamada telefónica. Nessa situação a aplicação entra em modo de background, continuando a executar as operações de verificação de amigos no raio de visualização, comunicação com o servidor ou o processo de alerta de notificações. Sempre que a aplicação passa para este estado, o utilizador é notificado através de um alerta que surge temporariamente no ecrã.

## 3 Projecto

Este capítulo pretende descrever de uma forma mais técnica, cada uma das componentes aplicacionais desenvolvidas.

### 3.1 Tecnologias utilizadas

Este projecto foi implementado utilizando como base a tecnologia Java, e o JDK6.

Para a aplicação móvel, como ambiente de trabalho foi utilizado o IDE Eclipse versão 3.6.2 (Hélios), com os plugin's de desenvolvimento para Android obtidos no site [Android Developers](#). Este disponibiliza também um emulador de Android, que foi utilizado no processo de implementação e testes da aplicação. Com o objectivo de abranger o maior número possível de utilizadores, utilizamos a API Level 7, que é a versão orientada à plataforma Android 2.1.

Como ambiente de trabalho para a componente MeOnMapServer, foi utilizada a mesma versão do IDE Eclipse, onde foram adicionadas as bibliotecas de JAX-RS 1.1 (Jersey 1.6) dada a necessidade de implementar o REST como protocolo de comunicações, pois o JDK6 não disponibiliza bibliotecas próprias para este fim.

Como servidor aplicacional escolhemos o *Apache Tomcat* versão 7.0.12, por cumprir os requisitos necessários a este projecto, e por já existir alguma experiencia profissional na utilização desta ferramenta.

Como repositório de dados optamos por utilizar o *PostgreSQL* versão 9. A escolha recaiu por este motor de base de dados por ser uma ferramenta *open source* e com funcionalidades que permitem o armazenamento de dados em formato geográfico, através da utilização de uma extensão específica que é o *PostGIS*. Com a evolução e implementação do projecto identificamos que as operações espaciais a realizar recaiam genericamente sobre a componente móvel do projecto, e como tal seriam implementadas através do *Google Maps API* para Android. Assim reavaliámos e optamos por não a utilizar a extensão do *PostGIS*. Para administração e criação da base de dados foi utilizado o software *PgAdmin III* versão 1.12.2.

A utilização da *Google Maps API* para Android implicou a criação de uma chave de utilização. Assim, no início do processo de desenvolvimento foi necessário registar a aplicação MeOnMap para obtermos a *Maps API Key* na plataforma *Google Maps*. Sem

esta chave e sem a inclusão da mesma no código, a componente de mapas não funciona.

Também foi necessário registar a aplicação MeOnMap na área destinada a programadores do Facebook. Neste processo de registo foi fornecido um *Application Id*, que é utilizado no processo de autenticação do MeOnMap no Facebook. Apenas desta forma é possível ter acesso à Facebook API, podendo assim recolher informação de cada utilizador e dos seus respectivos amigos. A API está disponível para download no endereço <https://github.com/facebook/facebook-android-sdk>. Apenas como curiosidade, o nome inicialmente escolhido para o sistema era “FaceOnMap”, no entanto durante este processo de registo fomos impedidos pelo Facebook de o fazer, por conter a palavra “Face”.

### 3.2 Integração entre componentes

Este sistema está dividido em dois componentes, a aplicação móvel MeOnMap e o MeOnMapServer que implementa um Web service, sendo este o mecanismo utilizado para a integração entre estas duas componentes.

Para implementação do processo de comunicações entre as componentes, considerámos a utilização dos protocolos *REST* e *SOAP*. A principal diferença entre estes, reside no tipo de estrutura necessário para *serializar* os dados a serem transmitidos. O *SOAP* faz a *serialização* tendo como base uma estrutura em formato *XML*. O *REST* utiliza um mecanismo diferente, onde a estrutura para a *serialização* é optimizada, conseguindo assim uma redução no volume de dados transmitido. Desta forma, é preferível a utilização deste protocolo em situações onde o volume de dados transmitido é um factor importante. O *REST* implementa diferentes formatos de comunicação sendo a nossa opção a utilização do formato *JSON (JavaScript Object Notation)*, devido á sua fácil implementação, e pela existência de classes e mecanismos que permitem fazer o *cast* directo para objectos JAVA. Este formato é utilizado também no processo de comunicação com a API do Facebook.

### 3.3 MeOnMapServer

O servidor é constituído por três *end points*, como mostra a figura.

