



Projeto de Mestrado em
Engenharia Informática - Computação Móvel

INTEGRAÇÃO DE TECNOLOGIAS MÓVEIS NOS TRANSPORTES PÚBLICOS

Pedro José de Sousa Ferreira

Leiria, Março de 2015



Projeto de Mestrado em
Engenharia Informática - Computação Móvel

INTEGRAÇÃO DE TECNOLOGIAS MÓVEIS NOS TRANSPORTES PÚBLICOS

Pedro José de Sousa Ferreira

Projeto de Mestrado realizado sob a orientação da Doutora Catarina Helena Branco Simões da Silva e do Doutor Luís Filipe Fernandes Silva Marcelino, ambos professores da Escola Superior de Tecnologia e Gestão do Instituto Politécnico de Leiria.

Leiria, Março de 2015

Agradecimentos

Apesar de se tratar de um trabalho proposto a nível individual, é injusto que não se deixem os devidos agradecimentos àqueles que foram contribuindo para o sucesso do mesmo. Como tal, deixo os meus agradecimentos:

À Professora Catarina Helena Branco Simões da Silva e ao Professor Luís Filipe Fernandes Silva Marcelino, ambos meus orientadores, pelo apoio e disponibilidade demonstrada. Agradecer-lhes também pelas críticas, correções e sugestões feitas durante os desenvolvimentos deste trabalho.

A todos os outros professores que lecionam no Mestrado em Engenharia Informática - Computação Móvel na Escola Superior de Tecnologia e Gestão do Instituto Politécnico de Leiria que fizeram parte desta etapa, não só no desenvolvimento deste trabalho, como também no primeiro ano deste mesmo Mestrado.

Às empresas de transporte público que prontamente se mostraram disponíveis para responder ao questionário efetuado.

Aos meus pais e ao meu irmão pelo apoio demonstrado e pela força transmitida em todos os momentos.

Aos meus amigos e colegas que tanto me ouviram falar deste trabalho, umas vezes com entusiasmo, outras nem tanto, mas que, de certo modo, foram essenciais para o levar a bom porto.

Ao Instituto Politécnico de Leiria, à Escola Superior de Tecnologia e Gestão e ao Departamento de Engenharia Informática desta instituição.

Esta página foi intencionalmente deixada em branco.

Resumo

Hoje em dia a tecnologia evolui a um ritmo alucinante, estando cada vez mais presente no dia-a-dia da população em geral. As tecnologias móveis não são exceção e, ao longo dos anos, têm vindo a emergir fazendo com que a população seja, no geral, portadora de um ou mais dispositivos móveis. Por outro lado, e ao contrário do que seria de esperar, estão algumas empresas de transportes públicos que não têm forma de divulgar informação sobre os seus serviços aos passageiros. Como tal, no sentido de auxiliar estas empresas e os seus passageiros e de contribuir para o aumento da utilização e eficácia dos transportes públicos e, conseqüentemente, para a melhoria das condições ambientais, foi desenhada e implementada uma solução que, através da conjugação de várias tecnologias, permite a integração de diferentes serviços que, divulgados aos passageiros através de uma aplicação cliente, lhes garante uma maior rentabilização do seu tempo pessoal.

Palavras-chave: tecnologia, transportes públicos, população, dispositivos móveis, aplicação, passageiros

Esta página foi intencionalmente deixada em branco.

Abstract

Nowadays, technology progresses at a remarkable pace, as it is becoming more and more present in everyday activities. Mobile technologies are not an exception and have emerged over the years, causing the general population to have at least one portable device. On the other hand, and contrary to what was expected, there are some public transportation companies that do not have the means to disclose their services to the passengers. As such, in the regards of helping this companies and their passengers, contributing to the growing use and efficiency of public transportation, therefore, improvement of environmental conditions, it was designed and implemented a solution that, through the combination of several technologies, allows the integration of different services that when released to passengers through a dedicated application grants them a better use of their personal time.

Keywords: technology, public transport, population, mobile devices, application, passengers

Esta página foi intencionalmente deixada em branco.

Lista de Figuras

| | | |
|------|---|----|
| 2.1 | Funcionamento do sistema <i>LisbonBus</i> | 8 |
| 2.2 | Arquitetura e funcionamento do sistema <i>Transit Trip Planner</i> | 9 |
| 2.3 | Interface da Aplicação One Bus Away | 11 |
| 2.4 | Interface da Aplicação Bongo | 12 |
| 2.5 | Interface da Aplicação UK Bus Checker | 13 |
| 2.6 | Interface da Aplicação Live London Bus Tracker | 14 |
| 2.7 | Interface da aplicação MOVE-ME | 15 |
| 2.8 | Interface da Aplicação IZI Carris | 16 |
| 2.9 | Interface da Aplicação Sapo Transportes | 17 |
| 2.10 | Interface do Serviço Transportes Publicos.pt com base no <i>OpenTripPlanner</i> | 20 |
| | | |
| 3.1 | Diagrama de Entidade Relacionamento | 39 |
| 3.2 | Arquitetura do sistema - Visão de alto nível | 41 |
| 3.3 | Arquitetura do sistema - Integração de Serviços | 46 |
| | | |
| 4.1 | Validação de Dados com <i>FeedValidator</i> | 57 |
| 4.2 | Instalação da API Node-GTFS | 58 |
| 4.3 | Ligação aos serviços GTFS | 59 |
| 4.4 | Sistema de Alertas em Execução | 60 |
| 4.5 | Criação de Alerta | 61 |
| 4.6 | Alerta em formato GTFS RT | 62 |
| 4.7 | Fluxo de criação de um alerta | 62 |
| 4.8 | Criação de uma ligação às Google API's | 65 |
| 4.9 | Modelo de Dados | 68 |
| | | |
| 5.1 | Menu Principal e Inicio | 72 |
| 5.2 | Paragens I | 73 |

| | | |
|------|---|-----|
| 5.3 | Paragens II | 74 |
| 5.4 | Rotas | 75 |
| 5.5 | Cálculo de Percursos | 76 |
| 5.6 | Horários | 77 |
| 5.7 | Alertas | 78 |
| 5.8 | Favoritos | 79 |
| 5.9 | Empresas | 80 |
| 5.10 | Pontos de Interesse | 81 |
| 5.11 | Preferências | 82 |
| 5.12 | Testes Funcionais - Resultados | 85 |
| 5.13 | Testes de Carga - Tempo de Obtenção de Informação | 87 |
| | | |
| A.1 | Interface da Aplicação UK Bus Checker | 99 |
| A.2 | Interface da Aplicação Live London Bus Tracker | 100 |
| A.3 | Interface da Aplicação New Madrid Metro - Bus - Cercanias | 101 |
| A.4 | Interface da Aplicação Bus K Fortaleza | 103 |
| A.5 | Interface da Aplicação Bus Brazil | 104 |
| A.6 | Interface da Aplicação Moobly | 105 |
| A.7 | Interface da Aplicação RATP: Paris | 106 |
| A.8 | Interface da Aplicação IZI Carris | 108 |
| A.9 | Interface da Aplicação Sapo Transportes | 109 |
| A.10 | Interface da Aplicação NUS NextBus | 110 |
| A.11 | Interface da aplicação MOVE-ME | 111 |
| A.12 | Interface da Aplicação Live London Countdown | 113 |
| A.13 | Interface da Aplicação NextBuses | 114 |
| A.14 | Interface da Aplicação iNextBus | 115 |
| A.15 | Interface da Aplicação InTime NextBus | 116 |
| A.16 | Interface da Aplicação SG NextBus | 117 |
| A.17 | Interface da Aplicação NUS NextBus | 118 |
| A.18 | Interface da Aplicação RATP:Paris | 119 |
| A.19 | Interface da Aplicação UK Bus Checker | 121 |
| A.20 | Interface da Aplicação Countdown | 123 |
| A.21 | Interface da Aplicação RATP | 124 |
| A.22 | Interface da Aplicação Buses Due | 125 |
| A.23 | Interface da Aplicação BH Buses | 126 |

| | |
|--|-----|
| A.24 Interface da Aplicação Déjà-bus | 127 |
| A.25 Interface da Aplicação Next Bus UK Live! | 128 |
| A.26 Interface da Aplicação London Bus Checker | 129 |
| A.27 Interface da Aplicação Bus Nearby | 130 |

Esta página foi intencionalmente deixada em branco.

Lista de Tabelas

| | | |
|------|---|-----|
| 2.1 | Funcionalidades das Aplicações Analisadas | 18 |
| 2.2 | Funcionalidades das Infra-Estruturas de Partilha de Informação Analisadas | 28 |
| 3.1 | Requisitos Funcionais de Paragens | 33 |
| 3.2 | Requisitos Funcionais de Rotas | 34 |
| 3.3 | Requisitos Funcionais de Favoritos | 35 |
| 3.4 | Outros Requisitos Funcionais | 36 |
| 3.5 | Requisitos Não Funcionais | 37 |
| 4.1 | Ferramentas GTFS | 55 |
| A.1 | Funcionalidades mais comuns | 131 |
| A.2 | Outras funcionalidades | 132 |
| A.3 | Estatísticas das aplicações analisadas | 133 |
| C.1 | agency.txt | 139 |
| C.2 | stops.txt | 140 |
| C.3 | routes.txt | 141 |
| C.4 | trips.txt | 142 |
| C.5 | stop_times.txt | 143 |
| C.6 | calendar.txt | 144 |
| C.7 | calendar_dates.txt | 145 |
| C.8 | fare_attributes.txt | 146 |
| C.9 | fare_rules.txt | 146 |
| C.10 | shapes.txt | 147 |
| C.11 | frequencies.txt | 147 |
| C.12 | transfers.txt | 148 |
| C.13 | feed_info.txt | 149 |

| | | |
|------|--------------------------|-----|
| D.1 | <i>FeedMessage</i> | 151 |
| D.2 | <i>FeedMessage</i> | 151 |
| D.3 | <i>FeedHeader</i> | 152 |
| D.4 | <i>FeedEntity</i> | 152 |
| D.5 | <i>TripUpdate</i> | 153 |
| D.6 | <i>StopTimeEvent</i> | 153 |
| D.7 | <i>StopTimeUpdate</i> | 154 |
| D.8 | <i>VehiclePosition</i> | 155 |
| D.9 | <i>VehiclePosition</i> | 156 |
| D.10 | <i>Alert</i> | 157 |
| D.11 | <i>TimeRange</i> | 158 |
| D.12 | <i>Position</i> | 158 |
| D.13 | <i>TripDescriptor</i> | 159 |
| D.14 | <i>VehicleDescriptor</i> | 160 |
| D.15 | <i>EntitySelector</i> | 160 |
| D.16 | <i>TranslatedString</i> | 161 |

Esta página foi intencionalmente deixada em branco.

Acrónimos

| Acrónimo | Significado |
|-----------------|------------------------------------|
| ADB | Android Debug Bridge |
| ADT | Android Developer Tools |
| API | Application Programming Interface |
| ARM | Advanced RISC Machine |
| DER | Diagrama Entidade Relacionamento |
| GIS | Geographic Information System |
| GPS | Global Positioning System |
| GTFS | General Transit Feed Specification |
| HTTP | Hypertext Transfer Protocol |
| JSON | JavaScript Object Notation |
| MVC | Model-View-Controller |
| MVP | Model-View-Presenter |
| OSM | OpenStreetMap |
| OTP | OpenTripPlanner |
| PaaS | Platform as a Service |
| REST | Representational State Transfer |

| Acrónimo | Significado |
|-----------------|----------------------------|
| SDK | Software Development Kit |
| SDL | Simple DirectMedia Layer |
| SQL | Structured Query Language |
| URL | Uniform Resource Locator |
| USB | Universal Serial Bus |
| XML | Extensible Markup Language |

Esta página foi intencionalmente deixada em branco.

Conteúdo

| | | |
|----------|--|----------|
| 1 | Introdução | 1 |
| 1.1 | Enquadramento | 1 |
| 1.2 | Motivação e Objetivos | 2 |
| 1.3 | Estrutura do Documento | 2 |
| 2 | Estado da Arte | 5 |
| 2.1 | Infra-Estruturas de Obtenção de Informação | 5 |
| 2.1.1 | <i>Where is My Bus?</i> | 5 |
| 2.1.2 | <i>How Long To Wait?</i> | 6 |
| 2.1.3 | <i>Lisbon Bus System: Retrieving Real-Time Information</i> | 8 |
| 2.1.4 | <i>Transit Trip Planner With Real Time Data</i> | 9 |
| 2.1.5 | <i>OneBusAway</i> | 10 |
| 2.1.6 | <i>Bongo - Bus on The Go</i> | 11 |
| 2.1.7 | <i>UK Bus Checker</i> | 12 |
| 2.1.8 | <i>Live London Bus Tracker</i> | 13 |
| 2.1.9 | <i>MOVE-ME</i> | 14 |
| 2.1.10 | <i>IZI Carris</i> | 15 |
| 2.1.11 | <i>Sapo Transportes</i> | 16 |
| 2.1.12 | <i>Análise Geral</i> | 17 |
| 2.2 | Infra-Estruturas de Partilha de Informação | 19 |
| 2.2.1 | <i>OpenTripPlanner</i> | 19 |
| 2.2.2 | <i>OneStopTransport</i> | 21 |
| 2.2.3 | <i>Transport API</i> | 21 |
| 2.2.4 | <i>Chicago Transit Authority API</i> | 22 |
| 2.2.5 | <i>OpenStreetMap</i> | 22 |
| 2.2.6 | <i>GTFS</i> | 23 |

| | | |
|----------|---|-----------|
| 2.2.7 | <i>GTFS - RealTime</i> | 24 |
| 2.2.8 | <i>Walk Score Public Transit API</i> | 26 |
| 2.2.9 | <i>Transport - Swiss Public Transport API</i> | 27 |
| 2.2.10 | Análise Geral | 27 |
| 2.3 | Conclusão | 29 |
| 3 | Abordagem Proposta | 31 |
| 3.1 | Introdução | 31 |
| 3.2 | Solução Proposta | 32 |
| 3.2.1 | Requisitos de Sistema | 32 |
| 3.2.2 | Arquitetura | 39 |
| 3.2.3 | Cliente | 41 |
| 3.2.4 | Servidor | 42 |
| 3.2.5 | Abordagem Escolhida | 45 |
| 3.2.6 | Detalhe e Integração de Sistemas | 45 |
| 3.3 | Conclusão | 49 |
| 4 | Implementação | 51 |
| 4.1 | Tecnologias | 51 |
| 4.1.1 | Servidor | 51 |
| 4.1.2 | Cliente | 63 |
| 4.2 | Metodologia de Desenvolvimento | 65 |
| 4.2.1 | MVC | 66 |
| 4.2.2 | Modelo de Dados | 67 |
| 4.3 | Conclusão | 68 |
| 5 | Solução Final | 71 |
| 5.1 | Cliente | 71 |
| 5.1.1 | Menu Principal | 71 |
| 5.1.2 | Início | 72 |
| 5.1.3 | Paragens | 73 |
| 5.1.4 | Rotas | 74 |
| 5.1.5 | Calcular Percurso | 75 |
| 5.1.6 | Horários | 76 |
| 5.1.7 | Alertas | 77 |
| 5.1.8 | Favoritos | 78 |

| | | |
|----------|--------------------------------------|------------|
| 5.1.9 | Empresas | 79 |
| 5.1.10 | Pontos de Interesse | 80 |
| 5.1.11 | Preferências | 81 |
| 5.2 | Testes | 82 |
| 5.2.1 | Funcionalidades | 82 |
| 5.2.2 | Conetividade | 85 |
| 5.2.3 | Carga | 86 |
| 5.3 | Conclusões | 87 |
| 6 | Conclusão | 89 |
| 6.1 | Trabalho Futuro | 90 |
| A | Análise de Aplicações | 97 |
| A.1 | Aplicações Existentes | 97 |
| A.1.1 | Android | 97 |
| A.1.2 | iOS | 111 |
| A.1.3 | Windows Phone | 122 |
| A.1.4 | Análise Geral | 130 |
| B | Protótipos da Solução | 137 |
| C | Especificação GTFS | 139 |
| D | GTFS - Realtime | 151 |
| E | Especificação GTFS - Mobilis | 163 |
| E.0.5 | <i>agency.txt</i> | 163 |
| E.0.6 | <i>calendar.txt</i> | 163 |
| E.0.7 | <i>calendar_dates.txt</i> | 163 |
| E.0.8 | <i>fare_attributes.txt</i> | 164 |
| E.0.9 | <i>fare_rules.txt</i> | 164 |
| E.0.10 | <i>feed_info.txt</i> | 164 |
| E.0.11 | <i>frequencies.txt</i> | 165 |
| E.0.12 | <i>frequencies.txt</i> | 165 |
| E.0.13 | <i>trips.txt</i> | 165 |
| E.0.14 | <i>routes.txt</i> | 166 |
| E.0.15 | <i>shapes.txt</i> | 166 |

| | |
|--|------------|
| E.0.16 <i>stop_times.txt</i> | 166 |
| E.0.17 <i>stops.txt</i> | 171 |
| F Script de Alteração à Referência GTFS Online | 175 |
| G Estrutura Aplicação Cliente | 177 |
| H Inquérito Realizado às Empresas de Transporte | 179 |

Capítulo 1

Introdução

1.1 Enquadramento

Hoje em dia, a densidade populacional nas cidades tem tendência a aumentar e este fator combinado com as dificuldades económicas que cada vez mais se fazem sentir, levam a que os transportes públicos vão ganhando algum destaque. São inúmeras as vantagens associadas a este tipo de transportes tanto em termos económicos como em termos ambientais. O facto de estarem disponíveis a preços bastante acessíveis e de garantirem conforto e segurança aos passageiros fazem com que sejam eleitos por muitos a principal forma de efetuar deslocações diárias. No entanto, estão também presentes vantagens associadas ao ambiente e à sua sustentabilidade uma vez que a utilização deste meio de transporte implica uma redução do número de veículos nas cidades e, conseqüentemente, contribui para a redução da poluição, melhorando também a qualidade de vida nas mesmas.

No mesmo sentido de crescimento do uso de transportes públicos está a utilização de dispositivos móveis, *smartphones*, por parte das pessoas [1]. Assim, da conjugação destes dois fatores surgiu a necessidade do estudo e implementação de uma solução que auxiliasse as empresas na partilha da sua informação com os seus passageiros. Desta informação divulgada pelas empresas constam dados de rotas, paragens, informação institucional, entre outros. Ao mesmo tempo, e para além de auxiliar as empresas, o objetivo é que a solução facilitasse também a vida aos passageiros através da disponibilização de uma forma de consumir e apresentar a informação referida, através de uma aplicação cliente.

1.2 Motivação e Objetivos

A disponibilização de informação aos passageiros por parte das empresas de transportes públicos é, hoje em dia e na maioria dos casos, praticamente nula. Para as exceções existentes, a informação está disponível muitas vezes em paragens ou até mesmo nos próprios autocarros (ver Anexo G). Como se sabe, informação deste género e neste contexto carece de atualização constante e, através desta forma de publicação, é praticamente impossível conseguir manter atualizada, dificultando assim o trabalho das entidades prestadoras dos serviços de transporte.

Assim sendo, e tendo em conta a importância tanto para estas entidades como para os passageiros de divulgar e obter a informação o mais atualizada possível, respetivamente, o presente trabalho tem como objetivo principal permitir às empresas uma aproximação aos seus passageiros, através de um sistema de divulgação de informação atualizada, a qualquer hora, em qualquer lugar. De modo a que essa aproximação seja conseguida, é fundamental aproveitar a elevada penetração das tecnologias - *smartphones* - no mercado [2] e, como tal, desenhar uma solução à medida dos objetivos propostos é o ponto chave para dar resposta a este problema.

Assim, a solução proposta e resultante do estudo efetuado, tem o intuito de divulgar a informação sobre os transportes públicos através da integração de várias fontes de dados e disponibilizá-la àqueles que fazem uso deste meio de transporte e que, ao mesmo tempo, possuam um dispositivo móvel. Esta solução será difundida através de uma arquitetura cliente-servidor, onde o servidor será responsável por divulgar a informação e o cliente, o uso e a descodificação da mesma para que seja apresentada ao utilizador. A informação é tudo aquilo que é útil apresentar ao utilizador, como por exemplo horários, paragens e rotas, entre outros.

1.3 Estrutura do Documento

O presente documento é constituído por seis capítulos ao longo dos quais são relatados os passos e decisões tomadas aquando do estudo, formulação e implementação da solução proposta.

O primeiro capítulo, onde está inserida esta secção, para além de enquadrar o estudo realizado, apresenta as motivações e objetivos definidos que darão resposta ao problema apresentado.

O capítulo 2, o Estado da Arte, aborda o que tem sido feito, estudado e trabalhado nesta

temática, dando assim uma visão dos esforços que têm vindo a ser realizados, ao longo dos tempos, para colmatar os problemas existentes nesta área. Assim, neste capítulo, serão apresentadas pesquisas científicas, análises de infra-estruturas de partilha e de obtenção de informação.

O terceiro capítulo apresenta a abordagem proposta. Entende-se por abordagem proposta, a descrição pormenorizada da solução a implementar que vem complementar os objetivos propostos.

De seguida, o quarto capítulo representa a forma e os detalhes de implementação tomados para que fosse possível a colocação em prática da abordagem proposta, através da apresentação das tecnologias e métodos utilizados para o efeito.

No capítulo 5 é apresentada a aplicação cliente final que será responsável por apresentar a informação divulgada pela empresa de transportes ao passageiro. Neste capítulo serão também analisados os testes efetuados à aplicação.

Por último, no capítulo 6 são apresentadas conclusões retiradas do estudo e implementação deste trabalho, bem como as melhorias a efetuar num trabalho futuro.

Capítulo 2

Estado da Arte

O presente capítulo tem como principal objetivo documentar o que já foi estudado e desenvolvido na área dos transportes a fim de obter uma perspectiva mais abrangente acerca do que existe atualmente em funcionamento. Assim, será apresentado o estudo e a análise de algumas infra-estruturas de partilha e obtenção de informação e aplicações existentes nas diversas lojas *online* com o objetivo de conhecer as funcionalidades de cada uma destas, permitindo assim saber quais as mais importantes e, consequentemente, as que melhor contribuem para a implementação da solução proposta.

2.1 Infra-Estruturas de Obtenção de Informação

No que diz respeito aos transportes públicos e no sentido de contribuir para a melhoria das infra-estruturas existentes para obtenção de informação, têm sido realizadas algumas pesquisas científicas que de seguida se apresentam. Para além disso, embora em crescimento mas não sendo tão comum, estão as aplicações criadas por entidades prestadoras de serviço de transportes públicos sob as quais é interessante destacar alguns pontos.

2.1.1 *Where is My Bus?*

Kari Edison Watkins, Brian Ferris, Alan Borning, G. Scott Rutherford e David Layton são autores de uma pesquisa científica [3] que surgiu da observação de utilizadores de transportes públicos, nomeadamente autocarros, nas paragens de Seattle, nos Estados

Unidos da América. De modo a efetuar o seu estudo, os autores abordaram os passageiros com o intuito de obter uma estimativa do tempo que os mesmos achavam que teriam de aguardar até que chegasse o próximo autocarro. O resultado final levou à conclusão de que os passageiros sem a noção do tempo de espera tinham a ideia que este seria superior aquele de que na realidade era o tempo real do próximo autocarro a chegar à paragem. Neste estudo foi ainda referido que os utilizadores com a informação real do tempo de espera, ao se deslocarem para a paragem e aguardarem pelo autocarro, passaram a esperar da média de nove minutos e noventa segundos (sem qualquer informação) para sete minutos e cinquenta segundos, sendo esta uma diferença de cerca de 30%. Posto isto, confirmou-se que o acesso à informação em tempo real não só reduz o tempo de espera esperado, mas também o tempo real de espera pelos passageiros. Outra das conclusões retiradas assenta na ideia de que aqueles passageiros com acesso às informações em tempo real esperam quase dois minutos a menos do que aqueles que não têm informação alguma.

Assim, deste estudo retira-se que as informações em tempo real têm a capacidade de melhorar a experiência de utilização de transportes públicos pela população, uma vez que têm toda esta informação ao seu alcance rapidamente e antes mesmo de chegarem às paragens, até seguirem os seus destinos.

2.1.2 How Long To Wait?

Também Pengfei Zhou, Yuanqing Zheng e Mo Li desenvolveram uma solução de previsão de tempo do próximo autocarro em determinada paragem, com base em *participatory sensing* [4], isto é, com a participação voluntária de pessoas. O conceito de *participatory sensing* tem estado em constante evolução nos últimos anos muito devido ao constante crescimento das tecnologias móveis também. Este conceito é definido num estudo efetuado por Emiliano De Cristofaro e Claudio Soriente [5] como sendo um paradigma computacional em ascensão que permite a distribuição de dados colhidos por pessoas ou comunidades. Estas pessoas, com recurso aos seus dispositivos móveis, nomeadamente *smartphones*, partilham informação através dos sensores equipados nestes, sendo dado como exemplo a monitorização e partilha da temperatura atual em determinado local. Além deste estudo, existem também outros que merecem destaque pelo que representam nesta área. Um desses estudos é o de Kate Shilton [6] onde esta define este conceito como sendo a utilização de dispositivos móveis para efetuar a cap-

tura de dados por uma pessoa ou comunidade abordando temas como a segurança e a transparência de dados do ponto de vista dos utilizadores. No entanto, são já algumas as aplicações nas lojas *online* que fazem uso deste paradigma. Desse conjunto de aplicações, uma das que mais se destaca é conhecida por *PulsodelaCiudad*¹ e encontra-se disponível tanto para as plataformas Android como iOS. Esta aplicação permite aos seus utilizadores enviarem informações acerca do estado do tempo, coordenadas GPS, dados ambientais como ruído ou temperatura para uma plataforma dedicada. Nesta, os utilizadores podem obter informações acerca dos estados ambientais noutras partes da cidade, sendo estas obtidas a partir de "relatos" de outros utilizadores que, também voluntariamente, forneceram esses dados. Baseando-se neste conceito, o sistema de Pengfei Zhou, Yuanqing Zheng e Mo Li foi desenvolvido e testado durante um período de sete semanas na rede de autocarros de Singapura. Este sistema assenta na utilização dos dispositivos móveis dos passageiros dos autocarros e tem como objetivo detetar o tempo que falta para a chegada do autocarro ou comboio a uma paragem existente na rota sendo esta requisitada por outro passageiro. Isto é conseguido uma vez que o dispositivo do passageiro que se encontra no autocarro ou comboio envia (com permissão) os dados de localização para um servidor que os processa, com base na rede móvel do dispositivo. Ao contrário de muitos outros estudos, que se baseiam nas localizações por GPS, este usa a rede móvel dos dispositivos como forma de não diminuir tão rapidamente a duração de bateria dos mesmos. Para além da rede móvel dos dispositivos, fazem-se uso de alguns sensores também, entre eles, o acelerómetro e o microfone. Este microfone é usado de modo a garantir que determinado utilizador está no interior do autocarro, uma vez que este sistema deteta o som - *beep* - que é despoletado aquando da passagem do cartão de viajante por parte dos passageiros na entrada do autocarro. Ao ser detetado este *beep*, o sistema sabe que o passageiro se encontra dentro de um autocarro e começa por detetar as torres de emissão de rede mais próximas à qual o dispositivo está ligado, de modo a obter-lhe a localização exata. Por outro lado, como forma de garantir que é o som de entrada é de um autocarro e não de um comboio, é utilizado o acelerómetro para garantir que as velocidades "a que o dispositivo se desloca" são de um autocarro e não de um comboio.

Após sete semanas de testes, a forma de funcionamento e de garantia da previsão de tempo de chegada do autocarro detetou erros máximos de oitenta segundos e este tempo é dado pela sequência de sinais enviada pelo dispositivo para o servidor. Caso o dispositivo envie parcialmente o sinal da sua localização mas de uma forma sequencial,

¹<http://www.smartsantander.eu/>

o servidor, que mantém a sequência completa, identifica a rota do autocarro e com base em algoritmos e cálculos específicos, envolvendo os segundos de envio de sinal, dá ao utilizador requisitante, o tempo que falta para o próximo autocarro.

2.1.3 *Lisbon Bus System: Retrieving Real-Time Information*

Ainda neste âmbito, David Alves, Luis M. Martinez e José M. Viegas elaboraram um estudo [7] acerca dos transportes públicos da cidade de Lisboa. Neste estudo é apresentado um *trip-planner* em tempo real, isto é, um sistema que auxilia o passageiro a calcular e prever a sua viagem. Este sistema tem a capacidade de informar os passageiros de quais são as melhores rotas para as suas viagens, os tempos de viagem esperados, com base nas localizações reais dos veículos e ainda as velocidades destas mesmas viagens com base nas informações de trânsito da última hora. Para a implementação deste sistema e de modo a obter dados coerentes, os autores deste basearam-se no sistema de *logs* do serviço de transporte público Carris² que continha informação sobre o tempo de viagens e velocidades das mesmas. A implementação do mesmo visou simular o funcionamento de uma rede de transportes públicos e criar um modelo para fazer os tempos de viagem mais curtos que o previsto.

A forma de funcionamento deste sistema (Figura 2.1) assenta no uso de um *Data Center* que filtra e processa a informação recebida por parte dos autocarros em circulação sendo esta utilizada para atualizar o histórico de informação de viagens bem como melhorar o tempo de espera estimado com base no algoritmo de previsão, desenvolvido com base na adaptação do algoritmo de Dijkstra³. Depois dessa informação é então feita a previsão do tempo de espera.

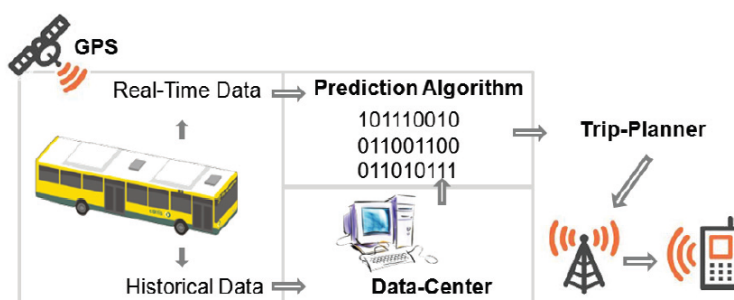


Figura 2.1: Funcionamento do sistema LisbonBus. [adaptado de <http://bit.ly/LisbonBusSystem>]

²<http://www.carris.pt/>

³<http://www.cs.auckland.ac.nz/software/AlgAnim/dijkstra.html>

2.1.4 Transit Trip Planner With Real Time Data

O estudo de Jerald Jariyasunant, Daniel B. Work, Branko Kerkez, Raja Sengupta, Steven Glaser e Alexandre Bayen datado de 2009 e denominado por "*Mobile Transit Trip Planning with Real Time Data*" foi efetuado com base na implementação de um *transit trip planner* para dispositivos móveis. A implementação do protótipo deste sistema, como é possível ver na Figura 2.2, prevê o caminho mais curto entre dois pontos usando para isso informações fornecidas pelo GPS presente nos autocarros e calculado através de um sistema dedicado. Para fazer uso desse mesmo sistema, o utilizador faz uso de uma aplicação disponível para iPhone tendo como base um mapa ou através do *browser* utilizando *Javascript*. Depois disso, essa informação é enviada a um servidor que faz os cálculos necessários utilizando um *Dynamic K-Shortest Path* que calcula o caminho mais curto e devolve informações para o utilizador com o tempo de demora da viagem e informações de rota, disponibilizadas através de um mapa.

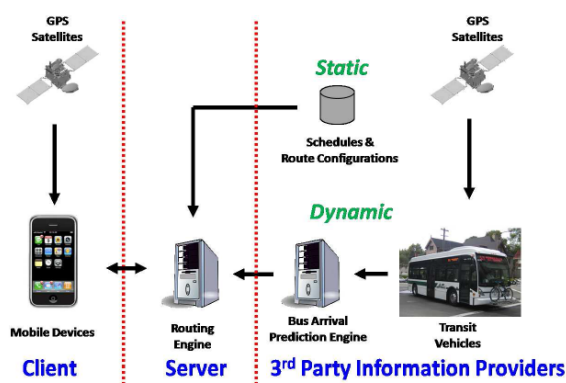


Figura 2.2: Arquitetura e funcionamento do sistema. [adaptado de <http://escholarship.org/uc/item/51t364vz>]

Na maioria das aplicações e estudos analisados, de forma a fornecer informação às aplicações cliente, independentemente da plataforma, existem as chamadas API's, que são responsáveis por fazer a gestão e devolução da mesma, como será introduzido na próxima secção.

No entanto, e para além dos estudos efetuados, têm sido feitos esforços por algumas empresas de transporte a fim de criar soluções móveis orientadas para os seus passageiros. Essas aplicações móveis encontram-se enumeradas de seguida.

2.1.5 *OneBusAway*

A solução móvel *OneBusAway*, de todas as analisadas, é a que mais funcionalidades disponibiliza ao utilizador e define-se por ser um conjunto API-Aplicação, acrescentando a isso o facto de ser gratuita. A informação desta aplicação é relativa às cidades de *Atlanta*, *Puget Sound*, *Tampa*, *Detroit* e Nova Iorque, nos Estados Unidos da América (o que é uma desvantagem) e os métodos suportados por esta são:

- Lista de companhias de transporte público e zonas de prestação de serviço;
- Detalhes de companhias de transporte;
- Tempo de chegada do próximo autocarro a determinada paragem;
- Tempos fixos de partida e chegada a determinada paragem;
- *Trip-Planner*;
- Criar alarmes de chegada/saída de autocarro de determinada paragem;
- Reportar problema com paragens/rota;
- Lista de rotas por companhia de transporte;
- Detalhes de cada rota;
- Procura de rotas por localização/nome;
- Acesso aos horários de determinada paragem para determinado dia;
- Lista de todas as paragens de cada companhia;
- Detalhes de cada paragem;
- Procura de paragens por nome, localização ou rota;
- Detalhes de determinada viagem;
- Lista de veículos de cada companhia de transporte.

As aplicações cliente *OneBusAway*, disponíveis tanto para *Android*, *Windows Phone* e *iOS* (Anexo A), todas aparentam estar bem conseguidas e desenhadas tendo em conta o que a sua própria API disponibiliza. Para *iOS* apresenta-se de uma forma "limpa" e de fácil utilização, permitindo a navegação no mapa e visualização das paragens de todos os autocarros (Figura 2.3 (a)). Para além disto, dá ainda ao utilizador a possibilidade de ver o tempo para o próximo autocarro com base na seleção da paragem no mapa (Figura 2.3 (b)), a marcação de paragens como favoritos e ainda a visualização de informações detalhadas de cada paragem (Figura 2.3 (c)). O grande ponto negativo desta aplicação é estar orientada "apenas" para as cidades já referidas e não permitir, para já, a adição de novas.

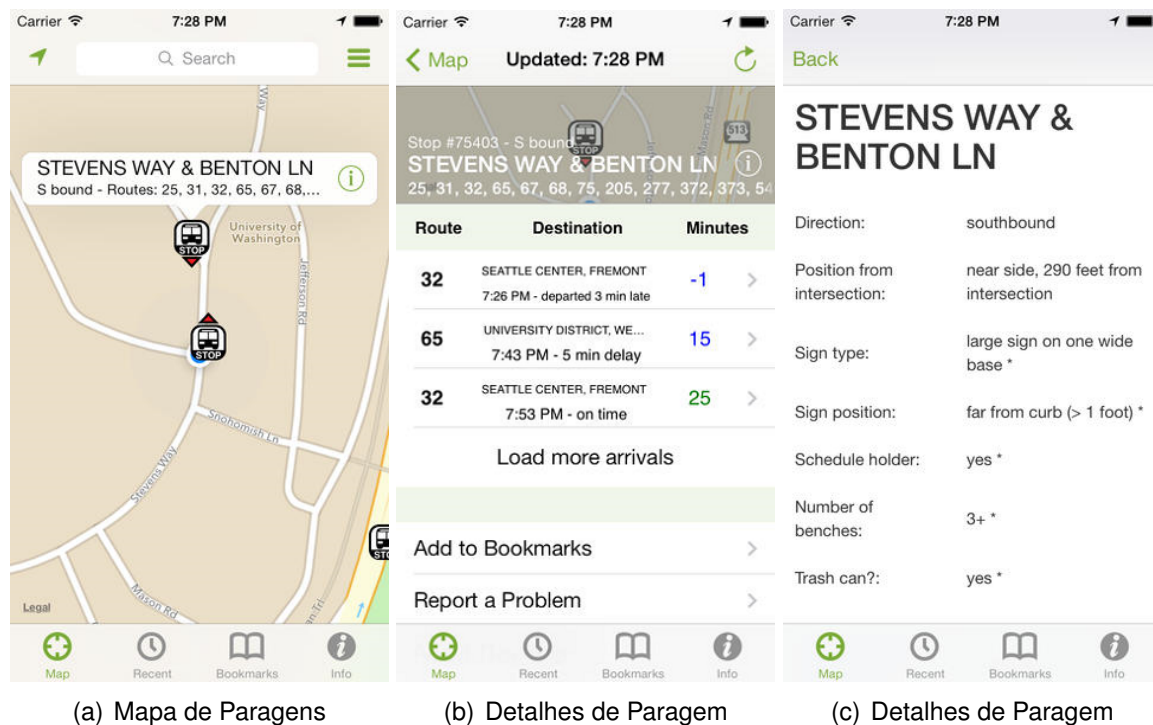


Figura 2.3: Interface da Aplicação One Bus Away

2.1.6 Bongo - Bus on The Go

Na mesma linha de funcionamento da *OneBusAway*, está a aplicação Bongo. Esta foi implementada pela Universidade de Iowa para as cidades de *Iowa* e *Coralville*. A informação agregada e devolvida à aplicação cliente, *Bongo - Bus on The Go*⁴, reúne a lista de rotas e detalhes destas, a lista de paragens e detalhes de localização das mesmas, bem como os tempos de espera previstos para os próximos autocarros. Garante ainda acesso à localização dos autocarros em tempo real. Em termos de interface, as duas versões (iOS e Android) estão bastante próximas e bem conseguidas. O destaque meteorológico na área de utilização da aplicação é algo incomum nas aplicações do género mas não está de todo desenquadrado com o âmbito aqui abordado, isto é, a utilização de aplicações móveis de forma a auxiliar os passageiros nas suas deslocações. Para além disso, a disponibilização dos tempos de espera nas paragens mais próximas é outro dos pontos positivos desta aplicação, tal como a apresentação de todas as rotas das duas cidades. Contudo, esta funcionalidade das rotas carece de um sistema de pesquisa. É ainda possível a listagem de todas as paragens no mapa bem como aceder

⁴disponível em <http://ebongo.org/>

aos detalhes das mesmas. Como ponto negativo a destacar é apenas o facto de serem poucas as funcionalidades disponibilizadas pela API e, conseqüentemente, poucas as existentes na aplicação cliente.