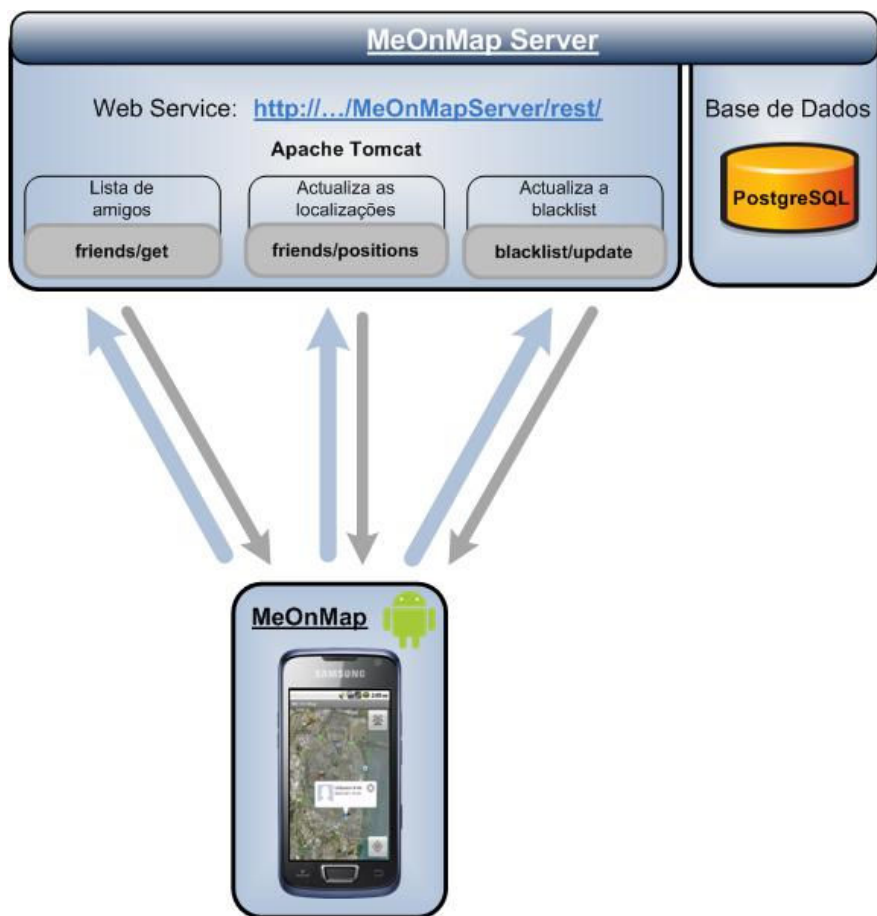


Imagem 2 – Endpoints do Web service

### 3.3.1 End point *friends/get*

O objectivo deste *end point*, é informar a aplicação MeOnMap da lista de amigos que são utilizadores do sistema. Assim, este recebe o identificador de Facebook do utilizador, e uma lista com todos os identificadores de Facebook dos seus amigos. Ao receber esta informação a mesma será confrontada com a base de dados do sistema, retornando à aplicação MeOnMap uma lista com os amigos utilizadores do sistema. Para cada um desses amigos da lista, é associado um atributo que identifica se o mesmo pertence à *blacklist* do utilizador.

Genericamente, as acções executadas são:

- Recebe o id de utilizador e a lista de amigos.



- Com base na lista recebida é efectuada uma consulta á base de dados.
- Com esses dados vai verificar os que estão na *blacklist* do utilizador.
- É enviada a informação para a aplicação móvel.

Apresentamos de seguida a especificação do end point *friends/get*.

Estrutura recebida:

Parâmetros	Tipo	Descrição
Id	String	Id de utilizador
Fids	String [ ]	Lista de id's de amigos

Exemplo:

```
{"id":"100000298932162","fids":["512246944,528267895,...,100002444586885"]}
```

Estrutura a enviar:

Parâmetros	Tipo	Descrição
list	ArrayList[objecto]	Lista de amigos
Estrutura de objecto		
id	String	Id de amigo
is	String	Valores (0-Não está na <i>blacklist</i>    1-Está na <i>blacklist</i> )

Exemplo:

```
[{"id":"100002444586885","is":"0"}, {"id":"100002203750541","is":"0"}, {"id":"100001187977369","is":"0"}, {"id":"100000298932162","is":"0"}]
```

### 3.3.2 End point friends/positions

O objectivo deste *end point*, é actualizar na base de dados do sistema MeOnMap, a localização do utilizador. Sempre que este é executado, é feito o registo da última localização do utilizador, e a data e hora de ocorrência da mesma. Esta execução retorna uma lista com as localizações dos amigos utilizadores do sistema, aplicando alguns filtros à informação tais como a *blacklist* e o raio de visualização.

Na execução deste *end point*, é ainda verificado se o utilizador já existe na base de dados, e caso ainda não exista, é feita a sua criação.

Um dos parâmetros que recebe, é a lista dos amigos de Facebook. No entanto nem todos são utilizadores do sistema MeOnMap. Desta forma é necessário fazer a confrontação entre a lista recebida e a lista a devolver, sendo ainda aplicado o filtro da *blacklist*. Quando um utilizador coloca um amigo em *blacklist*, significa que esse amigo não deverá receber a localização desse utilizador. Assim, serão devolvidos apenas, os amigos que o utilizador tem permissão para visualizar.

Outro parâmetro recebido é o raio de visualização do utilizador. Desta forma, a lista a devolver também é filtrada pelos amigos (que respeitam as regras anteriores) e se encontram dentro do raio definido pelo utilizador. Quando é registada a localização de um utilizador, a mesma tem associada uma data e hora de ocorrência. Assim, ao utilizador que pretende visualizar a localização dos seus amigos, interessa saber se essa localização é recente ou já é mais antiga, aplicando-se assim o conceito anteriormente descrito como “estado dos amigos”. Para cada um dos amigos a devolver na lista, será associado um estado (online ou offline), em função da última data e hora conhecida. Existe ainda o estado de “fora do raio”, mas esse é inteiramente gerido na aplicação MeOnMap.

Genericamente, as acções executadas são:

- Recebe o identificador de utilizador (Id), as suas coordenadas de localização, o valor do raio e a lista de amigos do Facebook.
- Verifica se o utilizador já existe na base de dados. Caso exista actualiza a sua posição, caso contrário, insere-o na base de dados com a sua posição.
- Pesquisa na base de dados, quais os amigos que o utilizador não tem permissões para visualizar a sua localização.
- A lista de amigos recebida é confrontada com a pesquisa anterior, resultando assim na lista de amigos que podem ser visualizados.
- Actualiza a lista anterior, com as coordenadas e a data e hora.
- Verifica o estado de cada amigo. Está online, caso a última localização tenha ocorrido à menos de 15 minutos. Está offline, caso a última localização tenha ocorrido à mais de 15 minutos.