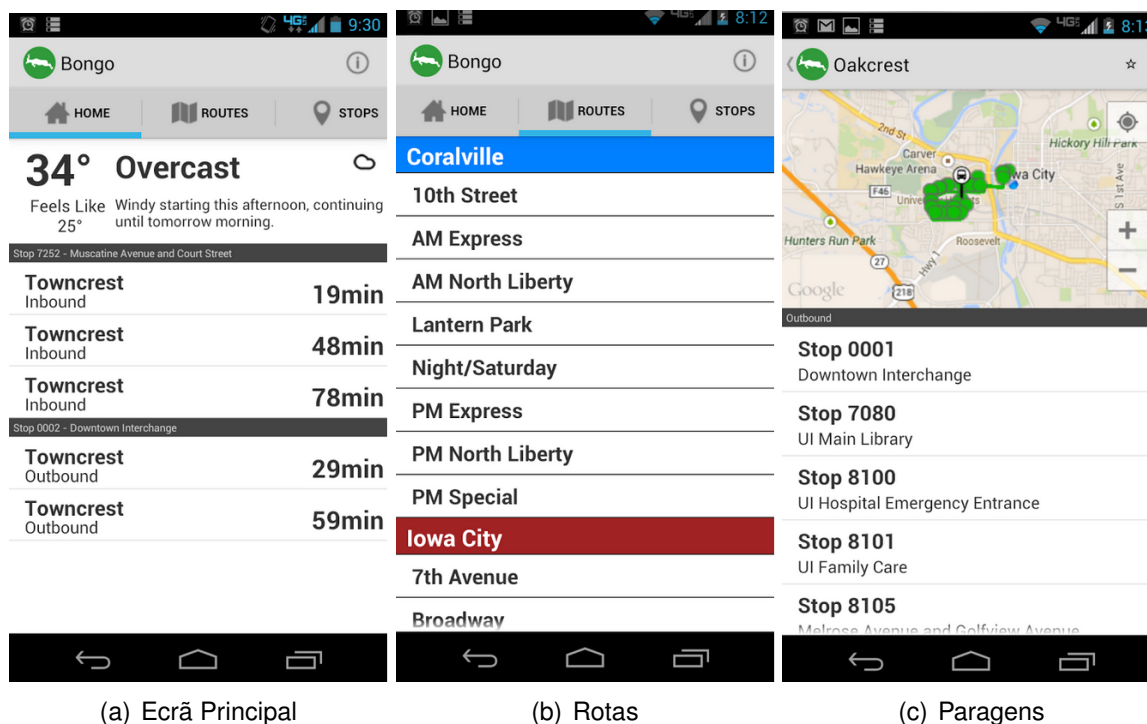


Figura 2.4: Interface da Aplicação Bongo

Aquando da análise destas aplicações (ver Anexo A) notou-se uma grande presença de aplicações concebidas para o Reino Unido. Assim serão apresentadas nesta secção a *UK Bus Checker*⁵ e *Live London Bus Tracker*⁶, ambas para o sistema operativo móvel Android⁷.

2.1.7 UK Bus Checker

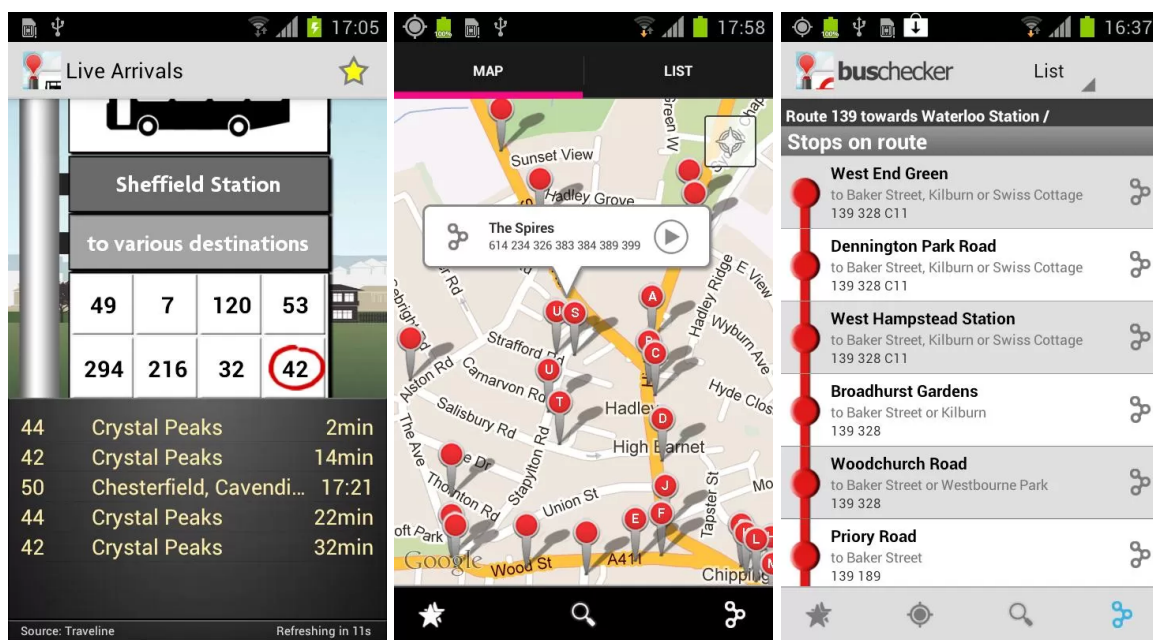
A aplicação *UK Bus Checker*, está disponível tanto nas versões *Lite* como *Pro* e o seu principal objetivo é permitir a monitorização dos percursos e tempos de chegada dos autocarros às diversas paragens do Reino Unido, em tempo real, indo de encontro ao

⁵de FatAttitude, disponível em <http://www.buschecker.com/?app=UK>

⁶de AppEffectsUK, disponível em <https://play.google.com/store/apps/details?id=com.appeffectsuk.bustracker>

⁷<http://www.android.com/>

estudo *Where is My Bus?* realizado na área de Seattle e já apresentado. Esta aplicação permite o acesso a todos os autocarros em determinada paragem, bem como o tempo que resta esperar para os mesmos. Para além disso, dá ainda a possibilidade de visualização da rota de cada autocarro no mapa permitindo ainda listar as várias paragens por rota e fazer uso de um sistema de favoritos para guardar essas mesmas paragens.



(a) Tempo de espera numa paragem

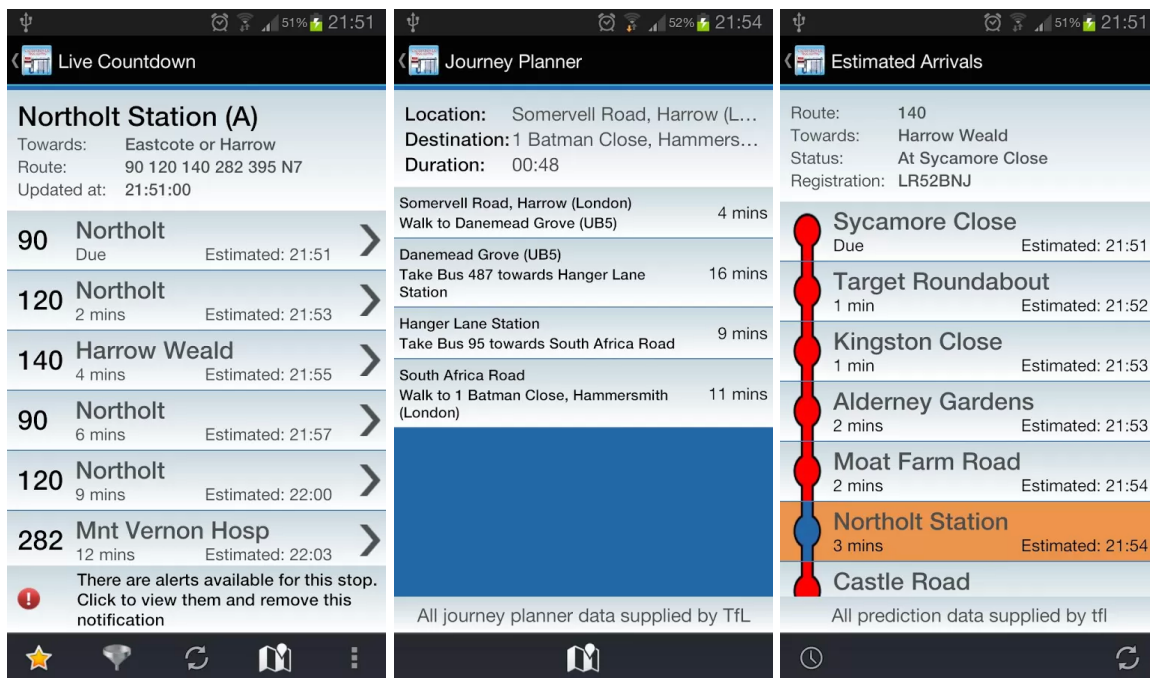
(b) Mapa de paragens

(c) Detalhes de uma rota

Figura 2.5: Interface da Aplicação UK Bus Checker

2.1.8 Live London Bus Tracker

Dentro do mesmo segmento de aplicações, a *Live London Bus Tracker* garante ao seu utilizador o tempo que falta para o próximo autocarro em determinada paragem. Da análise efetuada, destacam-se a diversidade de formas de planear uma rota, o acesso ao tempo que falta para o próximo autocarro em determinada paragem após a seleção desta no mapa e ainda a forma como as funcionalidades se apresentam no ecrã.



(a) Tempos de chegada

(b) Cálculo de um percurso

(c) Detalhes de uma rota

Figura 2.6: Interface da Aplicação Live London Bus Tracker

Para o estudo efetuado foram tido em conta não só pesquisas científicas portuguesas (como é o caso da *Lisbon Bus System: Retrieving Real-Time Information*), mas também aplicações móveis desenhadas para Portugal. Assim, tanto para os sistemas móveis Android e iOS encontram-se as aplicações Lisboa MOVE-ME⁸, IZI Carris⁹ e Sapo Transportes¹⁰, de seguida introduzidas.

2.1.9 MOVE-ME

Sendo a mais recente das três, para ambas as plataformas descritas, destaca-se por funcionalidades como a procura de paragens por raio ou ainda a disponibilização de pontos de interesse. Num cômputo geral, o número de funcionalidades é escasso, o que é um ponto negativo a salientar, tal como a sua interface, que neste caso, saindo do espectro de aplicações deste género, não é, de todo, a mais indicada, ressaltando-se aqui a utilização de botões de grandes dimensões, que auxilia quando se faz uso da

⁸de OPT, disponível em <https://play.google.com/store/apps/details?id=com.moveme>

⁹de IZIMOOVE, disponível em <https://play.google.com/store/apps/details?id=pt.izimooove.carris>

¹⁰de MEO - Serviços de Comunicações e Multimédia, S.A, disponível em <https://play.google.com/store/apps/details?id=pt.sapo.android.transportes>

aplicação em movimento.

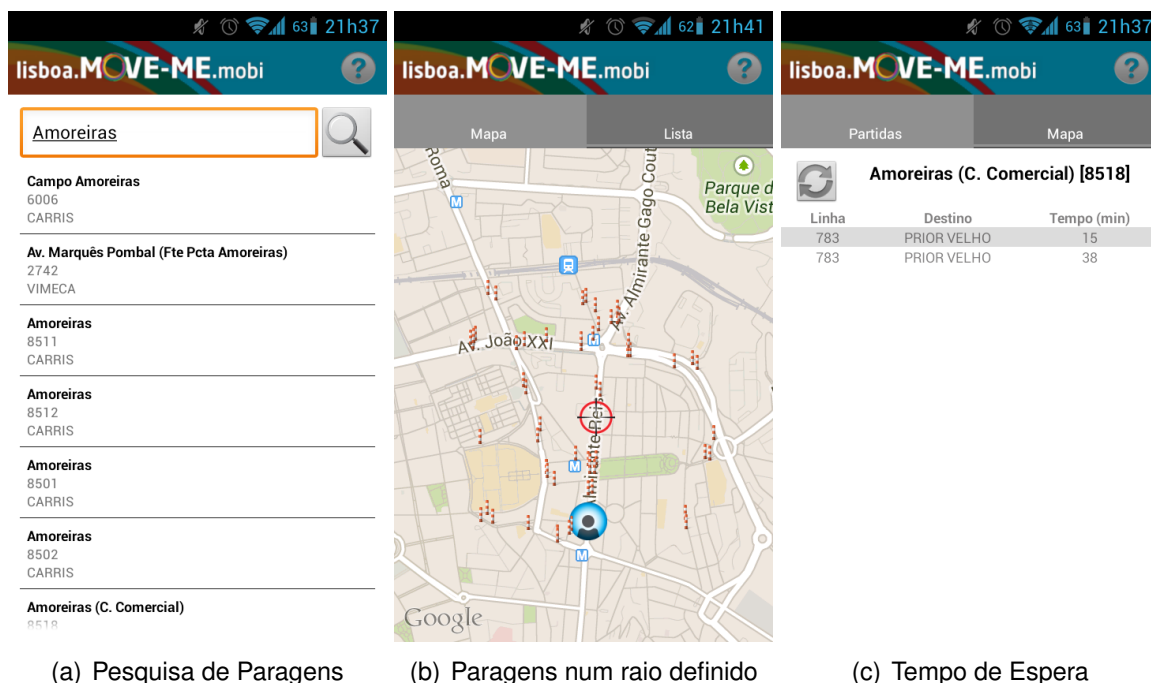
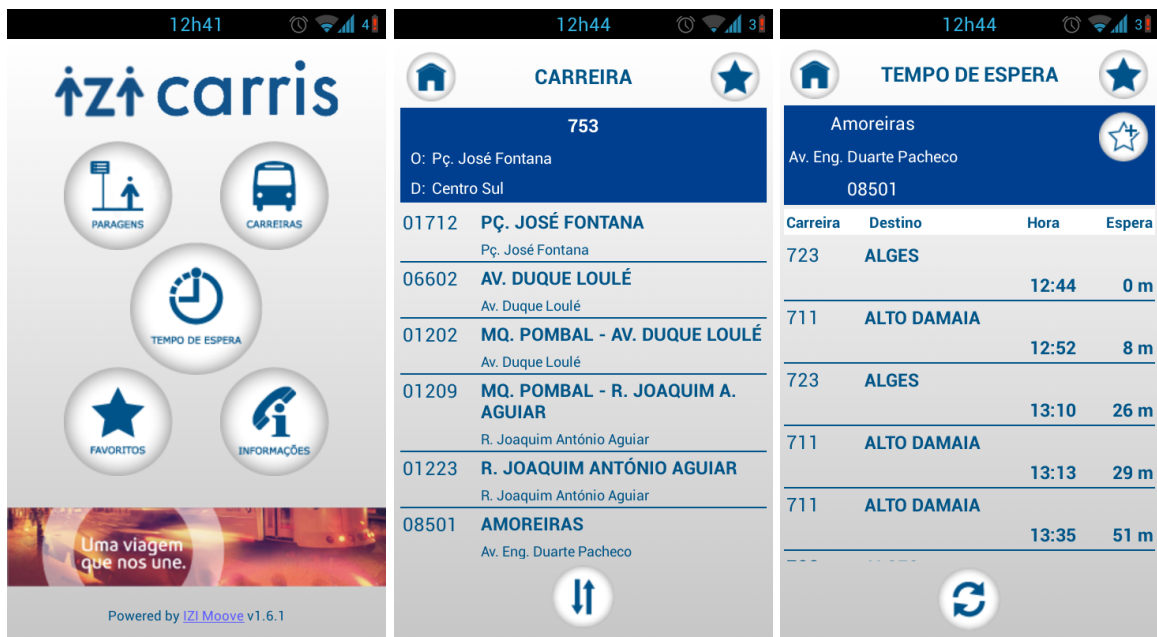


Figura 2.7: Interface da aplicação MOVE-ME

2.1.10 IZI Carris

A aplicação IZI Carris foi desenvolvida exclusivamente para a utilização de dados desta rede de transportes públicos, a mesma utilizada no estudo efetuado por David Alves, Luis M. Martinez e José M. Viegas, anteriormente abordado. Esta aplicação, orientada para os utilizadores desta companhia de transportes, permite a que estes tenham acesso à informação em qualquer lugar. Para isso, a IZI Carris permite, entre outras funcionalidades, o acesso ao tempo que falta para o próximo autocarro, em determinada paragem, a visualização das rotas destes e ainda a seleção e gestão destas paragens num sistema de favoritos. Ainda assim, apesar de conter um número razoável de funcionalidades, tendo em conta o raio de utilização, esta aplicação perde por depender de redes móveis ativas, não disponibilizando informação *offline*. Para além deste ponto negativo, a disponibilização dos elementos no ecrã não é a melhor, perdendo-se por vezes a noção de se está.



(a) Menu principal

(b) Lista de rotas

(c) Cálculo de uma rota

Figura 2.8: Interface da Aplicação IZI Carris

2.1.11 Sapo Transportes

A Sapo Transportes é uma aplicação, das analisadas, um pouco diferente das anteriores, uma vez que inclui mais do que um serviço de transportes, nomeadamente, autocarros, barcos, comboios e metro todos na área metropolitana de Lisboa, o que a torna numa mais valia. Para além desta, o sistema de favoritos, a possibilidade de seleção de várias companhias de transporte ou o acesso aos horários dos autocarros nas diversas paragens, são também funcionalidades que contribuem para o sucesso desta aplicação. Contudo, a não existência de um modo *offline* e, conseqüentemente, a dependência de rede para aceder à informação, a interface confusa ou o *delay* constante da aplicação, fazem-na descer uns pontos em termos de satisfação, por parte dos utilizadores diários da mesma.



Figura 2.9: Interface da Aplicação Sapo Transportes

2.1.12 Análise Geral

Para além das aplicações presentes neste capítulo, foram também analisadas um conjunto de outras que se encontram presentes no Anexo A. Para que tal fosse possível, foi selecionado um vasto conjunto de aplicações presentes nas três maiores lojas *online*, a *Google Play*¹¹, a *Windows Phone Store*¹² e a *Apple Store*¹³.

Finalizada a análise das aplicações (tanto desta secção como das presentes em anexo) chegou-se à conclusão que a maioria destas partilham um enorme leque número de funcionalidades, isto é, dado aquilo a que se propõem, conseguem ter, não só as funcionalidades em si, como também modos de funcionamento semelhantes. Isto é refletido nas três plataformas abordadas, uma vez que, apesar de serem todas diferentes assentam sobre os mesmos princípios de disponibilização de dados e interação com o utilizador, utilizando as mesmas *guidelines* e boas práticas no desenvolvimento de aplicações deste tipo.

De forma a complementar a análise efetuada, apresenta-se uma tabela resumo com as

¹¹<https://play.google.com/store>

¹²<http://www.windowsphone.com/pt-pt/store>

¹³<http://store.apple.com/pt>

funcionalidades encontradas nas aplicações analisadas nesta secção.

| Funcionalidade | Aplicação | | | | | | |
|----------------------------------|-------------------------|----------------|------------|------------------|---------|-------|--------------|
| | Live London Bus Tracker | UK Bus Checker | IZI Carris | Sapo Transportes | MOVE-ME | Bongo | One Bus Away |
| Horários e Preçário | | | | ✓ | | | |
| Vários Serviços Integrados | | | | ✓ | | | |
| Acesso Tempos de Espera | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Pesquisa de Rota | | | ✓ | ✓ | | | ✓ |
| Pesquisa de Paragens | ✓ | ✓ | ✓ | | ✓ | | ✓ |
| Visualização Paragens por Rota | | ✓ | | | ✓ | ✓ | |
| Visualização de Rota no Mapa | ✓ | ✓ | | | ✓ | ✓ | ✓ |
| Visualização de Paragens no Mapa | | ✓ | | | ✓ | ✓ | ✓ |
| Sistema de Favoritos | | ✓ | ✓ | | ✓ | ✓ | ✓ |
| Calculadora de Percurso | ✓ | ✓ | | ✓ | ✓ | | |
| Acesso a Paragens Mais Próximas | | | | | ✓ | ✓ | |

Tabela 2.1: Funcionalidades das Aplicações Analisadas

Como é possível observar na Tabela 2.1 há um número comum de funcionalidades partilhado pelas diversas aplicações. Das presentes, é importante destacar a importância de funcionalidades como o sistema de favoritos, a pesquisa de paragens e o acesso ao tempo de espera. O resultado da disponibilização do tempo de espera indica que esta funcionalidade é um dos pontos chave deste tipo de aplicações, tal como foi visto na pesquisa científica realizada por Kari Edison Watkins, Brian Ferris, Alan Borning, G. Scott Rutherford e David Layton.

Além da funcionalidade do tempo de espera, há ainda outras que merecem destaque tais como o sistema de favoritos ou até mesmo a pesquisa de paragens. Em primeiro lugar, como consequência do número de vezes que aparece nas diversas aplicações, o

sistema de favoritos é uma funcionalidade que merece ser destacada dada a sua importância numa aplicação deste género pois auxilia o utilizador nas tarefas repetitivas, isto é, permite-lhe aceder às paragens e percursos (por norma associados aos favoritos) sem que este tenha que efetuar vários passos e ultrapassar vários menus. Também a pesquisa de paragens é uma das funcionalidades importantes pois permite ao utilizador obter informação sob determinada paragem sendo que para isso, apenas tenha de introduzir a sua localização ou rota de autocarro associada.

2.2 Infra-Estruturas de Partilha de Informação

Para além das infra-estruturas de obtenção de informação existem ainda as de partilha que reúnem mecanismos necessários para a divulgação de informação, também conhecidas por API's. Estas API's, ou neste caso, *API's REST* ou *RESTful API's*, não são mais do que métodos que permitem a comunicação entre um sistema cliente e um sistema servidor. Com este mecanismo de gestão e disponibilização de informação cada vez mais em crescimento [8] [9] são várias as empresas a implementarem soluções deste género isto é, a implementarem API's que disponibilizem informação às aplicações cliente, suportando as várias plataformas. Neste âmbito, existem dois tipos de API's isto é, umas de carácter genérico, outras de carácter particular. As genéricas são normalmente comuns a vários tipos de aplicações sendo que estas têm que se adaptar a quem fornece a informação. Por outro lado, há as particulares que são desenhadas apenas e só para que o cliente faça uso da informação por ela enviada, não sendo necessário aqui o ajuste do cliente à informação devolvida pelo servidor. De seguida podem ver-se em detalhe alguns exemplos de API's fornecedoras de informação no âmbito dos transportes públicos.

2.2.1 *OpenTripPlanner*

*OpenTripPlanner*¹⁴ ou em português "planeador de percursos" é um *software open source* construído para efetuar cálculos de percursos e tempos com base numa arquitetura cliente-servidor. Além disto, são proporcionadas ainda interfaces *web* e uma API REST

¹⁴<http://www.opentripplanner.com/>

para as aplicações poderem fazer uso deste sistema. De modo a que este seja complementado, o OTP, baseia-se em dados *open source* tais como o GTFS (*General Transit Feed Specification*) para acompanhar o estado destes mesmos transportes aquando dos cálculos, como ainda o OpenStreetMap¹⁵ para disponibilizar a interface.

Baseado neste, o Transportes Públicos.pt é um sitio na Internet que disponibiliza informação acerca dos transportes públicos em Portugal. Essa informação é agrupada numa plataforma que a agrega e a disponibiliza a quem pretender fazer uso dela. Tal como o seu nome indica, este serviço engloba tudo a que isto diz respeito, isto é, transportes rodoviários, ferroviários e fluviais, isto é, autocarros urbanos, carreiras inter-urbanas, camionetas inter-regionais, comboios, metros, elétricos, barcos, entre outros. Assim, o Transportes Públicos.pt destaca-se pela vasta área de conteúdo que reúne, ou seja, o seu foco de incidência não são apenas os grandes polos urbanos, tais como Porto ou Lisboa, mas também outras cidades de Portugal, que vão sendo adicionadas consoante a disponibilização de dados por parte dos operadores de transportes públicos. Além da sua versão *online* (Figura 2.10), este sistema faz uso da aplicação *OpenTripPlanner*¹⁶ para o sistema operativo móvel Android. Esta aplicação é usada por diversos serviços de vários países e é responsável por, com base na informação do Transportes Públicos.pt, disponibilizar ao utilizador informação do percurso, tempo que este demorará a percorrer, número de transbordos informando ainda o utilizador das direções detalhadas caso este tenha partes do seu percurso que são realizadas a pé.

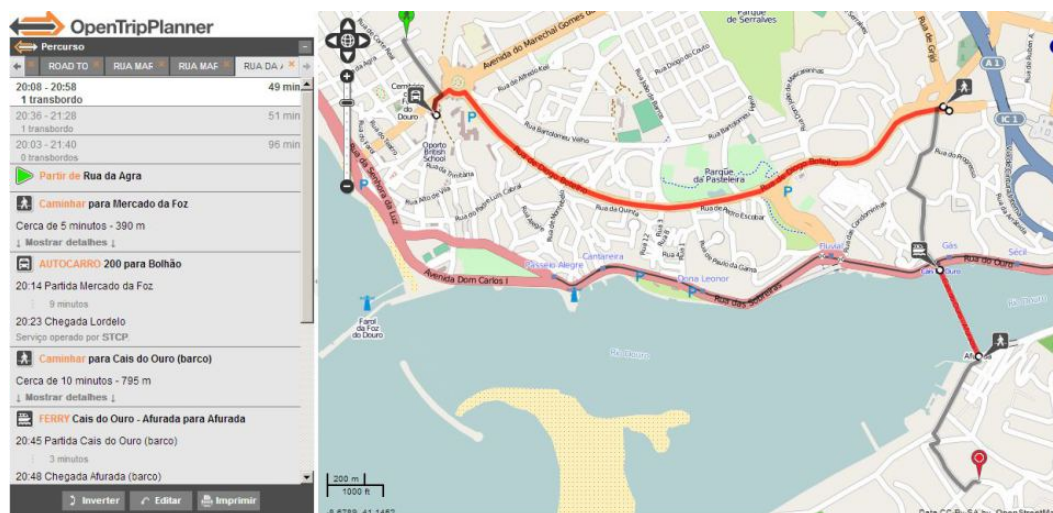


Figura 2.10: Interface do Serviço Transportes Públicos.pt com base no OpenTripPlanner [adaptado de <http://bit.ly/aventadores>]

¹⁵<http://www.openstreetmap.org/>

¹⁶de CUTR @ USF, disponível em <http://bit.ly/OpenTripPlanner>

2.2.2 OneStopTransport

Esta API portuguesa¹⁷ tem como objetivo disponibilizar informação para ser consumida por uma aplicação cliente. O objetivo dos criadores do serviço é promover e dinamizar o movimento "de dados abertos" na área da mobilidade de forma a que qualquer programador possa aceder e utilizar esses mesmos dados para o desenvolvimento de aplicações.

Esta API, em concreto, permite à aplicação cliente o uso de funcionalidades como o cálculo de rotas entre dois pontos e o melhor meio de transporte para o fazer, o acesso a pontos de interesse e eventos da cidade do Porto bem como dados geográficos de outros pontos do país. Permite ainda aceder a dados, ainda que estáticos, sobre os transportes urbanos de Coimbra e Aveiro, bem como, dos comboios de Portugal. Para além disto, é possível ainda aos serviços que disponibilizam o transporte, às rotas e percursos possíveis, paragens e tempos de espera, alertas, eventos, dados de trânsito, entre outros.

Contudo, apesar do número elevado de funcionalidades disponibilizadas e vantagens associadas às mesmas, esta API deixou de ter suporte em Janeiro de 2014, pelo que, num serviço deste género, é um grande inconveniente.

2.2.3 Transport API

Desenvolvida por um vasto grupo de programadores, esta API britânica¹⁸, promove a disponibilização de dados sobre transportes públicos de modo a serem consumidos por diversos tipos de clientes, desde aplicações móveis a portais de transportes.

Os dados disponibilizados por esta API, devolvidos através do formato JSON, baseiam-se em informações dos transportes públicos britânicos, nomeadamente, a rede de caminhos de ferro e metro, abrangendo ainda a rede de autocarros. Estes mesmos dados, para além das paragens mais próximas e mais próximas por pesquisa de área - dos diversos tipos de transporte - apresenta ainda as partidas (em tempo real) dos autocarros e os seus horários.

Contudo, e por ser uma API "profissional" tem a desvantagem do limite de pedidos de informação. Este limite está fixado nos mil pedidos diários sendo que, caso sejam necessários mais, levará a custos por pedido, ainda que muito reduzidos.

¹⁷<https://www.ost.pt/>

¹⁸<http://transportapi.com/>

2.2.4 *Chicago Transit Authority API*

Podendo ser utilizada tanto em versão *mobile* ou versão *web*, esta API¹⁹ fornece a informação necessária aos passageiros de transportes públicos, dando-lhes a noção real do tempo para o próximo transporte público, em determinado local, dentro da área de Chicago, nos Estados Unidos. Para isso, tal como as analisadas anteriormente neste capítulo, esta faz uso do sistema de GPS de forma a determinar a posição do autocarro e com isso mesmo, calcular e disponibilizar essa informação aos consumidores finais, neste caso os passageiros. Tal como a *Transport API*, esta tem um limite de pedidos de informação diários, neste caso de dez mil, sendo que, a partir deste valor seja tido em consideração por parte dos administradores do serviço se podem ou não estender este limite, de forma a gerir da melhor forma a largura de banda consumida.

2.2.5 *OpenStreetMap*

Embora não sendo uma infra-estrutura de partilha de informação de posições de autocarro ou tempos que faltam para o próximo em determinada paragem, é uma API interessante de apresentar para o âmbito deste trabalho. Como tal, o *OpenStreetMap*²⁰ ou em português "Mapa Aberto de Ruas" é um sistema de mapas criado por Steve Coast²¹ em 2004 que inicialmente foi pensado para o Reino Unido. Contando com o apoio inicial da gigante americana Yahoo!²² que forneceu dados de imagens aéreas, Steve foi encorajado a continuar o seu projeto. Três anos mais tarde, juntamente com a Yahoo! e já fazendo parte da *OpenStreetMap Foundation*²³, surgiram também "ofertas" de informação geográfica da Holanda, Índia, China e Estados Unidos da América.

Hoje em dia, são utilizadores de dados deste serviço empresas como a Apple - que terminou contrato de mapas com a Google e passou a adotar os mapas TomTom com *OpenStreetMap*, o *Foursquare*²⁴ e ainda o famoso sistema de anúncios, *Craigslist*²⁵. No entanto é importante ter em consideração que este sistema é desenvolvido por uma comunidade voluntária que contribui para manter a informação atualizada sobre os mais variados locais por todo o mundo. Esta informação é tudo aquilo que sejam caminhos

¹⁹<http://www.transitchicago.com/developers/bustracker.aspx>

²⁰<http://www.openstreetmap.org/>

²¹http://en.wikipedia.org/wiki/Steve_Coast

²²<https://www.yahoo.com/>

²³http://osmfoundation.org/wiki/Main_Page

²⁴<https://pt.foursquare.com/>

²⁵www.craigslist.org/

de ferro, paragens de autocarro, estradas, ruas, entre outros e permite ser editada e/ou complementada por membros de toda a comunidade sendo que para isso apenas seja necessário um registo. Como consequência, os dados presentes são introduzidos por membros e de forma alguma são validados por um sistema, o que significa que terá de se garantir confiança nos mesmos, assegurando que são válidos, os mais corretos e se encontram atualizados. Em jeito de estatística, é importante ainda referir que em 2011 foi ultrapassada a barreira do milhão e meio de utilizadores registados [10] que contribuem para o sucesso deste sistema.

A sua presença neste estudo surge tendo em conta a informação contida neste mapa (paragens, ruas, lojas, entre outros) em que é possível, com recurso a API's próprias ou de terceiros, fazer uso desta para aplicações de transportes públicos.

2.2.6 GTFS

A criação do formato GTFS surgiu em 2005, pela mão de Bibiana McHugh²⁶ então gestora de projetos na empresa de transportes de Portland, nos Estados Unidos, de nome TriMet²⁷. Após uma viagem para fora do seu país em que se viu confrontada sem informações de trânsito, ou transportes públicos no seu destino, decidiu questionar a Google e a Yahoo! de forma a saber se estas iriam incorporar o serviço de trânsito nos seus mapas.

Uma vez que apenas a Google lhe deu resposta, e após um trabalho em conjunto da sua empresa, a Trimet com a gigante americana, rapidamente se conseguiu aproximar as especificações necessárias pela Google com aquelas que a Trimet tinha, do ponto de vista de empresa de transporte. Meses após a ideia, surgiu um piloto que correu durante um ano, apenas com dados da empresa Trimet, sendo este denominado de *Google Transit Trip Planner* então conhecido por GTFS [11]. Um ano mais tarde, em Setembro de 2006, foram adicionadas mais cinco cidades. Apesar do receio por parte da Google da não adoção das empresas ao Google Transit Trip Planner conjugado com o facto da não existência de infraestruturas necessárias para essa adoção, certo é que hoje em dia são mais de cem as empresas a utilizar o GTFS nos Estados Unidos, e cerca de quatrocentas pelo resto do mundo [12].

Neste ponto, em que GTFS significava *Google Transit Feed Specification* e dada a cres-

²⁶<http://trimet.org/difference/bibi.htm>

²⁷<http://trimet.org/>

cente utilização não só como integração no sistema de mapas, como também no formato de partilha de informação pelas empresas a Google decidiu então renomear o nome do formato para *General Transit Feed Specification*²⁸.

2.2.7 GTFS - RealTime

A especificação GTFS foi um trabalho em conjunto bem conseguido através de muito esforço e dedicação de duas entidades, a Google e a TriMet, como já referido neste documento. Ainda assim, este sistema "estático" por si só não resolvia o problema dos dados em tempo real para as empresas. Cientes deste problema surgiu, em Junho de 2011, através da Google em conjunto com programadores independentes e dedicados apenas a melhorar os sistemas de trânsito²⁹ uma nova especificação, o GTFS RT (Anexo C). Esta nova especificação veio acrescentar três novas funcionalidades essenciais para este âmbito:

- Atualizações de Viagem;
- Alertas de Serviços;
- Posições de Veículos.

As atualizações de viagem permitem divulgar possíveis atrasos, alterações e/ou cancelamento destas mesmas viagens. Para além disso, estas atualizações fornecem aos passageiros informação acerca dos horários de partida e chegada nas diversas paragens independentemente da rota da qual fazem parte. Neste contexto, uma atualização de viagem consiste na atualização dos horários de paragem dos veículos em cada uma destas e conseqüentemente a atualização de cada *StopTimeUpdate* (ver Anexo D). Cada *StopTimeUpdate* está associado a uma paragem, pelo que, será atualizado segundo a sequência definida no ficheiro *stop_times.txt*. Para além do tempo de chegada e partida exato de e para essa determinada paragem, respetivamente, deverá também ser fornecido o atraso existente em relação ao horário de chegada previsto.

No entanto, os Alertas de Serviços são também um ponto importante nesta referência. Do ponto de vista da empresa é possível fornecer mais detalhe ao passageiro, podendo ser-lhe indicado um URL com mais informação acerca de determinado alerta, um título para o mesmo e ainda uma descrição com detalhes do mesmo. Para além destes de-

²⁸https://groups.google.com/forum/#!topic/gtfs-changes/ob_7MIOvOxU

²⁹<http://www.bart.gov/schedules/developers/gtfs-realttime>

talhes, há ainda outro tipo de informação, embora não tão relevante, que faz parte de um alerta de viagem. Essa informação é constituída pelo intervalo de tempo em que o alerta estará ativo, a entidade afetada, que indica o que o alerta afetará, neste caso se toda a rede de transportes, uma rota, um viagem ou uma paragem. Para além destes, há ainda a relação causa-efeito. As causas possíveis para descrever os alertas são:

- Acidente;
- Atividade policial;
- Causa Desconhecida;
- Construção;
- Emergência Médica;
- Feriado;
- Greve;
- Manifestação;
- Manutenção;
- Outra;
- Problema Técnico;
- Tempo.

Os efeitos podem classificar-se da seguinte forma:

- Atrasos Significativos;
- Desconhecido;
- Desvio;
- Outro Efeito;
- Paragem Deslocada;
- Sem Serviço;
- Serviço Adicional;
- Serviço Modificado;
- Serviço Reduzido.

De forma a complementar as duas funcionalidades abordadas surgiu ainda a necessidade da propagação da posição de veículos através desta especificação. Em ambiente de testes, e à semelhança das atualizações de viagem, é um trabalho complexo pois

ambas as funcionalidades requerem um dispositivo a bordo³⁰. Como tal, cada veículo a circular deverá ter um dispositivo - por norma com base em localizações GPS - e divulgar a sua posição. Esta divulgação é conseguida através da conjugação de vários parâmetros como a latitude e longitude (ambos obrigatórios), a direção tomada pelo veículo, a distância total percorrida por este e ainda a velocidade, todos opcionais. A posição do veículo permite ainda à entidade transportadora saber, em cada momento, o nível de congestionamento. Este nível é agrupado nas seguintes categorias:

- Congestionamento;
- Congestionamento Grave;
- Estável;
- Desconhecido;
- Paragens Frequentes.

Para além da posição e do nível de congestionamento, existem ainda dois parâmetros, o estado de paragem do veículo que se caracteriza pela relação deste com as paragens, isto é, se está a chegar, se está a partir ou se está presente numa. O parâmetro final é o detalhe do veículo, que é composto por um identificador de cada veículo acompanhado do nome deste e ainda da matrícula do mesmo.

Embora o funcionamento seja mais complexo e, conseqüentemente, a sua utilização seja mais reduzida que a especificação GTFS, há já alguns casos em que o *Real Time* se encontra em funcionamento³¹. Neste caso, a troca de dados baseia-se em *Protocol Buffers*³²[13], formato de dados criado pela Google para tornar a troca de informação "menos pesada" para os diversos sistemas capazes de a encaminhar.

2.2.8 *Walk Score Public Transit API*

Contendo dados de mais de duzentas companhias de transportes de vários países de todo o mundo, a *Walk Score Public Transit API*³³ encarrega-se de agregar esses mesmo dados e disponibilizá-los às aplicações cliente, que consomem esta informação. A informação disponibilizada fica-se pelas paragens, detalhes destas e rotas daquela mesma

³⁰<https://developers.google.com/transit/gtfs-realtime/vehicle-positions>

³¹http://www.mbta.com/rider_tools/developers/default.asp?id=22393

³²<https://developers.google.com/protocol-buffers/?hl=pt-PT>

³³<http://www.walkscore.com/professional/public-transit-api.php>

cidade.

2.2.9 Transport - Swiss Public Transport API

Esta API Suíça³⁴ é disponibilizada a todos aqueles que pretendem implementar aplicações utilizando dados públicos reais daquele país. Assim, são garantidos às aplicações cliente vários tipos de dados. Esses mesmos dados são nomeadamente o que existe em redor da procura efetuada (paragens, pontos de interesse, ...), ligações a partir de um ponto actual ou um a partir ponto procurado e como chegar a outro bem como a lista de paragens existentes e os detalhes das mesmas.

2.2.10 Análise Geral

Para além das aplicações, há também um conjunto de mecanismos responsáveis por garantir que a informação é correta e que está disponível sempre que requisitada. Esse conjunto de mecanismos são conhecidos por API's REST. Das API's analisadas encontram-se dois tipos distintos sendo eles as particulares e genéricas. As primeiras são normalmente desenhadas para ir de encontro ao que o cliente pretende disponibilizar ao utilizador final, enquanto que as genéricas implementam um conjunto de normas previamente definidas às quais o sistema cliente terá de se adaptar de forma a que tudo funcione como é suposto.

Posto isto, em jeito de resumo, a Tabela 2.2 apresenta a comparação dos pontos comuns entre estas mesmas API's.