- Calcula a distância linear entre o utilizador e cada amigo, sendo devolvidos apenas os que se encontram dentro do raio.
- É enviada a informação a aplicação MeOnMap.

Apresentamos de seguida a especificação do end point *friends/positions*.

Estrutura recebida:

Parâmetros	Tipo	Descrição
id	String	Id de utilizador
r	Numérico	Raio
fids	String [ ]	Lista de id's de amigos
y	Numérico	Coordenada Y da posição do utilizador
x	Numérico	Coordenada X da posição do utilizador

Exemplo:

```
{"id":"100000298932162","r":"1000","fids":["512246944,528267895,536357511,537947705,...,100002444586885"],"y":"38773862","x":"-9115544"}
```

Estrutura a enviar:

Parâmetros	Tipo	Descrição
lista	ArrayList(Objecto)	Lista de amigos
Estrutura de objecto		
id	String	Id de amigo
x	Numérico	Coordenada X da posição do utilizador
y	Numérico	Coordenada Y da posição do utilizador
dtm	timestamp	Data Hora
on	String	Valores ( 0-Off Line    1-On Line)

Exemplo:

```
[{"id":"100001187977369","x":"-9115601","y":"38773803","dtm":"2011-07-16 23:11:13.291+01","on":"0"},{"id":"100000298932162","x":"-9115544","y":"38773862","dtm":"2011-07-17 05:20:11.47+01","on":"1"}]
```

### 3.3.3 End point blacklist/update

Este *end point* tem como objectivo guardar os identificadores de amigos, que o utilizador não quer que tenham permissões de o visualizarem na aplicação MeOnMap.

É recebido o identificador de utilizador e uma lista de identificadores de amigos.

O processo de registo na base de dados da *blacklist* é efectuado através da execução dos seguintes passos: a) é efectuado uma consulta, b) o utilizador é eliminado de todos os registos de *blacklist* onde ele se encontra c) o utilizador é acrescentado na coluna *blacklist* de cada um dos amigos.

Genericamente, as acções executadas são:

- Recebe o identificador de utilizador (Id), e a lista de amigos do Facebook.
- Executa uma consulta onde vai retirar o identificador de utilizador em todos os registos onde se encontra na coluna *blacklist*.
- Actualiza na base de dados o campo *blacklist* de cada amigo incluído na lista recebida com o identificador do utilizador.
- Neste caso o Web service não irá retornar nenhum valor, apenas é validado no lado da aplicação MeOnMap o valor do *http status*, que é gerido pelo próprio protocolo de comunicação.

Apresentamos de seguida a especificação do end point *blacklist/update*.

Estrutura recebida:

Parâmetros	Tipo	Descrição
id	String	Id de utilizador
bids	String [ ]	Lista de id's de amigos

Exemplo:

```
{"id":"100000298932162","bids":["100002218006840,100002203750541,100002444586885"]}
```

Não é feito o envio de nenhuma estrutura de dados.

### 3.3.4 Base de Dados

Como repositório de dados, foi utilizado o motor de base de dados PostgreSQL versão 9. Este serve para guardar os identificadores, localizações, data hora e *blacklist*.

Sendo um dos objectivos do projecto que os dados relativos ao utilizador e seus amigos fossem geridos através da rede social, a informação a guardar no sistema MeOnMap são apenas as localizações dos utilizadores. Desta forma não se justificou a criação de um modelo de dados com relações, sendo suficiente uma única tabela (*Users*), com a seguinte estrutura:

Nome	Tipo	Descrição
iduser	character varying(20)	Identificador de utilizador
posx	bigint	Coordenada X da localização do utilizador
posy	bigint	Coordenada Y da localização do utilizador
datetime	timestamp with time	Data e hora da localização do utilizador
blacklist	text	Lista de amigos que o utilizador não tem acesso à sua localização

A utilização do sistema origina um conjunto de registos conforme apresentados:

iduser [PK] character varying(20)	posx bigint	posy bigint	datetime timestamp with time zone	blacklist text
100002195824737	-7923360	41105699	2011-07-09 15:19:01.217+01	100000330866876,
100002203750541	-9110661	38817541	2011-07-08 20:04:11.612+01	100002218006840,100000298932162,
100002218006840	-9091276	38776669	2011-07-17 23:55:12.723+01	100000298932162,
100002258685278	-7919940	41112529	2011-07-10 18:25:55.114+01	' '
100002301527826	-7916515	41119369	2011-07-10 20:45:20.517+01	' '
100002332647471	-7913085	41126219	2011-07-08 20:05:26.664+01	' '
100002339406266	-7909650	41133079	2011-07-18 00:45:44.125+01	' '
100002354017288	-7906210	41139949	2011-07-15 21:55:42.458+01	' '
100002357081055	-7902765	41146829	2011-07-12 10:15:28.258+01	100000330866876,
100002386103129	-7899315	41153719	2011-07-02 20:55:45.458+01	100000330866876,
100002389876428	-7895860	41160619	2011-07-03 11:20:15.523+01	' '
100002421509021	-7892400	41167529	2011-07-05 12:33:12.125+01	100000330866876,
100002444586885	-9184383	38760146	2011-07-05 00:50:55.926+01	100002203750541,100002218006840,100000298932162,
100002496492225	-7888935	41174449	2011-07-05 12:34:45.123+01	100000330866876,
100002512300609	-7885465	41181379	2011-07-06 13:45:36.785+01	100000330866876,
100002596266486	-7881990	41188319	2011-07-07 16:28:32.302+01	100000330866876,
1003908220	-9087960	38782969	2011-07-07 16:30:49.204+01	100000330866876,
1003985089	-9087775	38783329	2011-07-08 15:45:20.304+01	100000330866876,

Imagem 3 - Tabela *Users* com alguns dados

Não foi efectuada nenhuma configuração específica à base de dados, para além das configurações já existentes por defeito.