³⁴<http://transport.opendata.ch/>

| | Funcionalidade | | | | | | |
|----------------------------------|-----------------------------|-----------------------|----------------------|-------------------|--|-----------------------------|--|
| | Aplicação Cliente Associada | Formato de dados JSON | Formato de dados XML | Custos Associados | Diferentes Tipos de Serviços de Transporte | Solução para Várias Cidades | Informação de Rotas, Paragens e Horários |
| <i>OneBusAway</i> | ✓ | ✓ | ✓ | | | ✓ | ✓ |
| <i>Bongo</i> | ✓ | | ✓ | | | ✓ | ✓ |
| <i>OpenTripPlanner</i> | ✓ | ✓ | ✓ | | | ✓ | |
| <i>OneStopTransport</i> | | ✓ | | | | ✓ | ✓ |
| <i>Transport</i> | | ✓ | | ✓ | ✓ | | ✓ |
| <i>Chicago Transit Authority</i> | | | ✓ | ✓ | | ✓ | ✓ |
| <i>GTFS</i> | | | | | ✓ | ✓ | ✓ |
| <i>GTFS RealTime</i> | | | | | ✓ | ✓ | ✓ |
| <i>OpenStreetMap</i> | | | ✓ | | ✓ | ✓ | ✓ |
| <i>Walk Score Public Transit</i> | | ✓ | | | | | ✓ |
| <i>Swiss Public Transport</i> | | ✓ | | ✓ | | ✓ | ✓ |

Tabela 2.2: Funcionalidades das Infra-Estruturas de Partilha de Informação Analisadas

Das API's analisadas neste documento e acima enumeradas, apenas duas têm custos se excederem determinado número de pedidos de informação por dia ou por mês. Este facto demonstra uma grande adesão e disponibilização de informação através deste tipo de infra-estruturas. Contudo, é também possível verificar que tanto o formato XML como JSON são comuns na partilha de informação e que a maioria das infra-estruturas garantem aos seus clientes, em diversas cidades, o acesso a informação acerca de paragens, rotas e horários, entre outros.

2.3 Conclusão

Cada vez mais os sistemas de informação estão unidos com o objetivo de auxiliar a população, independentemente das áreas de operação. Posto isto e com o mesmo objetivo se encontra a solução a desenvolver e que ao longo deste documento será abordada, isto é, o de auxiliar a população nas suas deslocações diárias através do uso de transportes públicos garantindo-lhes mais comodidade e segurança tendo, ao mesmo tempo, o objetivo de aumentar os pontos a favor da utilização dos transportes públicos. Além disso e para que sejam objetivos exequíveis é necessário ainda garantir o auxílio às companhias de transporte, de forma a colocar a sua informação *online* e acessível a todos, uma vez que hoje em dia ainda não é muito comum. Como tal, através dos estudos apresentados, bem como, das aplicações analisadas pode concluir-se que a área dos transportes tem ganho alguma relevância, embora com um défice grande de aplicações genéricas capazes de auxiliar os utilizadores. Porém, com o auxílio de pesquisas científicas como as de Kari Edison Watkins, Brian Ferris, Alan Borning, G. Scott Rutherford e David Layton ou Pengfei Zhou, Yuanqing Zheng e Mo Li consegue saber-se o estado em que está a evolução de tudo a que aos transportes e monitorização dos mesmos diz respeito. Por outro lado, com base em estudos deste género, cada vez são as aplicações que surgem nas lojas *online* e que se apresentam disponíveis para o público, de forma gratuita ou não.

Do conjunto de aplicações analisadas, há um lote de duas, três funcionalidades comuns que se destacam e que estão relacionadas com uma melhor experiência de utilização por parte dos utilizadores das quais é importante salientar o acesso ao tempo que falta para o próximo autocarro, o sistema de favoritos e ainda a possibilidade de visualizar as diversas paragens no mapa. Contudo, para que as aplicações funcionem tal como esperado, é necessária uma base que disponibilize a informação, neste caso, as API's de camada intermédia. Estas API's, genéricas ou implementadas para o cliente específico que se destinam a servir, têm a capacidade de reunir a informação e a disponibilizar, tornando-a acessível em qualquer lugar, a qualquer hora.

Capítulo 3

Abordagem Proposta

O facto de não existirem soluções práticas de partilha de informação pensadas para as empresas de transporte e, juntando a isso, o facto não existir o mesmo tipo de soluções de auxílio aos passageiros, também, deu-se a necessidade de estudar e implementar uma solução que resolvesse este problema, como já referido anteriormente neste documento. Desta forma, e focando agora a realidade portuguesa, esta secção irá apresentar a abordagem tomada e arquitetura desenhada para as soluções cliente e servidor (das empresas). Para isso, é necessário que sejam definidos previamente os requisitos que se pretendem cumprir, bem como as tecnologias utilizadas para esse efeito.

3.1 Introdução

O ter que sair de casa, da escola ou do estabelecimento de trabalho a horas para entrar no próximo autocarro para que tudo corra conforme o planeado é um desafio diário para muitas pessoas. Porém, e apesar do inquérito realizado a algumas empresas de transporte (ver Anexo G) concluir que não tem havido um aumento de passageiros a utilizar transportes públicos, são alguns os estudos que provam que a disponibilização, ao passageiro, da informação das entidades que prestam o serviço de transporte, bem como, os detalhes desses mesmos serviços ajudam a poupar algum tempo nas suas deslocações para as paragens. Isto é conseguido uma vez que estes passageiros organizam a sua vida de modo a sair dos locais onde se encontram apenas em momentos antes à passagem do autocarro. Com base nisto e no questionário efetuado que comprova que a existência de uma aplicação móvel é uma mais valia pois auxilia os passageiros nas suas deslocações diárias, a solução a implementar tem como objetivo complemen-

tar, sempre que possível, as soluções já existentes no mercado português no que ao tema dos transportes diz respeito, pois, apesar de bem conseguidas, ainda apresentam algumas falhas ao nível de interface com o utilizador e dependência constante de uma rede de dados ativa mas, sobretudo, porque não permitem a integração de várias fontes de informação, como é caso do que se pretende obter no final da implementação deste sistema, resultante do estudo efetuado.

3.2 Solução Proposta

3.2.1 Requisitos de Sistema

A especificação de requisitos de um sistema visa identificar e documentar as características do mesmo, assegurando que todos estes correspondem aos objetivos a que se propõem. Para isto e como princípio de construção de qualquer sistema ou aplicação informática, o levantamento de requisitos e definição de entidades ajudam na criação de um sistema robusto que não sofra demasiadas alterações no seu âmbito, aquando da sua conceção. No caso da solução em questão, estes foram definidos tendo em conta a análise efetuada e documentada no capítulo anterior bem como pelo questionário efetuado às empresas de transporte que revelou, por exemplo, para a solução cliente, que seria interessante a presença de funcionalidades como o cálculo de rotas, acesso a detalhes de paragens ou até mesmo à informação institucional das empresas.

Para que esta etapa seja realizada com sucesso e para que estes requisitos não sejam alterados durante a conceção do sistema é importante que sejam definidos com uma visão geral e final do do mesmo, de modo a evitar, ao máximo, alterações desnecessárias que coloquem em causa o prazo previsto para a entrega do produto final.

Requisitos Funcionais

Os requisitos funcionais descrevem as características de um sistema, partindo do ponto de vista dos seus utilizadores. Perante a arquitetura que de seguida será apresentada, o cliente é a aplicação móvel utilizada pelos passageiros de modo a terem acesso à informação.

Tabela 3.1: Requisitos Funcionais de Paragens

| # | Requisito | Prioridade |
|----|--|------------|
| F1 | O sistema deverá permitir ao utilizador ter acesso às paragens mais próximas da sua localização atual | Alta |
| F2 | O sistema deverá permitir ao utilizador ter acesso a todas as paragens existentes | Alta |
| F3 | O sistema deverá permitir ao utilizador procurar uma paragem pelo nome da rua/zona onde esta se encontra | Alta |
| F4 | O sistema deverá permitir ao utilizador visualizar todas as paragens no mapa | Alta |
| F5 | O sistema deverá permitir ao utilizador consultar os detalhes de uma paragem após a seleccionar | Alta |
| F6 | Nos detalhes de uma paragem, o sistema deverá informar o utilizador da rota a que esta pertence, bem como do código do mesmo (caso exista) | Alta |
| F7 | O sistema, através dos detalhes de uma paragem, deverá permitir ao utilizador calcular a rota para um destino, a partir daquela paragem | Média |
| F8 | O sistema deverá permitir ao utilizador procurar todas as paragens existentes num determinado raio de procura, por si definido | Alta |
| F9 | O sistema deverá permitir ao utilizador listar todas as paragens por rota | Média |

As rotas são o itinerário tomado pelos diversos autocarros. Posto isto, é expectável que o cliente móvel, enquanto versão final, execute as funcionalidades listadas na tabela 3.2, dos requisitos funcionais.

Tabela 3.2: Requisitos Funcionais de Rotas

| # | Requisito | Prioridade |
|-----|--|------------|
| F10 | O sistema deverá permitir ao seu utilizador introduzir uma paragem origem e uma paragem destino e ter acesso à rota (paragens intermédias) | Alta |
| F11 | O sistema deverá permitir ao utilizador listar as rotas existentes | Alta |
| F12 | O sistema deverá informar o utilizador do se existe transbordo na rota até chegar ao destino da rota seleccionada | Alta |
| F13 | O sistema deverá permitir ao utilizador consultar os detalhes de uma rota seleccionada | Alta |
| F14 | O sistema deverá permitir ao utilizador consultar a lista de paragens contidas em cada rota | Alta |
| F15 | O sistema deverá permitir ao utilizador consultar a rota no mapa | Alta |

Apesar do questionário não concluir que é uma funcionalidade importante, em contraste com a análise da aplicações efetuada no capítulo 2, optou-se pela adesão ao sistema de favoritos. Isto significa que a inclusão de um sistema deste género na solução proposta a tornará mais viável e melhorará a experiência de utilização por parte do seu utilizador, evitando assim que este tenha que percorrer um número elevado de passos a fim de chegar a uma funcionalidade num menu recôndito. Assim, pensou-se incluir um sistema de favoritos que auxilie o utilizador a aceder tanto às paragens como às rotas. Neste acesso será disponibilizada informação sobre essas mesmas paragens ou rotas bem como lhe dará a possibilidade de eliminar ou adicionar novos.

Tabela 3.3: Requisitos Funcionais de Favoritos

| # | Requisito | Prioridade |
|-----|--|------------|
| F16 | O sistema deverá permitir ao utilizador adicionar uma rota aos favoritos | Alta |
| F17 | O sistema deverá permitir ao utilizador adicionar uma paragem aos favoritos | Alta |
| F18 | O sistema deverá permitir ao utilizador adicionar uma empresa de transporte aos favoritos | Alta |
| F19 | O sistema deverá permitir ao utilizador remover uma rota dos favoritos | Alta |
| F20 | O sistema deverá permitir ao utilizador remover uma paragem dos favoritos | Alta |
| F21 | O sistema deverá permitir ao utilizador remover uma empresa de transporte aos favoritos | Alta |
| F22 | O sistema deverá permitir ao utilizador consultar as suas rotas favoritos | Alta |
| F23 | O sistema deverá permitir ao utilizador consultar as suas paragens favoritas | Alta |
| F24 | O sistema deverá permitir ao utilizador consultar as suas empresas de transporte favoritas | Alta |

No que toca à interação do utilizador com o sistema, esta tem de ser pensada com cautela uma vez que a aprovação e o sucesso da aplicação dependem desta. O sistema Android a nível de personalizações continua em vantagem em relação aos outros sistemas operativos muito por culpa de ser baseado em Linux e nas suas diversas Distros [14]. Isto significa que os utilizadores de uma aplicação para este sistema esperam que possa haver alguma personalização e/ou alteração das configurações.

Dado que nem todos os requisitos são possíveis de agrupar por categorias, a tabela apresentada de seguida contém os restantes requisitos funcionais desta solução. Estes encontram-se, também, organizados pela sua prioridade e completam o leque dos existentes para esta solução.

Tabela 3.4: Outros Requisitos Funcionais

| # | Requisito | Prioridade |
|-----|---|------------|
| F25 | O sistema deverá permitir ao utilizador efetuar o <i>backup</i> das preferências do utilizador | Alta |
| F26 | O sistema deverá permitir ao utilizador aceder às funcionalidades em modo <i>offline</i> | Alta |
| F27 | O sistema deverá permitir ao utilizador configurar atualizações iniciais | Alta |
| F28 | O sistema deverá permitir ao utilizador ter acesso aos detalhes de uma paragem através do QRCode | Baixa |
| F29 | O sistema deverá permitir ao utilizador ter acesso aos detalhes de uma rota através do seu QRCode | Baixa |
| F30 | O sistema deverá permitir ao utilizador consultar contatos que lhe possam ser úteis | Baixa |
| F31 | O sistema deverá permitir ao utilizador configurar um alerta para quando o autocarro estiver a chegar a determinada paragem | Média |
| F32 | O sistema deverá apresentar ao utilizador os preçários das viagens | Baixa |
| F33 | O sistema deverá conter uma secção de notícias | Baixa |
| F34 | O sistema deverá permitir ao utilizador configurar o tempo de aviso prévio para o alerta ser despoletado | Média |

Requisitos Não Funcionais

Os requisitos não funcionais descrevem as restrições de um sistema com impacto no produto final. Neste sistema estão descritos na tabela 3.6 e englobam tudo o que uma aplicação móvel deste género depende.

Tabela 3.5: Requisitos Não Funcionais

| # | Requisito |
|------|---|
| NF1 | O cliente nesta arquitetura deverá ser uma aplicação móvel |
| NF2 | O sistema deverá ter menus e uma estrutura simples |
| NF3 | O sistema deverá apresentar cores claras de modo a que o seja possível utilizá-lo sem problema em ambientes de muita claridade |
| NF4 | O sistema deverá ajustar-se a diferentes tamanhos de ecrãs |
| NF5 | O sistema deverá permitir ao utilizador interagir com a aplicação enquanto esta descarrega os dados |
| NF6 | O sistema deverá permitir ao utilizador escolher a forma como pretende atualizar a sua aplicação (3G/4G ou <i>Wifi</i>) |
| NF7 | O sistema deverá atualizar-se automaticamente, caso haja uma ligação ativa, entre os períodos de tempo definidos pelo utilizador |
| NF8 | O sistema deverá manter o seu comportamento e número de funcionalidades caso exista ou não uma ligação à Internet |
| NF9 | O sistema deverá manter em <i>cache</i> os dados que são acedidos frequentemente, evitando assim acessos desnecessários ao servidor da informação |
| NF10 | A aplicação estará disponível para ser utilizada desde a versão de Android 2.2 até à mais recente |
| NF11 | A aplicação cliente fará uso do sistema de localização do Android |

Entidades

Da apresentação dos requisitos que se pretendem cumprir no final da implementação deste sistema, surgem então as entidades. Entendem-se por entidades o que é especificado na definição de requisitos, sejam eles funcionais ou não funcionais. Estas têm como objetivo facilitar a perceção do sistema durante a sua fase de estudo, pois enumeram tudo aquilo que é abordado no momento da sua conceção. Neste trabalho, as entidades definidas encontram-se listadas na tabela seguinte.

| Entidades | |
|------------------|--|
| Sistema | Aplicação cliente responsável por aceder ao conteúdo e disponibilizá-lo ao utilizador. |
| Paragem | Local onde o autocarro para de modo a acolher e deixar passageiros. |
| Rota | Itinerário de determinado autocarro. É composto por um conjunto de paragens. |
| Serviço | Tipo de serviço prestado. É a oferta dos vários tipos de transporte pelas diversas empresas. |
| Empresa | Entidade que fornece um ou mais serviços. |

Desta definição de entidades surge a necessidade de as representar através de um DER (Figura 3.1). A representação através deste tipo de diagrama tem como objetivo descrever o modelo de dados de forma abstrata através da atribuição de características a estas entidades.

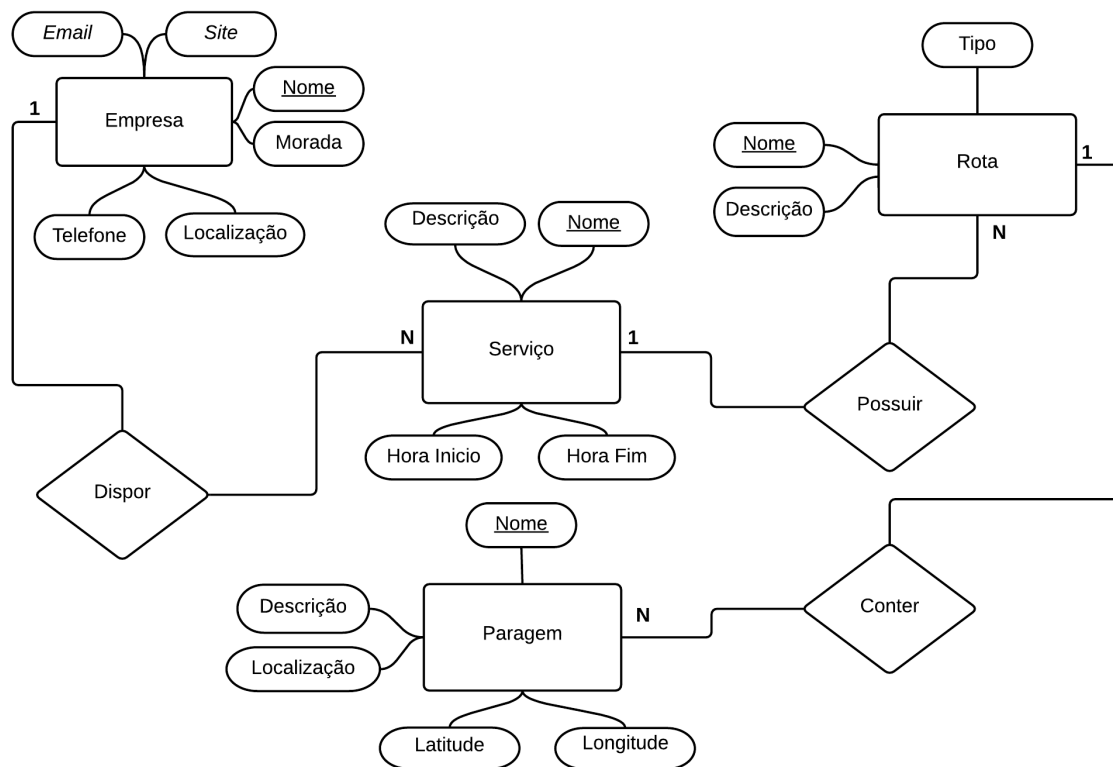


Figura 3.1: Diagrama de Entidade Relacionamento

Assim, através do DER apresentado é possível verificar que são 4 as entidades necessárias para a conceção desta solução e são elas a Empresa, a Rota, o Serviço e a Paragem. A Empresa é composta pelos atributos nome (que a identifica perante as outras), *email*, *site*, *telefone*, *morada* e *localização*. Cada empresa dispõe de Serviços que são divulgados ao utilizador sendo estes identificados por uma hora de início e de fim, uma descrição e um nome. Para além destes, estão presentes ainda as Rotas que, fazendo parte dos serviços, contêm paragens. Estas rotas distinguem-se de outras entidades através dos seus atributos descrição, tipo e nome. No entanto, e fazendo parte de todo este Sistema composto pelas entidades atrás referidas, estão ainda as Paragens que se caracterizam pelo seu nome, a sua descrição, latitude, longitude e localização.

3.2.2 Arquitetura

De modo a dar resposta aos objetivos definidos neste trabalho, é proposto um sistema que faça uso da arquitetura cliente-servidor pois a centralização da informação faz com que o processamento seja mais rápido, pois é efetuado por um sistema próprio e concebido diretamente para o fazer, ao invés de arquiteturas *peer-to-peer* ou até mesmo

híbridas. Este tipo de arquitetura é amplamente utilizada hoje em dia e não é mais do que a disponibilização de informação - dados - por um ou mais servidores para um "consumidor" final, também conhecidos como clientes. Estes clientes apenas têm como funções obter e decodificar a informação fornecida pelo servidor, sendo que para isso apenas tenham que ter uma ligação com o servidor (neste caso ligação de dados ou *wifi*).

Como se pode visualizar na Figura 3.2, o sistema estará disponível sempre que o utilizador entender, esteja este onde estiver. Para isto, basta fazer uso do cliente, neste caso a aplicação móvel, e consultar as paragens, as rotas e os serviços, entre outros. Caso esta consulta seja realizada sem acesso à Internet, a aplicação manterá os dados acessíveis na mesma, uma vez que serão presadas as funcionalidades fundamentais em modo *offline*. Em contrapartida, a utilização de uma ligação de rede móvel garante um melhor funcionamento da aplicação cliente pois permite obter a informação correta depois desta ser 'pedida' ao servidor, aquando das atualizações automáticas. No que toca à parte que serve a informação, esta estará alojada num ou mais servidores *web* que estarão dispostos, após um processo transparente para o utilizador, a disponibilizar os dados pedidos pelo cliente, tal como qualquer servidor deste género.

A escolha deste tipo de arquitetura é a que mais se adequa à solução uma vez que toda a informação está centralizada sendo assim mais fácil a sua alteração caso necessário. Desta forma, consegue-se também dar resposta a um maior número de clientes, tornando-se o servidor o elemento chave neste processo.

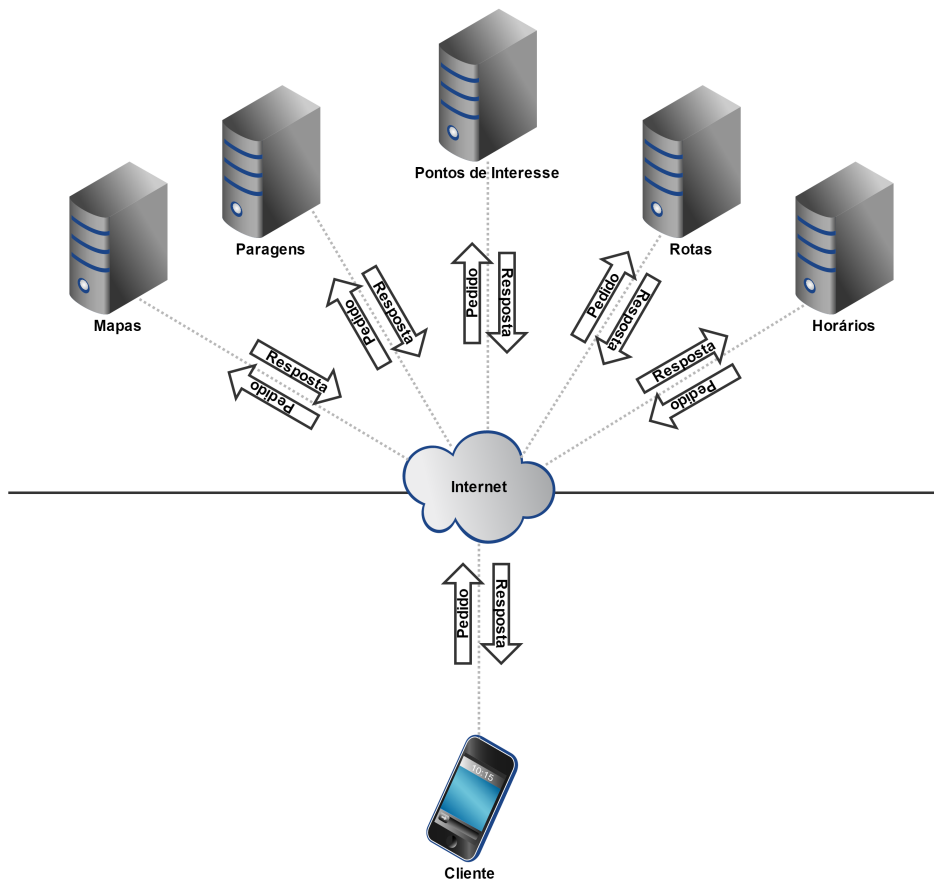


Figura 3.2: Arquitetura do sistema - Visão de alto nível

3.2.3 Cliente

O cliente nesta arquitetura é o consumidor da informação disponibilizada pelo servidor. Este será implementado para o sistema operativo móvel Android, uma vez que esta solução se centrará num nicho de utilizadores específico. Este tipo de utilizadores são pessoas que usam constantemente o autocarro como meio de transporte e que, ao mesmo tempo, são portadoras de um *smartphone*. Assim, assumindo que o Android é o sistema operativo mais utilizado em Portugal [15] muito devido ao custo dos dispositivos [16] e sendo este *open source*, optou-se por esta plataforma para implementar a solução. O facto de dominar o mercado e em grande parte sendo lançado em equipamentos de grandes marcas como Samsung, HTC, Motorola, LG, Sony, Asus, Acer ou BQ, foi mais um dos fatores a influenciar esta escolha.

3.2.4 Servidor

O servidor, tal como já anteriormente referido, é aquele que fornece a informação. Em sistemas deste género é este o responsável por responder a tudo o que lhe é pedido, sendo um ponto fundamental neste tipo de arquiteturas. Neste caso, o servidor responderá aos pedidos efetuados pelo dispositivo cliente, do passageiro, sempre que este invocar uma atualização dos dados ou sempre que o cliente estiver implementado para o fazer automaticamente.

Assim, o servidor conterà toda a informação acerca de paragens e rotas, entre outros, devolvendo toda esta informação ao cliente, sempre que pedida. Esta informação será proveniente através de duas tecnologias gratuitas, de seguida introduzidas e explicadas em detalhe.

Serviços a Integrar

OpenStreetMap: Como já introduzido no estado da arte, o *OpenStreetMap* é um serviço de mapas aberto que se baseia no conceito de *participatory sensing*, isto é, em casos práticos, na contribuição e partilha de informação por parte dos seus utilizadores. A sua inclusão neste sistema tem como objetivo demonstrar que é possível a introdução de serviços tornando os sistemas mais completos em termos de fontes de informação, mantendo ao mesmo tempo a sua escalabilidade e constante *performance*. No entanto, a integração deste serviço de mapas tem pontos negativos pois a alteração de paragens existentes e acedidas pelo cliente, podem ser editadas por qualquer pessoa registada neste serviço.

General Transit Feed Specification: O GTFS¹ conhecido por "Especificação Geral de Feeds de Transporte Público", em português, define um formato de ficheiros comum para a partilha de dados entre as empresas de transportes públicos e programadores de aplicações com o auxílio de dados geográficos. Este formato foi criado pela Google que muito se esforçou pela integração com o seu serviço de mapas² e de trânsito³ já

¹<https://developers.google.com/transit/gtfs/>

²maps.google.com

³maps.google.com/transit

existente.

Feed: Um *feed* é um conjunto de ficheiros de texto agregados que resultarão num único ficheiro com a extensão ".zip". Cada ficheiro contém detalhes específicos de empresas de transportes públicos para o qual foram desenhados. Estes detalhes são separados pelos diversos ficheiros de texto e contém informação acerca de paragens, rotas e entre outros, horários.

Para que tudo funcione de acordo com as regras definidas para esta referência, é necessário conjugar os diversos tipos de *feed*. Como tal, existem dois tipos de termos para classificar cada *feed* e atributos do mesmo. Estes termos são os **campo obrigatório** e **campo opcional**. Os campos obrigatórios devem constar sempre nos ficheiros, enquanto que os opcionais podem ser omitidos. No entanto, se um campo existir, sendo opcional, mas sem valor definido, deverá encontrar-se entre vírgulas, para ser corretamente interpretado.

Assim, os ficheiros obrigatórios em cada *feed* são:

agency.txt - Informação acerca da empresa que presta o serviço.

calendar.txt - O horário de uma rota ativa.

routes.txt - Todas as rotas disponíveis.

stoptimes.txt - Tempos de paragens nos locais identificados no *stops.txt*.

stops.txt - Paragens de cada rota.

trips.txt - Voltas dentro de cada rota.

Para além destes, há ainda os opcionais que podem e idealmente, devem ser definidos pois tornam o sistema mais completo. Esses mesmos ficheiros de *feed* são:

calendardates.txt - Contém as exceções às datas presentes no ficheiro *calendar.txt*.

Neste ficheiro encontram-se, normalmente, feriados.

fareattributes.txt - Informação sobre os preços de viagens.

farerules.txt - Define a forma como os preços definidos no ficheiro *fareattributes.txt* são aplicadas a uma rota.

shapes.txt - Contém regras para desenhar uma rota no mapa.

frequencies.txt - Horários e frequências numa determinada paragem.

transfers.txt - Define as regras de transbordo entre paragens numa rota.

feedinfo.txt - Informação do *feed* atual tal como a sua validade e a sua versão.

Modo de Funcionamento: De forma a propagar a informação contida nestes *feeds* é necessário que os mesmos estejam presentes na *web*. Após isso, estarão disponíveis para quem fizer uso destes, funcionando assim como qualquer outro tipo de serviço presente neste meio. Assim sendo, e depois de alojados, as aplicações cliente não necessitarão de permissão alguma para os consumir. No entanto, há duas formas de facilitar a propagação dessa informação e são elas:

- *GoogleTransitDataFeed*⁴;
- *GTFS Data Exchange*⁵.

As duas formas de publicação são válidas. Ainda assim, a mais popular entre os programadores de aplicações é a segunda, uma vez que apenas basta um registo para publicar os *feeds* e automaticamente estes ficam disponíveis para utilização. Este projeto foi criado por Jehiah Czebotar⁶ com o objetivo de, face ao *GoogleTransitDataFeed*, eliminar o tempo de espera entre a publicação do *feed* e a sua disponibilização. Contudo, apresenta algumas desvantagens em relação ao sistema da Google pois em caso algum faz a validação dos *feeds* submetidos, ou fornece ferramentas para o fazer. Por outro lado, o *GoogleTransitDataFeed* apresenta ferramentas para efetuar a validação de *feeds* e apenas os validados pelo programador e numa fase posterior, pela Google, serão publicados, tornando o processo moroso.

General Transit Feed Specification - Real Time: Embora com um nome semelhante ao GTFS as diferenças entre este e o GTFS *RealTime* são vastas, pois o primeiro divulga a informação estática e que apenas carece de alterações ocasionalmente, enquanto que o segundo colmata esta "falha" e surgiu como um sistema dinâmico de partilha de dados . A especificação *RealTime* como já abordada neste documento (ver secção 2.3.7 - *GTFS - RealTime*), baseia-se na informação em tempo real que se centra em três pontos, atualizações de viagem, alertas de serviços e posições de veículos. Assim, a inclusão neste sistema visa completar o já existente GTFS uma vez que este, por si só, não dá resposta a tudo o que uma entidade de transportes necessita, retra-

⁴<https://code.google.com/p/googletransitdatafeed/>

⁵<http://www.gtfs-data-exchange.com/>

⁶<http://jehiah.cz/>

tando-se como uma mais valia.

3.2.5 Abordagem Escolhida

De forma a responder ao problema da não existência de uma forma das empresas de transportes partilharem facilmente a sua informação com os seus passageiros e ainda com o objetivo destes usufruírem de informação atualizada a todo o momento, foram abordadas algumas formas de o fazer, embora a "oferta" seja ainda bastante reduzida.

No âmbito dos transportes públicos, para além de API's particulares de partilha de informação criadas pelas próprias empresas, ou as genéricas mas para um conjunto de duas ou três cidades de uma região (ver secção 2.3 - Infra-Estruturas de Partilha de Informação) e das API's que fazem uso do GTFS, criado pela Google, não é comum outro tipo de soluções que orientem na resolução do problema apresentado.

Posto isto, a opção foi retratar um caso de estudo para a cidade de Leiria, fazendo uso do serviço de transportes Mobilis, da Rodotejo [17]. Este caso de estudo foi tido em conta dada a localização deste serviço de transporte, sendo possível estender este sistema a qualquer outra cidade, podendo também ser conjugadas várias cidades com uma mesma aplicação cliente e um mesmo sistema servidor, através do GTFS.

Por outro lado, esta escolha ganhou alguma vantagem em relação ao sistema de API's particulares dado que, não só do ponto de vista de implementação como também do ponto de vista do cliente apresenta um custo inicial menor e é menos complexa a sua implementação. Apesar desta escolha, nada invalida a possibilidade de integrar outro tipo de serviços para complementar esta tecnologia criada pela Google. Ao serem referidos estes serviços está a incluir-se o *OpenStreetMap*, para que se demonstre que é possível a integração de vários serviços num só de forma a criar uma solução compacta, adequada na resposta ao problema e passível de ser utilizada todos os dias pelos passageiros.

3.2.6 Detalhe e Integração de Sistemas

De forma a assegurar que a comunicação entre o cliente com os fornecedores de informação é realizada da melhor forma, irá ser explicado nesta secção em pormenor como se vai processar essa comunicação.

Analisando a Figura 3.3, pode verificar-se a comunicação e integração dos serviços com

a aplicação cliente sendo eles o *OpenStreetMap* e o *General Transit Feed Specification*, fazendo uso das duas versões, a estática GTFS e a dinâmica, a *RealTime*.

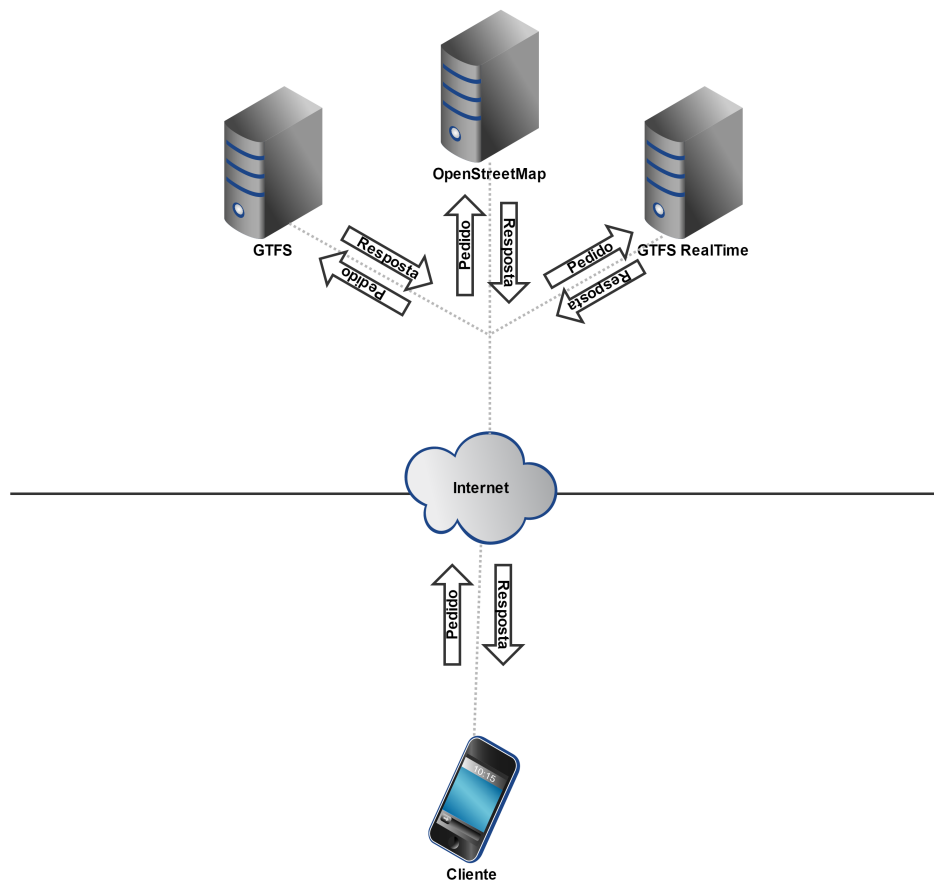


Figura 3.3: Arquitetura do sistema - Integração de Serviços

O cliente fará todas as chamadas necessárias aos serviços disponibilizados. Na figura acima, os serviços GTFS encontram-se separados pois apesar de partilharem o mesmo nome, são serviços com especificações e formas de utilização bastante diferentes daí que, também a forma de interação com a informação nestes contida seja, naturalmente, diferente também.

Apesar do *OpenStreetMap* ser um serviço bastante utilizado hoje em dia é necessário garantir uma forma correta de obtenção da informação. Para esse efeito existem algumas API's responsáveis e orientadas para esse efeito.

- Xapi;

- *OpenLayers 3*;
- *Leaflet*;
- OSM API;
- *MapQuest*;
- *OverPass API*;

Das anteriores mencionadas, é importante realçar a própria OSM API, a Xapi, a MapQuest e a OverPass API. As duas restantes, *OpenLayers 3*⁷ e *OverPass*⁸ são incluídas nestes estudo apenas como referência à sua existência e ao trabalho desenvolvido para contribuir para o aumento do sucesso da utilização do *OpenStreetMap*, uma vez sendo desenvolvidas em JavaScript e não se aplicarem ao âmbito e à arquitetura neste capítulo já apresentados.

Retratando agora as restantes, a OSM API⁹ é a proprietária da OSM Foundation, tendo sido desenvolvida para obter e editar dados diretamente das bases de dados do *OpenStreetMap*. É uma API REST comum que se baseia em pedidos HTTP com comandos GET, PUT, POST e DELETE, devolvendo os dados em forma de XML. A escolha para a solução proposta recaiu sob esta forma de interação com os dados pois é "oficial" do OSM e o facto de não necessitar de sistemas intermédios de terceiros é uma mais valia. O facto de ser a mais completa (pois permite a edição e consulta da informação) em relação às restantes foi mais um dos pontos a ter em conta. Semelhante à OSM API existe a Xapi, ou *OSM Extended API*¹⁰ que é uma versão alterada da OSM API, uma vez que permite apenas a leitura da informação, tornando-a assim mais rápida. Existem já algumas implementações e API's baseadas nesta Xapi, como é o caso da *MapQuest*¹¹ e ainda da *OverPass API*¹². A primeira é proprietária sendo assim conhecida como um sistema intermédio, uma vez que recebe os dados do *OpenStreetMap*, os processa e os devolve, tornando o processo mais lento. Ainda assim, esta é mais abrangente nos serviços que oferece, como por exemplos, implementações em *Javascript* ou *Flash*. Do ponto de vista do cliente, embora assente sobre a Xapi, as suas chamadas para interação com os dados são mais complexas.

Das OSM API's referidas, é necessário ainda referir a *OverPass*¹³ que, tal como a Xapi,

⁷<http://openlayers.org/>

⁸<http://leafletjs.com/>

⁹<http://api.openstreetmap.org/>

¹⁰<http://wiki.openstreetmap.org/wiki/XAPI>

¹¹<http://open.mapquestapi.com/>

¹²http://wiki.openstreetmap.org/wiki/Overpass_API

¹³<http://overpass-api.de/>

está otimizada para o melhor desempenho e devolução dos dados ao cliente de forma mais rápida, mas que, como a *MapQuest*, faz parte de um sistema de terceiros.