### 3.3.5 Volume das comunicações

Como já referido, o volume de dados utilizado no processo de comunicações foi um dos factores que tivemos em consideração. O Google maps já implementa um mecanismo de *cache* que permite o arquivo local das áreas previamente visualizadas pelo utilizador, mas mesmo assim, o maior volume de tráfego da aplicação MeOnMap reside no carregamento dos mapas.

O volume de dados utilizado pode variar bastante de utilizador para utilizador, em função da área de visualização dos amigos e da quantidade de amigos do Facebook utilizadores do sistema. Desta forma não nos é possível quantificar o volume de dados consumido.

Para as componentes utilizadas no sistema foi possível otimizar o volume de dados transmitidos, apenas no processo de comunicação entre a aplicação móvel e o servidor MeOnMapServer, onde utilizamos os protocolos e formatos de comunicação otimizados para este tipo de equipamentos (*REST/JSON*). Contudo consideramos que com um maior esforço nesta componente, será possível implementar outros mecanismos e/ou processos no sentido de reduzir o volume de dados transmitidos, otimizando também o processamento e tempo de envio e recepção dos dados.

## 3.4 Aplicação MeOnMap

Este capítulo pretende descrever tecnicamente a aplicação MeOnMap, apresentando a arquitectura lógica da aplicação, e uma descrição genérica das principais funcionalidades e mecanismos de integração com as restantes componentes.

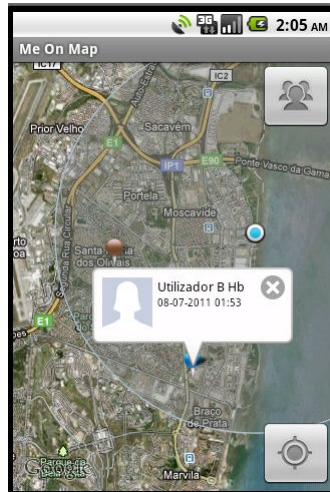


Imagem 4 – Interface de mapa da aplicação MeOnMap

A descrição mais funcional será descrita no anexo do manual de utilizador.

### 3.4.1 Arquitectura de componentes

Conforme referido anteriormente, a aplicação MeOnMap foi implementada utilizando a tecnologia Java, em conjunto com diversas API orientadas ao desenvolvimento para Android, nomeadamente a Android API 2.1, Android Maps API e a Facebook Android SDK.

Neste processo de implementação, fomos sendo confrontados com algumas dificuldades decorrentes da inexperiência no desenvolvimento de aplicações sobre estas API, e das particularidades inerentes ao próprio sistema. Algumas dessas particularidades estão relacionadas com os mecanismos de *multitask*, a forma de invocação dos interfaces (*activity*), os mecanismos para implementação e gestão de *threads* e eventos.

Conceptualmente a aplicação MeOnMap está dividida em diferentes componentes aplicativos, que podem ser organizados em camadas. A aplicação apresenta interfaces que comunicam com as *classes* da camada de negócio, que são “alimentadas” com os dados resultantes das chamadas aos Web services do MeOnMapServer e do Facebook. Adicionalmente existem algumas opções de configuração da aplicação, com impacto no funcionamento, que estão armazenadas numa base de dados da aplicação.

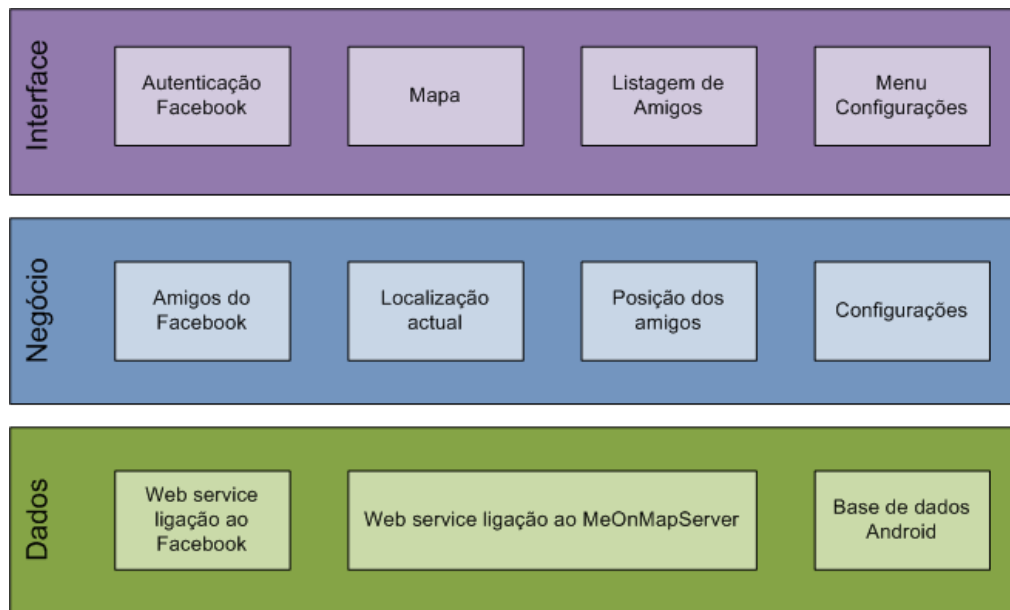


Imagem 5 – Arquitectura lógica

### 3.4.2 Descrição técnica

Ao iniciar a aplicação são executados alguns procedimentos, que permitem verificar o estado e disponibilidade de alguns componentes essenciais ao funcionamento da aplicação.