Tal como para o *OpenStreetMap*, também para o GTFS foram criadas algumas API's para dar consumir e responder aos pedidos de clientes. Neste caso, é visível a clara separação entre as API's criadas para a divulgação de informação estática e de informação dinâmica, dada a sua complexidade e especificação serem diferentes, como já referido anteriormente. Apesar da Google se ter afastado neste ponto, existiram entidades que decidiram estudar a especificação para criar uma forma de consumir essa informação. Daí surgiram algumas API's das quais se destacam as seguintes:

- *RouteShout*;
- *GeoGTFS*;
- *OneBusAway*;
- *GTFS DataExchange* API;
- *TransLink*;
- *Node-GTFS*;

A primeira apresentada, *RouteShout*¹⁴ é uma API gratuita apenas para uso não comercial. A forma de "entrega" de dados ao cliente é feita através de JSON sendo que para que esta "entrega" seja concluída com sucesso é necessária uma chave identificadora da entidade requerente, neste caso, a entidade prestadora do serviço de transporte. A grande desvantagem em relação ao uso deste sistema seria o facto das empresas de transporte terem que estar registadas na base de dados da *RouteMatch, Inc.*, companhia que faz a gestão do mesmo. Caso não esteja presente, é necessária a criação de um acordo e, após este ser consumado, é que se poderá fazer o registo e usufruir da referida API, o que torna todo o processo complexo. Por outro lado, e com um modo de funcionamento mais simples, encontra-se a *GeoGTFS*, API que não requer qualquer tipo de autenticação e foi criada para que os clientes acedam à informação contida nos ficheiros ".txt" diretamente e não através de um *download* prévio. Neste momento, esta API agrega todas as empresas de transporte presentes no site *GTFS Data-Exchange* e devolve informação acerca de rotas e paragens das mesmas.

No entanto, a mais completa de todas é a *OneBusAway*¹⁵ (ver secção 2.1 - Infra-Estruturas de Obtenção de Informação) criada por um dos grandes responsáveis pelo

¹⁴<http://www.routeshout.com/main/api>

¹⁵<http://onebusaway.org/>

sistema de trânsito da Google neste momento, Brian Ferris¹⁶. Atualmente este sistema está presente em cinco cidades dos Estados Unidos da América. O mais interessante deste também conhecido "*The OneBusAway Project*" é o facto de serem disponibilizadas inúmeras ferramentas para trabalhar com a referência criada pela Google, o GTFS "estático" e "dinâmico", o *RealTime*.

Embora com modo funcionamento diferente, a *TransLink*¹⁷ é uma API de *RealTime* que assenta no conceito de partilha de informação em tempo real para veículos equipados com um sistema *TransLink's Trip-Tracker* que não é mais que um dispositivo GPS instalado nos autocarros e que partilha a sua localização em cada momento, tirando o máximo partido da especificação do GTFS RT, as atualizações de viagem, os alertas de serviço e ainda as posições de veículos.

Tal como a *OneBusAway*, a *Node-GTFS* é mais uma excelente ferramenta de interação com os dados definidos nos ficheiros ".txt" que compõem a referência GTFS. Esta API carrega os ficheiros ".zip" do site *GTFS DataExchange*, extrai cada um deles e guarda-os numa base de dados *online*, à qual o cliente acede através de *queries* pré-definidas por esta mesma API. Com este mecanismo, é possível obter informação acerca de empresas de transporte, rotas, paragens e tempos de viagens.

Ainda sobre o site *GTFS DataExchange* é necessário fazer a referência à API que embora de dimensão reduzida, devolve as entidades prestadores de serviços e detalhes destas presentes e registadas neste mesmo site.

3.3 Conclusão

Diariamente são muitas as pessoas que utilizam transportes públicos. Como tal, fornecer-lhes uma aplicação que lhes permita facilitar esta tarefa é o ponto chave deste trabalho. Por outro lado, facilitar a tarefa às empresas de transportes é também um dos pontos importantes, uma vez que as soluções existentes no mercado apenas estão direcionadas para serem consumidas por aplicações cliente próprias. Para isso e para que a solução final seja coerente com o que é especificado, é necessário que a abordagem defina corretamente a forma como funcionará todo o sistema.

Posto isto, uma clara definição de objetivos e requisitos leva a que o produto não só esteja dentro dos prazos de entrega, como também, seja aquilo a que se propôs. Como

¹⁶<https://plus.google.com/+BrianFerris>

¹⁷<https://gtfsrt.api.translink.com.au/>

consequência e após a especificação de requisitos levantados não só na sequência da análise efetuada no capítulo anterior, como também, no questionário realizado a algumas empresas de transporte, é necessário estudar mecanismos que permitam a implementação desses mesmos objetivos propostos. Para esta implementação, surgem API's e tecnologias existentes no mercado que conjugadas, auxiliam e dão resposta ao problema existente.

Neste sistema, a abordagem propõe um sistema compacto baseado na integração várias fontes de informação como é o caso do GTFS e GTFS *RealTime*, da Google e ainda o sistema de mapas aberto, *OpenStreetMap* que assenta no conceito de *Participatory Sensing*. O primeiro, o GTFS, garante e devolve ao cliente a informação sobre os serviços da empresa de transportes como horários, rotas, horários, entre outros, enquanto que o segundo, o *RealTime* divulga as informações em tempo real e são elas alertas de serviços existentes nas rotas, posições dos veículos e ainda atualizações de viagem. A juntar a estes dois serviços, existe ainda o *OpenStreetMap* que será integrado como serviço de mapas deste sistema.

Capítulo 4

Implementação

Uma vez terminada a proposta da abordagem tomada para dar resolução ao problema existente é então necessário detalhar, tecnicamente, a forma de o resolver, transcrevendo as ideias abordadas, conjugando-as de forma a criar um produto final à medida do que é pretendido.

4.1 Tecnologias

Assim sendo, esta secção é responsável por apresentar as tecnologias utilizadas aquando da implementação da solução. Estas tecnologias estão incluídas num vasto leque de ferramentas que se encontram ao dispor de todos e, como tal, a melhor seleção das mesmas garante uma maior probabilidade de sucesso aquando, não só do desenvolvimento do produto, como também, posteriormente, em ambiente de produção, ou seja, quando em utilização pelos passageiros das companhias de transporte.

4.1.1 Servidor

OpenStreetMap

Na abordagem efetuada no capítulo anterior foram apresentadas várias API's possíveis para a comunicação e obtenção de dados presentes no OSM. Excluindo as duas implementadas em *Javascript*, *OpenLayers 3* e *Leaflet* (pelo facto de não se adequarem à arquitetura proposta), restam então a própria do *OpenStreetMap*, *Xapi*, *MapQuest* e *OverPass API*. Tendo em conta as formas de funcionamento de cada uma das restantes e após a análise da documentação das mesmas, era de supor que a *Xapi* seria a mais

indicada para esse efeito uma vez que não é um sistema de terceiros e está otimizada para o melhor desempenho, dado que apenas permite efetuar *queries* de consulta. No entanto, aquando do início da implementação da solução, esta API deixou de fazer sentido pois não reproduzia as alterações efetuadas às paragens introduzidas no editor de mapas do *OpenStreetMap*. De seguida, embora não sendo a ideal, o objetivo era colocar a *MapQuest* operacional de forma a devolver a informação requerida e previamente introduzida no sistema de mapas. Tendo em conta que prevalecia o facto de ser uma API proprietária do OSM e visto que ambas foram colocadas em funcionamento com sucesso, a escolha recaiu pela *OSM API* que permite não só consultas como também edição de dados.

Funcionamento: O *OpenStreetMap* apresenta uma forma simplista de representar a informação geograficamente, baseada em sistemas GIS. Estes sistemas são conhecidos por representarem os seus dados através de três tipos de objetos: pontos, linhas e polígonos. Baseando-se nesta analogia deu-se a criação dos *nodes*, *ways* e *relations* no OSM que fazem uso de *tags* para detalhar e garantir características aos elementos. Os *nodes*, à semelhança dos sistemas GIS, são pontos que representam objetos no mapa. Estes objetos são lojas, paragens de autocarro, caixas de correio, bancos de jardim, entre outros. As *ways* são as formas de efetuar a ligação entre pontos como passeios, estradas, linhas de comboio ou linhas elétricas. Por último existem as *relations* que não são mais que o resultado da agregação de *ways* e *nodes*, tal como os polígonos em GIS. Para retratar as paragens de autocarro de uma cidade, por exemplo, deve definir-se uma *relation* com todos os elementos de forma a que facilite o acesso aos detalhes das mesmas, assim que requerido. Como tal e de acordo com a documentação da API é possível fazer a invocação ao serviço de várias formas, porém, as mais comuns são as seguintes:

- Obter a zona em questão (*GET /api/0.6/map?bbox=left,bottom,right,top*);
- Obter a *relation* e os seus dados através do seu *id*. *GET /api/0.6/[nodewayrelation]/#id*.

Obter a zona em questão significa definir "caixa" na zona do mapa correspondente à zona pretendida que, quando invocada pelo serviço, devolve todos os *nodes* e *relations* presentes dentro da mesma. Apesar de ser bastante comum a sua utilização, é

mais trabalhoso de se obter, uma vez que primeiro define-se essa mesma "caixa"¹ e só depois se pode chamar a API para ter acesso à informação. Por outro lado, e de forma mais facilitada, é utilizado o *id* da *relation* para ter acesso à informação, como de seguida se apresenta. Aqui o objetivo passa por obter as paragens definidas dentro da *relation*, para a cidade em estudo, Leiria, definida com o nome "Mobilis" ao qual foi atribuído o identificador 3518824, quando introduzidas no *OpenStreetMap*. Assim sendo, a invocação foi efetuada da seguinte forma:

<https://www.openstreetmap.org/api/0.6/relation/3518824>

Após a invocação é devolvida a seguinte resposta:

```
<osm version="0.6"
generator="CGImap_0.3.3_(29785_thorn-03.openstreetmap.org)"
copyright="OpenStreetMap_and_contributors"
attribution="http://www.openstreetmap.org/copyright"
license="http://opendatacommons.org/licenses/odbl/1-0/">
  <relation id="3518824" visible="true" version="42"
    changeset="28386255"
    timestamp="2015-01-25T01:27:21Z"
    uid="445220">
    <member type="node" ref="2677149650" role=""/>
    <member type="node" ref="2677151056" role=""/>
    <member type="node" ref="1583252216" role=""/>
    <member type="way" ref="267160421" role=""/>
    <member type="node" ref="2677253638" role=""/>
    (...)
    <member type="node" ref="3284997878" role=""/>
    <member type="node" ref="3284899928" role=""/>
    <member type="node" ref="3284928373" role=""/>
    <member type="node" ref="3284880051" role=""/>
    <member type="node" ref="3284863106" role=""/>
    <tag k="colour" v="#D7E13A"/>
    <tag k="name" v="Mobilis"/>
    <tag k="network" v="Mobilis"/>
    <tag k="ref" v="Mobilis_1"/>
    <tag k="route" v="bus"/>
    <tag k="type" v="route"/>
  </relation>
</osm>
```

¹<http://boundingbox.klokantech.com/>

Perante este excerto é importante destacar alguns elementos. Ignorando o *header* gerado pela versão da API do OSM que neste caso é a 0.6 - última lançada em 2012 - e todos os detalhes aqui presentes, é necessário salientar a entrada *relation* que se faz acompanhar de um *id* único, sob a qual é efetuada a chamada à API com o objetivo de obter as paragens. Este *id* identifica cada conjunto de elementos associados e que fazem parte do mesmo elemento-pai a que neste caso foi atribuído o valor 3518824. Esta *relation* é composta por diversos elementos *tag*, que lhe conferem propriedades como é o caso da cor associada, do nome, do tipo de elemento ou ainda do tipo de rota. Contudo, é importante ainda fazer referência ao elemento *member* presente no resultado devolvido pelo serviço. Este é composto por vários *types*, podendo estes, para cada *relation* tomar valores como *node* ou *way*. O primeiro, o mais importante para a finalidade em questão, é o que indica as paragens, sendo que o atributo *way* aponta para as ruas existentes nesta *relation*, não sendo relevante para o caso de estudo em questão.

Como consequência da divulgação das paragens definidas no mapa, dá-se a necessidade também de obter detalhe de cada uma destas. Deste modo, e uma vez preparada para esta necessidade, esta API garante o acesso a esses mesmos detalhes. No exemplo que se segue para uma chamada efetuada ao serviço para os detalhes da paragem atrás identificada através do *id* 2677253638 a conjugação de parâmetros é a seguinte: <https://www.openstreetmap.org/api/0.6/node/2677253638>

```
<osm version="0.6"
  generator="CGIMap_0.3.3_(29785_thorn-03.openstreetmap.org)"
  copyright="OpenStreetMap_and_contributors"
  attribution="http://www.openstreetmap.org/copyright"
  license="http://opendatacommons.org/licenses/odbl/1-0/">
  <node id="2677253638" visible="true" version="2"
    changeset="25736025"
    timestamp="2014-09-28T23:07:18Z"
    uid="290680"
    lat="39.7496821" lon="-8.8065583">
    <tag k="bus" v="yes"/>
    <tag k="highway" v="bus_stop"/>
    <tag k="name" v="Mercado_Municipal"/>
    <tag k="public_transport" v="platform"/>
    <tag k="shelter" v="yes"/>
    <tag k="wheelchair" v="yes"/>
  </node>
</osm>
```

Neste caso é possível tirar algumas ilações quanto à obtenção dos detalhes de uma paragem presente numa *relation*. Cada um destes detalhes é acompanhado pelo tipo de paragem, o seu nome, se é ou não coberta e ainda se permite o embarque a pessoas que se deslocam auxiliadas por cadeiras de rodas.

GTFS

Aquando da divulgação da abordagem proposta foi feita referência à utilização de várias API's que, embora não muito comuns hoje em dia, têm a capacidade de divulgar, através de métodos próprios, o conteúdo dos ficheiros presentes dentro de uma definição GTFS. Confrontando as possibilidades existentes, a escolha recaiu pela utilização da API *Node-GTFS* dado o conjunto de funcionalidades que oferece em comparação com as restantes opções apresentadas para este efeito. A *Node-GTFS* foi criada em 2012 por Brendan Nee e além de ser gratuita, tem a particularidade de não necessitar de qualquer servidor local dedicado a devolver informação, uma vez que esta se encontra centralizada *online*, como será detalhado mais à frente nesta secção. Como tal, para que a implementação e configuração deste sistema fique operacional, é necessário serem respeitadas algumas regras e fazer uso de algumas ferramentas de forma a validar os *feeds* GTFS previamente criados, como é demonstrado na tabela 4.1.

Tabela 4.1: Ferramentas GTFS

| Nome | Descrição |
|---------------------------|--|
| <i>TimeTablePublisher</i> | Converte dados de <i>feeds</i> GTFS em tabelas de horários estáticas |
| <i>Graphserver</i> | Biblioteca em python que implementa um <i>Trip Planner</i> baseado nos <i>feeds</i> GTFS incorporado com o OSM |
| <i>Trapeze</i> | Permite exportar os dados GTFS para o <i>software</i> de gestão de dados de transportes públicos <i>Trapeze FX</i> |
| <i>Trillum Solutions</i> | Gestor de <i>feeds</i> GTFS <i>online</i> |

Continua na próxima página

Tabela 4.1 – em continuação da página anterior

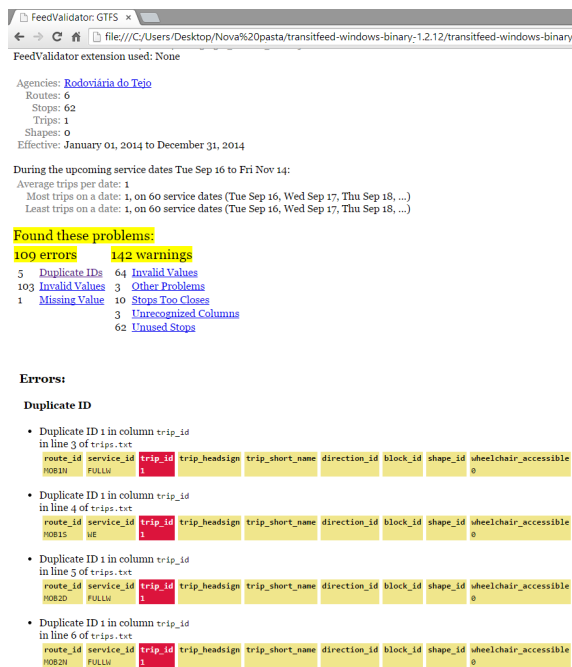
| Nome | Descrição |
|---|--|
| <i>Next Insight Transportation Software</i> | Oferece uma interface de criação de <i>feeds</i> GTFS |
| <i>Merakas</i> | Permite a leitura e criação de ficheiros GTFS |
| <i>Mentor Streets® Schedule</i> | Ferramenta que permite exportar dados dos diversos <i>feeds</i> |
| <i>GTFS - OpenStreet-Map Sync (GO-Sync)</i> | Sincroniza os dados dos <i>feeds</i> e exporta-os automaticamente para o OSM |
| <i>Stoply</i> | Permite a visualização em <i>browsers</i> móveis de horários de rotas |
| <i>TransitEditor</i> | Gestor de <i>feeds</i> GTFS |
| <i>TransitDataFeeder</i> | Ferramenta <i>online</i> de criação de <i>feeds</i> GTFS |
| <i>Transit Wand</i> | Utilitário <i>opensource</i> para obter dados de trânsito |
| <i>GTFS Editor</i> | Ferramenta que permite criar e editar <i>feeds</i> GTFS |
| <i>OneBusAway</i> | Conjunto de utilitários que permitem a interação com os serviços de trânsito da Google |

Validação de *feed*: Relembrando que o caso retratado é relativo à cidade de Leiria, é necessária a criação do seu *feed* GTFS (ver Anexo D). Após a criação deste *feed* é aconselhável a validação do mesmo, não sendo esta necessária, uma vez que o *site* de destino é o *GTFS Data Exchange*. De forma a validá-lo foi então utilizada a ferramenta *FeedValidator* disponibilizada para esse efeito pela própria Google².

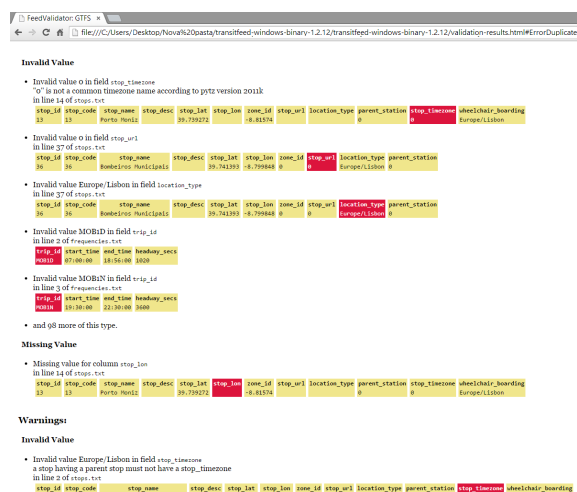
Como é possível visualizar na Figura 4.1 (a), a criação do *feed* iniciou-se com algumas dificuldades. Analisando a primeira de todas as versões deste, existiam 109 erros e 142 *warnings*, sendo que o próprio validador alertava para os dados que não estavam

²<https://developers.google.com/transit/tools?hl=pt>

corretos (Figura 4.1 (b)).



(a) FeedValidator I



(b) FeedValidator II

Figura 4.1: Validação de Dados com FeedValidator

No entanto, após a correção dos dados incorretos nos diversos ficheiros que compõem o *feed*, voltando a efetuar a validação e garantindo o correto formato dos dados, este foi então enviado para o *GTFS Data Exchange* com a garantia que não existiria problema algum.