O primeiro grande passo passa por verificar se o Android tem acesso à Internet. Caso não tenha, será apresentada uma mensagem a informar o utilizador e a aplicação será fechada.

O passo seguinte verifica se o utilizador já se encontra autenticado no Facebook. Este processo de autenticação, foi implementado utilizando a API específica para dispositivos móveis, e implica a criação prévia de uma aplicação dentro do Facebook, utilizando as ferramentas para programadores do site. Nesse processo é atribuído um *ApplicationId* que deve ser utilizado para autenticação. O utilizador ao fazer login no MeOnMap, está a permitir que a aplicação tenha determinado nível de acesso aos seus dados de Facebook, sendo utilizado neste caso o nível mínimo. Ao utilizar a aplicação MeOnMap pela primeira vez, será solicitado ao utilizador a introdução do *username* e *password* de acesso ao Facebook, juntamente com a confirmação de utilização da aplicação e respectivos privilégios de acesso aos dados. As credenciais de acesso serão solicitadas apenas na primeira utilização da aplicação. Para os restantes acessos, e durante o



processo de autenticação, as credenciais são verificadas directamente sobre a API de Facebook, e caso ainda sejam válidas o utilizador faz login automático.

Durante a fase de testes da aplicação, tentamos dentro do possível, criar o maior número de cenários de utilização. Durante este período, foi disponibilizada uma nova versão da aplicação de Facebook para Android, que aparentemente utiliza um mecanismo de autenticação diferente do que era anteriormente usado, causando algumas dificuldades no processo de autenticação da aplicação MeOnMap junto do Facebook. Deduzimos que esta nova situação esteja relacionada com a disponibilização do mecanismo de autenticação via *Single Sign On*. A situação identificada ocorre, caso o utilizador ainda não esteja autenticado no Facebook, e ao entrar no MeOnMap com a introdução de credenciais, é remetido automaticamente para a aplicação Facebook ao invés da aplicação MeOnMap. Caso esteja autenticado (credenciais introduzidas via aplicação Facebook) ao aceder ao MeOnMap, ocorrem erros ao tentar aceder à informação, acabando por ser forçado a sair da aplicação MeOnMap. Embora esta situação tenha sido detectada, não foi possível adaptar o MeOnMap a esta nova realidade, em tempo útil. A anomalia identificada ocorre apenas aos utilizadores que tenham instalado a versão 1.6.0 ou superior da aplicação Facebook para Android, sendo assim requisito, a utilização de uma versão anterior.

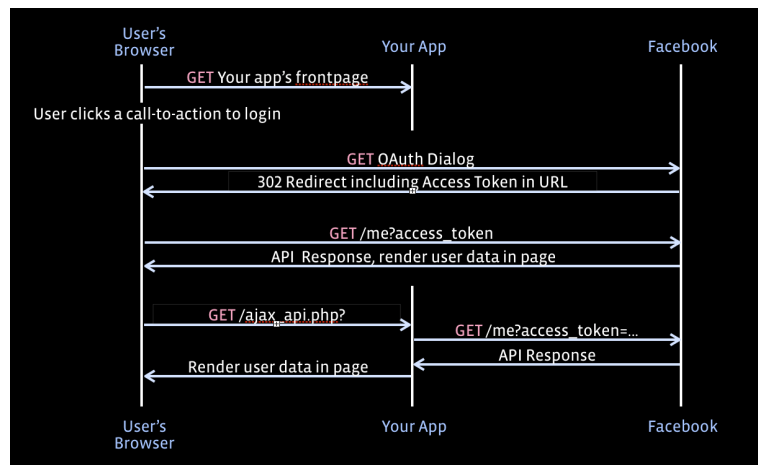


Imagem 6 – Mecanismo de autenticação API Facebook

Com o processo de autenticação concluído, o próximo passo é solicitar ao Facebook mais informação acerca desse utilizador. Esse pedido é feito utilizando a API e o protocolo de comunicação *REST*, sendo retornado um identificador único do Facebook correspondente ao utilizador autenticado. Esse identificador também será utilizado para identificar univocamente o utilizador em todo o sistema MeOnMap, sendo este o

elemento chave para todas as comunicações com o servidor MeOnMapServer. De seguida é solicitado ao Facebook, a lista de amigos do utilizador sendo retornado o nome e um identificador para cada um dos seus amigos.

Tendo a informação acerca do utilizador e dos seus amigos, essa é enviada através de um Web service ao MeOnMapServer (end point friends/get), que confronta a lista de amigos do Facebook, com a base de dados de utilizadores MeOnMap, já registados no sistema. A execução deste Web service devolve à aplicação MeOnMap, uma lista filtrada contendo apenas os utilizadores do sistema MeOnMap, e a informação acerca de quais os amigos que estão registados na blacklist do utilizador. Esta informação fica armazenada num *array* de objectos da aplicação, a ser utilizado posteriormente.

Durante a fase de arranque da aplicação, também é inicializada a componente de localização. Embora sejam opções de configuração do próprio equipamento, esta localização pode ser obtida através do sinal de GPS (Global Position System), através da rede GSM ou através da uma rede WiFi. O utilizador pode optar por escolher a melhor opção entre as fontes disponibilizadas, devendo considerar que o GPS oferece uma melhor precisão mas também um maior consumo de bateria. A utilização do sinal de localização através de GSM / rede WiFi oferecem uma menor precisão mas o consumo de bateria é mais moderado. A API de Android já disponibiliza métodos que permitem avaliar a precisão do sinal disponibilizado por cada uma das fontes, e em conjunto com a Android Maps API, fazer a representação em mapa dessa localização.



Imagem 7 – Configuração de mecanismo de localização do Android

Após obter uma coordenada de localização válida, e tendo a informação acerca de quais os amigos de Facebook, é possível submeter essa informação ao servidor

MeOnMapServer (end point friends/position), para que a mesma seja processada. A lista a enviar, será sempre a lista de amigos do Facebook, e não a lista de amigos utilizadores da aplicação. Desta forma garante-se a independência entre o sistema MeOnMap e a rede social, pois caso um amigo de Facebook comece a utilizar a aplicação, o utilizador tem acesso imediato à sua localização, sem necessidade de actualização da lista de amigos de Facebook. Como resposta ao envio da localização do utilizador para o MeOnMapServer, é feito o retorno da lista de amigos utilizadores do sistema que estão dentro do raio de visualização definido e que o utilizador tem permissões para visualizar. Cada elemento dessa lista tem associada uma coordenada de localização, uma data e hora, e um estado, sendo feita a sua representação em mapa, aplicando diferentes simbologias em função do estado (consultar 2.3.3 Estado dos amigos). Dependendo das configurações utilizadas, a recepção desta lista de amigos poderá dar origem à criação de uma notificação, conforme descrito no capítulo de conceitos 2.3.6 Notificações.

Esta acção de envio da coordenada de localização do utilizador e a recepção das coordenadas de localização dos seus amigos ocorre periodicamente, em função de um valor de tempo configurado pelo utilizador.

Sobre o interface de mapa existe um botão onde são listados todos os amigos utilizadores do sistema. A selecção de um elemento desta listagem vai solicitar ao servidor um pedido de localização para esse amigo, sendo essa localização devolvida, mesmo que o amigo se encontre fora do raio de visualização definido.

A selecção em mapa da localização de um amigo, permite obter mais alguma informação acerca do mesmo, através de uma janela informativa onde é apresentado o nome do amigo, a data e hora da coordenada visualizada e a fotografia actualmente utilizada na rede social Facebook. O carregamento desta fotografia é feito através da geração de um URL utilizando o identificador de amigo.

Existem algumas configurações da aplicação que o utilizador pode definir. Essas ficarão armazenadas sobre uma base de dados que é o SQL Lite. Este motor de base de dados é orientado à utilização em dispositivos móveis.

Todas as mensagens e textos utilizados na aplicação foram configurados num ficheiro de *localization* da API de Android. Caso existam outras versões de ficheiros, em outras línguas, as mensagens serão apresentadas na língua correspondente à definida no equipamento. Actualmente existe apenas uma configuração que é em língua inglesa.

### 3.4.3 Workflow utilizado

De acordo ao descrito no capítulo anterior apresentamos um esquema representativo do Workflow aplicacional utilizado. Este não pretende fazer uma representação técnica em UML dos processos endereçados pela aplicação, mas sim uma ilustração explicativa para facilitar a compreensão.

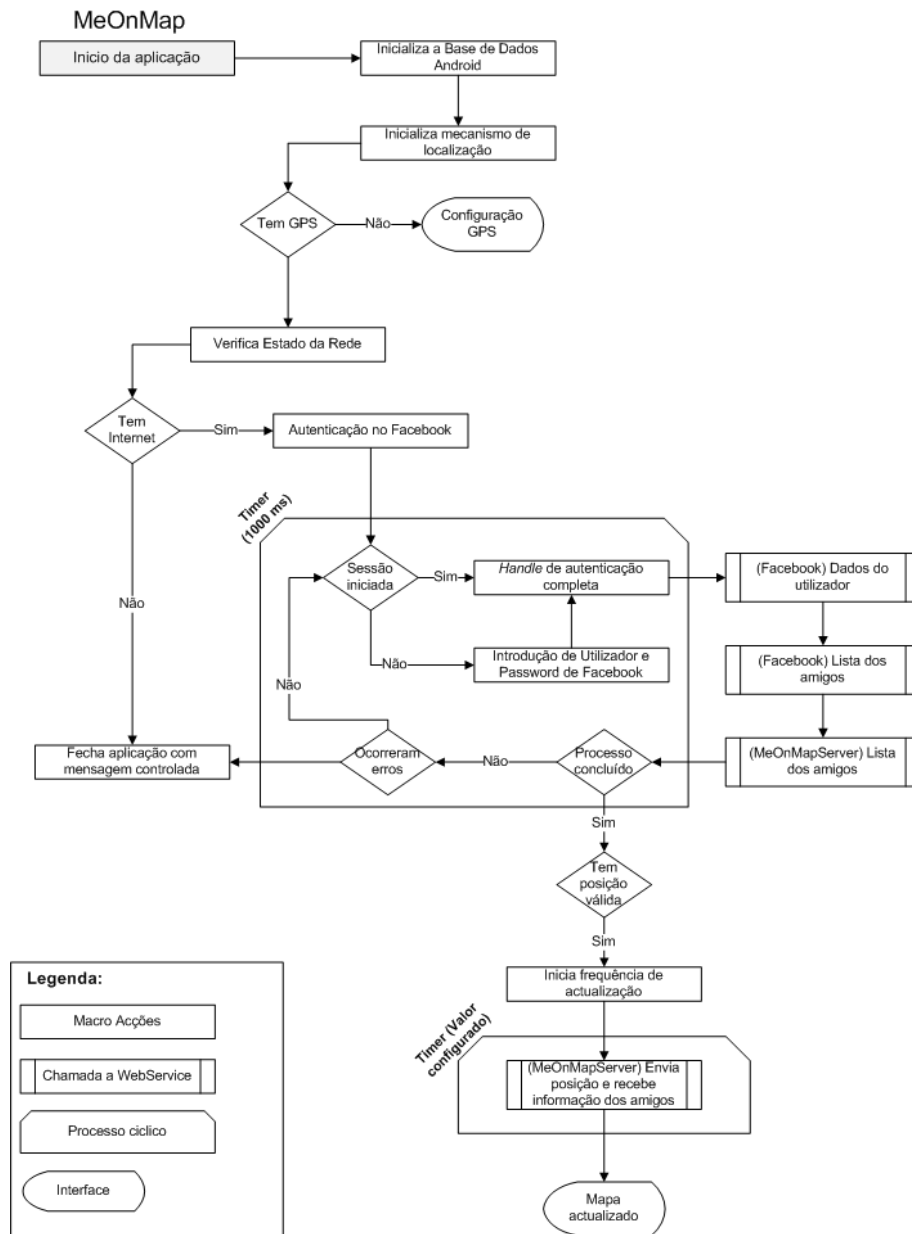


Imagem 8 – Representação do Workflow em esquema

## 4 Conclusões e trabalho futuro

A execução deste trabalho de final de curso, permitiu uma aprendizagem e um reforçar de conhecimentos e conceitos associados às tecnologias envolvidas. Conforme foi sendo descrito ao longo do documento, este projecto é composto por duas grandes componentes, uma no contexto de uma aplicação servidor e outra numa vertente de mobilidade. Adicionalmente é ainda introduzido o conceito de consumo de dados oriundos da internet via Facebook.

Com o aparecimento e vulgarização da informática, as empresas de TI focaram os seus esforços na criação e desenvolvimento de soluções essencialmente orientadas ao mercado empresarial, criando ferramentas para ambientes desktop. O grande desafio era o baixar custos e aumentar a produtividade das empresas cliente, garantindo assim obviamente, a continuidade na implementação de projectos. Entretanto com o aumento na velocidade das comunicações, a Internet passou a ser o grande foco, e as empresas de TI tiveram de se adaptar a essa realidade. Os sistemas de informação mudaram, e em complemento aos sistemas já existentes, existiu a necessidade de os expandir acrescentando-lhes uma vertente que passa pela Internet. Actualmente, estamos perante novas mudanças no processo de evolução dos sistemas de informação. O expandir da *cloud*, o aumento na capacidade de armazenamento e processamento dos equipamentos móveis, e o baixo custo da tecnologia fomentam o aparecimento de ferramentas e produtos de grande consumo, orientados ao mercado dos utilizadores particulares. Com isto, surgem os novos desafios, nomeadamente ao nível da publicidade, que assentam em oferecer ao consumidor final exactamente o que ele pretende, quase de uma forma personalizada. Com a utilização de tecnologias e mecanismos para a localização do utilizador, surge o conceito de LBS (*Location Based Services*), que na prática é a disponibilização de serviços orientados ao utilizador em função da sua localização geográfica, como por exemplo saber qual a caixa multibanco ou os restaurantes mais próximos.

Ao juntar estas novas tendências e conceitos, ao “boom” das redes sociais, no caso o Facebook, estamos perante um produto muito actual e com um elevado potencial de crescimento, quer ao nível de utilizadores quer ao nível de funcionalidades, que hipoteticamente pode vir a ser utilizado num mercado global e de fácil distribuição, como é o mercado dos *smartphones*.

Embora o MeOnMap implemente já algumas funcionalidades, ainda não atinge o estado de maturidade necessário para poder “competir” nesse mercado. Desta forma, ainda o consideramos como um projecto protótipo, que necessita de algum esforço na implementação e optimização de funcionalidades, na definição de um plano de acções para uma bateria de testes real, da avaliação dos resultados obtidos através desses testes, e eventualmente da definição de um *product roadmap*, de forma a minimizar os riscos da introdução de um novo produto no mercado.

Neste projecto, existem duas zonas de grande risco relativamente à performance do sistema. O modo de funcionamento do sistema faz com que cada utilizador esteja a fazer o envio e recepção de dados com bastante frequência. A localização de um utilizador (input de dados) é parte da informação que é distribuída aos restantes utilizadores (output de dados), tendo assim um sistema de dados em cadeia, onde cada utilizador gera a informação para os restantes utilizadores. O volume de dados transaccionado depende directamente da quantidade de utilizadores do sistema, e todos vão enviar e receber dados através do servidor MeOnMapServer, tornando este o ponto crítico do sistema, onde (em situação futura) devem ser considerados sistemas redundantes para a componente aplicacional e de base de dados. Aplicacionalmente existem algumas melhorias a serem implementadas nesta componente, nomeadamente no sentido de reduzir o número de acessos feitos à base de dados. Outro ponto crítico no que toca à performance do sistema tem a ver com a componente de mobilidade, e para os casos em que o utilizador tem um elevado número de amigos, é necessário fazer a gestão dos dados, processar estados e localizações de amigos, num equipamento que ainda tem capacidades de processamento e memória moderadas. Ainda na componente de mobilidade, poderá justificar-se avaliar mecanismos de optimização para o processo de envio e recepção de dados.

No processo evolutivo para a consolidação em produto, e dada a particular natureza e variedade dos equipamentos a utilizar, será necessário efectuar uma bateria de testes bastante diversificada quer ao nível de equipamentos, quer ao nível da validação e exploração de funcionalidades. Estes testes devem compreender acções com diferentes modelos de equipamentos e de sistema operativo, avaliando a estabilidade da aplicação, métricas de consumo de bateria e dados e eventuais incompatibilidades com outras aplicações. Funcionalmente é necessário identificar um padrão na utilização da aplicação, avaliando os hábitos de utilização e funcionalidades mais utilizadas. A primeira fase de testes / experimentação, deve ser realizada por um número restrito de utilizadores, permitindo assim a avaliação destes critérios e ajuda na identificação das funcionalidades evolutivas do sistema, podendo estas vir a ser mais orientadas à

utilização genérica e massiva pelos utilizadores, ou mais orientadas a um determinado nicho de mercado.

No caso de uma utilização massiva, existem algumas precauções a ter em conta, nomeadamente com o número de acessos à API do Facebook, conforme descrito nas políticas de utilização: “*If you exceed, or plan to exceed, any of the following thresholds please contact us as you may be subject to additional terms: (>5M MAU) or (>100M API calls per day) or (>50M impressions per day).*”. Relativamente à utilização da API do Google Maps não foi identificada nenhuma restrição significativa, não obstante de uma leitura e análise mais cuidada das suas políticas de utilização. Deve ser considerada a criação de mecanismos que assegurem a protecção e privacidade dos dados do utilizador, bem como a criação de políticas de utilização do sistema. À semelhança de outros sistemas, uma dessas políticas poderá ser o aceitar que, a informação produzida pela aplicação venha a ser analisada (de forma não individualizada). Com esses dados, será possível fazer análises e extrapolações aos mesmos, de forma a calcular tendências com vista à produção de acções de marketing ou estudos de mercado.

Para atingir este objectivo, é necessário que exista um volume significativo de dados e utilizações, mas para tal, o sistema tem de evoluir no sentido de atrair e cativar muitos utilizadores. Desta forma, e como visão de produto, lançamos algumas ideias genéricas à evolução do mesmo:

- Apresentação dos amigos em função da área e escala de mapa apresentada (em substituição do raio de visualização);
- Maior integração com as funcionalidades do Facebook (ex: envio de mensagens privadas, actualização do estado e localização);
- Possibilidade de tirar fotografias fazendo upload das mesmas para o Facebook. Na aplicação MeOnMap seria apresentado um novo *layer* no mapa com a localização das fotografias que foram tiradas;
- Avaliar a integração com outras redes sociais (ex: *LinkedIn*, *Twitter*, *MySpace*);
- Integração com aplicações de realidade aumentada (ex: *Layar*);
- Possibilidade de acesso a conteúdos (ex: publicidade) orientada ao perfil do utilizador, com base na sua localização;
- Partilha da localização do utilizador com outros sistemas de âmbito profissional (ex: serviços de distribuição de encomendas podem localizar o destinatário);
- Evolução para outras plataformas (ex: *Apple iOS*, *Windows Mobile*, *Blackberry*);

Assim, e fazendo um ponto de situação acerca do trabalho já realizado, actualmente temos um protótipo funcional que implementa um conjunto de ideias e conceitos interessantes, com um potencial de utilizadores e eventual retorno financeiro considerável, mas que nesta fase, ainda necessita de algum esforço no sentido de o transformar em produto.



## 5 Bibliografia

Android API - <http://developer.android.com/sdk/index.html>

Android Maps - <http://code.google.com/intl/pt/android/add-ons/google-apis/reference/index.html>

Android Maps Key - <http://code.google.com/intl/pt-PT/android/maps-api-signup.html>

Android Maps Balloons - <https://github.com/jgilfelt/android-mapviewballoons>

Facebook API - <http://developers.facebook.com/docs/sdks/> ;  
<http://developers.facebook.com/policy/#policies>

Java Docs - <http://download.oracle.com/javase/6/docs/api/>

Web Services - <http://www.coderanch.com/how-to/java/WebServicesFaq>

Java Practices - <http://www.java practices.com/home/HomeAction.do>

Protocolo Rest - <http://www.vogella.de/articles/REST/article.html>

Base de dados SQLite – <http://www.reigndesign.com/blog/using-your-own-sqlite-database-in-android-applications/>

Carregamento dinâmico de listas em Android - <http://www.androidkit.com/loading-images-from-remote-server-over-http-on-a-separate-thread> ; <http://appfulcrum.com/?p=281> ;  
<http://www.vogella.de/articles/AndroidListView/article.html>

Organização de Layouts em Android - <http://mobiforge.com/designing/story/understanding-user-interface-android-part-1-layouts>

Tomcat - <http://tomcat.apache.org/>

Cálculo de distância com coordenadas - <http://www.movable-type.co.uk/scripts/latlong.html>

Base de dados PostgreSQL - <http://www.postgresql.org/> ; <http://postgis.refrains.net/>