Finalizada a validação do *feed* foi necessário obter a API, configurar e alterar o *script* necessário ao carregamento de dados para a base de dados *online*. Este sistema de base de dados foi montado baseado no conceito MongoDB³ que se caracteriza por ser um sistema de base de dados - multi-plataforma - de alta performance baseado em documentos, ignorando o formato comum e, atualmente ainda muito utilizado, de tabelas. Para além disto, este sistema de base de dados apresenta inúmeras vantagens quer ao nível de redução de custos (em comparação com o formato de tabelas), quer ao nível de *performance*. A este, poderão ainda ser atribuídas vantagens ao nível de integração com serviços de *PaaS* e segurança da informação [18]. Posto isto, o primeiro

³<http://www.mongodb.org/>

passo foi criar a base de dados. Para isso, fez-se uso da plataforma *online* disponível em <https://www.compose.io/>. Após a criação deste sistema, a base de dados ficou acessível a partir do endereço:

`mongokahana.mongohq.com:10082/bustogoDB`

Terminando a criação do sistema de armazenamento de dados foi necessário montar o sistema responsável por devolver a informação contida no *site* do *GTFS Data Exchange*. Para isto, criou-se, em ambiente local, uma réplica do que seria, posteriormente, colocado *online*, através da configuração da API para obter e decodificar os dados, inserindo os mesmos numa *MongoDB* configurada localmente, resultando numa consulta de dados como é possível verificar na Figura 4.2.

```
C:\Users\PedroF\Documents\Github\node-gtfs>node ./scripts/download
Starting rodoviria-do-tejo
rodoviria-do-tejo: Download successful
rodoviria-do-tejo: agency Importing data
rodoviria-do-tejo: calendar_dates Importing data
rodoviria-do-tejo: calendar Importing data
rodoviria-do-tejo: fare_attributes Importing data
rodoviria-do-tejo: feed_info Importing data
rodoviria-do-tejo: frequencies Importing data
rodoviria-do-tejo: routes Importing data
rodoviria-do-tejo: stop_times Importing data
rodoviria-do-tejo: stops Importing data
rodoviria-do-tejo: trips Importing data
rodoviria-do-tejo: Post Processing data
rodoviria-do-tejo: Completed
All agencies completed (1 total)

C:\Users\PedroF\Documents\Github\node-gtfs>node ./GTFSWebSite/express/index.js
express deprecated app.configure: Check app.get('env') in an if statement GTFSWebSite\express\config.js:4:7
connect deprecated multipart: use parser (multipart, busboy, formidable) directly node_modules\express\node_modules\connect\lib\middleware\body
connect deprecated limit: Restrict request size at location of read node_modules\express\node_modules\connect\lib\middleware\multipart.js:86:15
express deprecated app.configure: Check app.get('env') in an if statement GTFSWebSite\express\config.js:14:7
express deprecated app.configure: Check app.get('env') in an if statement GTFSWebSite\express\config.js:24:7
localhost running on port 8081
127.0.0.1 - GET/api/agencies 200 14.170 ms
127.0.0.1 - GET/favicon.ico 200 1.193 ms
```

Figura 4.2: Instalação da API Node-GTFS

Após os testes locais, foi necessária a colocação *online* do sistema de resposta aos dados. Para isto, fez-se uso de uma PaaS de nome *Heroku*⁴ recomendada pelo criador da API de processamento dos dados GTFS, Brendan Nee. Estes sistemas de *Platform as a Service* caracterizam-se por conter o *software* em infraestruturas próprias devolvendo os seus serviços aos consumidores desse mesmo *software*[19].

Assim sendo, e após a criação da conta neste serviço, foi necessário efetuar o *download* dos dados do GTFS para ambiente local, verificar que tudo se encontrou bem configurado e então fazer o *deploy* dos dados para o *Heroku*, ficando estes disponíveis para todos.

Uma vez criados os dois serviços, base dados e *Heroku* é então necessário detalhar a forma como estes serviços comunicam e como é possível alterar os dados, caso

⁴<https://www.heroku.com/>

a especificação GTFS para a empresa, neste caso a Rodotejo, seja alterada. Esse detalhe pode ser visualizado na Figura 4.3, de seguida apresentada.

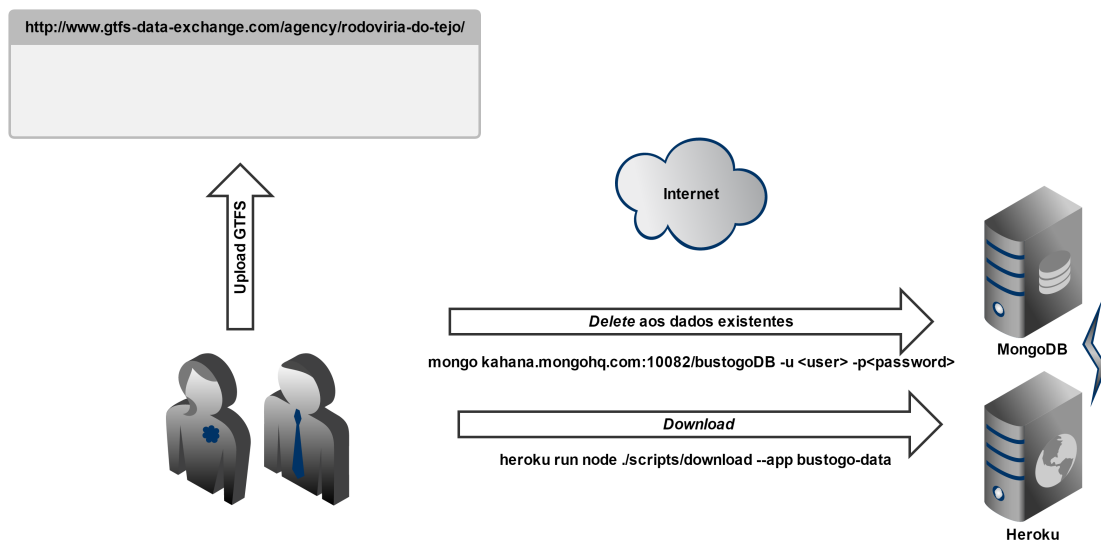


Figura 4.3: Ligação aos serviços GTFS

Esta figura descreve os passos para uma entidade de transporte tem de efetuar para partilhar a sua informação com os seus passageiros, após uma alteração. Em primeiro lugar é necessário ao funcionário da empresa de transportes submeter a referência para o *site* GTFS *DataExchange*. De seguida, é necessário correr um *script* (Anexo E) que irá apagar os dados (antigos) da base de dados e voltar a fazer *upload* dos novos, ficando estes imediatamente disponíveis para ser consumidos pela aplicação cliente.

GTFS *RealTime*

A referência *RealTime* tem como objetivo fornecer às empresas de transporte público uma forma de partilhar dados em tempo real com os seus passageiros. Como tal, e uma vez que este é um dos pontos cruciais desta arquitetura, a sua implementação torna-se importante uma vez que contribui para uma maior aproximação das empresas aos clientes. Dada a complexidade de simulação do *RealTime*, nomeadamente do sistema de posições de veículo e ainda das atualizações de viagem (uma vez dependente de mecanismos GPS para esse efeito), optou-se por simular a referência através da divulgação dos alertas de serviço fornecidos pela própria empresa de transportes. Estes alertas de serviço, tal como descritos na secção 2.3.7 - *GTFS RealTime*, servem de aproximação

da empresa aos seus passageiros pois permitem, como o próprio nome indica, alertar o passageiro dos problemas que possam ocorrer na rota ou em qualquer outro elemento que esta dependa direta ou indiretamente. Toda a criação deste sistema de alertas é conseguido através de uma API de nome *OneBusAway* para isso mesma desenvolvida. Este desenvolvimento foi levado a cabo por Brian Ferris, anteriormente referido neste documento, sendo este uma das pessoas mais importantes no que à criação desta referência *Realtime* diz respeito. De um modo geral, esta tem como funcionalidade base a divulgação, através do formato de dados JSON, dos alertas existentes numa rota para que à aplicação cliente final seja facilitada a sua interpretação e apresentação ao utilizador. Tal como é possível visualizar através da Figura 4.4, a colocação em funcionamento desta API foi conseguida com a configuração de um tempo de refrescamento na ordem dos dez segundos, tendo este parâmetro alterado e ajustado ao código da especificação existente. A este código foi também alterado o sistema de *logs*.

```
ce\GTFS RT - Alerts\OneBusAway-onebusaway-gtfs-realtime-alerts-producer-demo-a602ffe\target>cd ..
ce\GTFS RT - Alerts\OneBusAway-onebusaway-gtfs-realtime-alerts-producer-demo-a602ffe>java -jar target/onebusaway-gtfs-realtime-alerts-producer
[GTfsRealtimeAlertsProducerDemoMain.java:83] :
ts Producer Log***] - URL de origem de alertas: http://localhost:82/BusToGo/alerts/showAlerts.php

[Slf4jLog.java:67] : Logging to org.slf4j.impl.Log4jLoggerAdapter(org.mortbay.log) via org.mortbay.log.Slf4jLog
[Slf4jLog.java:67] : jetty-6.1.x
[Slf4jLog.java:67] : Started SocketConnector@0.0.0.0:8080
[GTfsRealtimeProviderImpl.java:114] : [*** OneBusAway GTFS RT Alerts Producer ***] - A iniciar o servico.

[GTfsRealtimeProviderImpl.java:115] : [*** OneBusAway GTFS RT Alerts Producer ***] - Tempo entre atualizacao de alertas: 10 (s)
[GTfsRealtimeProviderImpl.java:372] : [*** OneBusAway GTFS RT Alerts Producer ***] - Actualizando lista de alertas.
[GTfsRealtimeProviderImpl.java:331] : [*** OneBusAway GTFS RT Alerts Producer ***] - Alertas extraidos: 2
[GTfsRealtimeProviderImpl.java:372] : [*** OneBusAway GTFS RT Alerts Producer ***] - Actualizando lista de alertas.
[GTfsRealtimeProviderImpl.java:331] : [*** OneBusAway GTFS RT Alerts Producer ***] - Alertas extraidos: 2
[GTfsRealtimeProviderImpl.java:372] : [*** OneBusAway GTFS RT Alerts Producer ***] - Actualizando lista de alertas.
[GTfsRealtimeProviderImpl.java:331] : [*** OneBusAway GTFS RT Alerts Producer ***] - Alertas extraidos: 2
[GTfsRealtimeProviderImpl.java:128] : [*** OneBusAway GTFS RT Alerts Producer ***] - A parar o servico. [Tempo decorrido: 21 (s)]
```

Figura 4.4: Sistema de Alertas em Execução

Backoffice de Gestão de Alertas O sistema de gestão de alertas foi criado para as empresas de transporte adicionarem e divulgarem informação sobre problemas que possam ocorrer nas vias, sendo previamente informadas pelos condutores destes autocarros. Neste *backoffice*, visível na Figura 4.5, é possível criar e eliminar um alerta.

Figura 4.5: Criação de Alerta

Ao ser criado, este é registado numa base de dados que periodicamente será refrescada na tentativa de procurar nova informação. Caso um alerta criado e ativo não seja mais válido, poderá ser feita a sua eliminação através deste mesmo portal, deixando assim de ser divulgado para os passageiros. Uma vez registado um alerta este será exposto para ser "consumido" pela API. Para isso, foi criada uma forma de expor esses dados presentes nessa base de dados para a API lhes aceder e os divulgar, através da referência GTFS RT (Figura 4.6) (ver Anexo C), tal como a sua função o impõe.

```

http://localhost:80/alerts/debug
http://localhost:8080/alerts/debug

header {
  gtfs_realtime_version: "1.0"
  incrementaldata: FULL_DATASET
  timestamp: 1428127613
}
entity {
  id: "c51557a26bb2b8fa74eb71021d637ac7"
  alert {
    informed_entity {
      route_id: "Mobills 1"
    }
    cause: ACCIDENT
    effect: MODIFIED_SERVICE
    url {
      translation {
        text: "Localizacao: Mercado Municipal;DataInicio: 2014-10-21;HoraInicio: 00:00;DataFim: 2014-11-10;HoraFim: 12:15"
      }
    }
    header_text {
      translation {
        text: "Estrada Cortada"
      }
    }
    description_text {
      translation {
        text: "Devido a reparavimento da via, a rota do Mobills 1 ficar\303\203\302\241 interdita. Aconselhamos os passageiros a deslocarem-se para as paragens imediatamente antes ou depois desta paragem do Mercado, nesta rota. Pedimos desculpa pelo incmodo causado."
      }
    }
  }
}
entity {
  id: "0bce7e73a9091a28474617aedc0e7a97"
  alert {
    informed_entity {
      route_id: "NOME DA ROTA"
    }
    cause: TECHNICAL_PROBLEM
    effect: SIGNIFICANT_DELAYS
    url {
      translation {
        text: "Localizacao: LOCALIZACAO ALERTA;DataInicio: 2014-11-29;HoraInicio: 12:12;DataFim: 2014-11-29;HoraFim: 14:14"
      }
    }
    header_text {
      translation {
        text: "TITULO DO ALERTA"
      }
    }
    description_text {
      translation {
        text: "DESCRICAO DO ALERTA"
      }
    }
  }
}

```

Figura 4.6: Alerta em formato GTFS RT

Fluxo de comunicação de alertas A forma de criação de um alerta desde a sua criação, a publicação do mesmo e à sua disponibilização à aplicação cliente final é ilustrada na Figura 4.7 que de seguida se apresenta.

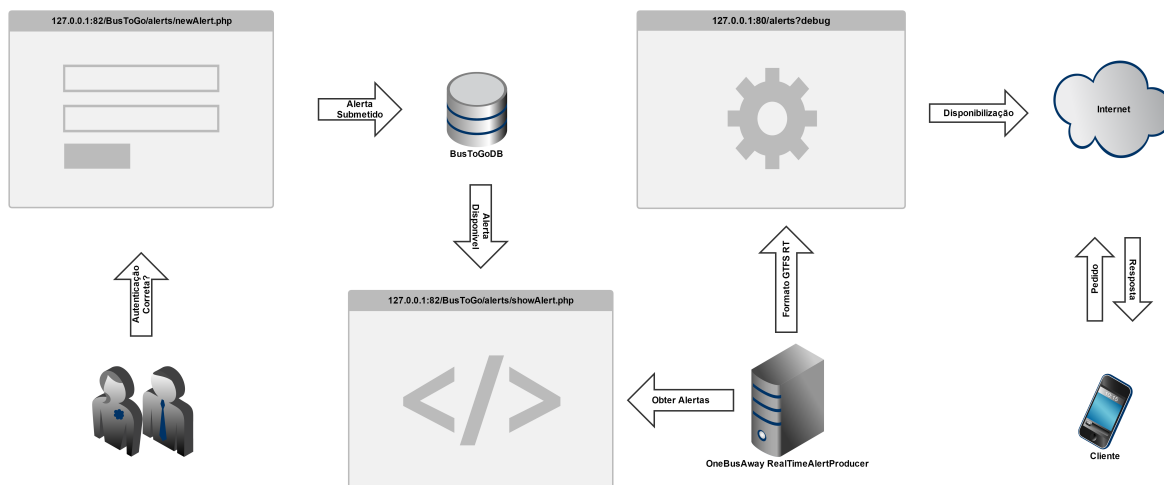


Figura 4.7: Fluxo de criação de um alerta

O operador da companhia de transportes tem informação acerca do alerta e, de forma a alertar os seus passageiros, irá ao *backoffice* registar o mesmo. Após este registo, este

será gravado numa base de dados local e ficará disponível numa página que apenas a API terá conhecimento. Uma vez aqui presente, a API na tentativa de refrescar os seus dados, irá efetuar um pedido a esta página e transformará o que aqui existir em formato *Realtime* ficando assim disponível para o cliente consultar.

4.1.2 Cliente

Android

O cliente móvel desta solução foi pensado para *smartphones* sendo também compatível com *tablets*, ambos Android. A utilização desta plataforma deve-se em muito, e como já abordado anteriormente neste documento, aos números da sua utilização não só em Portugal [15][20], como também, em todo o mundo [21], questões de licenças ou até mesmo partilha de código *online*.

O Android, lançado pela *Open Handset Alliance*⁵ nos finais de 2007 é, de longe, o sistema operativo móvel mais utilizado hoje em dia, sendo também o que mais dispositivos vende, muito devido ao facto de ser mais acessível financeiramente em relação aos seus "rivais" *Windows Phone*, da Microsoft e *iPhone*, da Apple.[22].

SQLite

Sendo o SQLite⁶ uma base de dados *open source* e uma vez que vem incorporada nos dispositivos Android, decidi utilizar-se este tipo como forma de armazenar a informação. O SQLite suporta todas as funcionalidades do SQL, desde a sintaxe, às transações bem como os *prepared statements*. Além disso, este tipo de base de dados suporta tipos de dados semelhantes ao Java, tais como TEXT que atua como *String*, INTEGER, semelhante ao *Long* e ainda o REAL que em tudo se relaciona com o *Double* nas linguagens de programação. Como tal, e independentemente do tipo, todos os dados devem ser convertidos num dos tipos referidos, uma vez que esta base de dados não valida se o que é escrito nas colunas da tabela é do mesmo tipo do que o dos dados associado à mesma.

⁵Grupo de 84 empresas de tecnologia e comunicações que tem como objetivos inovar a tecnologia móvel, melhorar a experiência de utilização deste tipo de dispositivos e a disponibilização dos mesmos a preços mais acessíveis. [adaptado de <http://www.openhandsetalliance.com/>]

⁶<http://www.sqlite.org/>

SQLite e o Android: Como antes referido, este tipo de base de dados vem integrada em todos os dispositivos Android. Posto isto, usar SQLite não pressupõe um procedimento de configuração ou administração da base de dados sendo que apenas precisa que sejam escritas as *queries* para criar ou fazer atualização à base de dados. Após a definição destas *queries*, o sistema operativo faz a gestão automática e de forma dinâmica os dados da base de dados.

ADT - Android Development Tools

Embora seja constante o crescimento da utilização *Android Studio* face ao ADT, para esta solução foi utilizado o *Android Development Tools* dada não só a sua simplicidade, como também, o facto de ser já familiar a sua utilização, tornando assim os desenvolvimentos não tão morosos. Este ADT não é mais do que um *plugin* para o IDE de desenvolvimento Eclipse que agrega e estende as funcionalidades já disponibilizadas por este. Com este *plugin* é possível a criação de projetos/aplicações Android, interagir com o SDK, efetuar *debug* às aplicações ou ainda gerar e disponibilizar o *.apk* da aplicação em questão, entre outras funcionalidades.

ActionBarSherlock Side Menu Navigation Drawer

De modo a garantir a compatibilidade com as versões de Android mais antigas, optou-se por utilizar a biblioteca de Jake Wharton⁷ de nome *ActionBarSherlock*. Esta biblioteca tem como objetivo a introdução da *action bar* ou barra de ação, em português, nas versões anteriores à *Ice Cream Sandwich*, isto é, a 4.0 do sistema Android. Da mesma forma foi disponibilizada uma barra lateral esquerda, conhecida como *Navigation Drawer*. Esta barra, ao permitir o acesso a todas as funcionalidades a partir de um ecrã torna-se fundamental para o tipo de aplicação em questão, tendo em conta também a forma e as ocasiões em que será utilizada, isto é, na maioria das vezes, em movimento.

Google API's

Uma aplicação cliente que consuma serviços de mapas requer, por norma, a utilização de mecanismos que permitam transcrever a informação codificada em coordenadas GPS, em informação visível e perceptível ao utilizador da mesma, neste caso, ao

⁷<http://actionbarsherlock.com/>

utilizador da aplicação e, conseqüentemente, passageiro do autocarro. Como tal, e não sendo exceção, está a aplicação cliente desenvolvida. Esta foi desenvolvida assentando algumas das suas funcionalidades em API's fornecidas pela Google. As referidas API's definem-se por permitirem a comunicação com serviços da gigante norte americana. Estes serviços são, entre outros, o Google Search, o Google Mail, o Google Maps e o Google Places.

Para que a sua integração seja possível, é necessária a autenticação da aplicação perante a entidade de *developers* da Google. Como tal, e como se pode verificar na Figura 4.8, para a aplicação implementada foi necessária a criação dessa mesma autenticação que, através de uma chave no formato MD5, disponibiliza o acesso aos serviços atrás referidos e que, neste âmbito, foram o Maps e o Places em que o primeiro disponibiliza o serviço de mapas e o segundo serve ao utilizador os locais mais próximos deste.

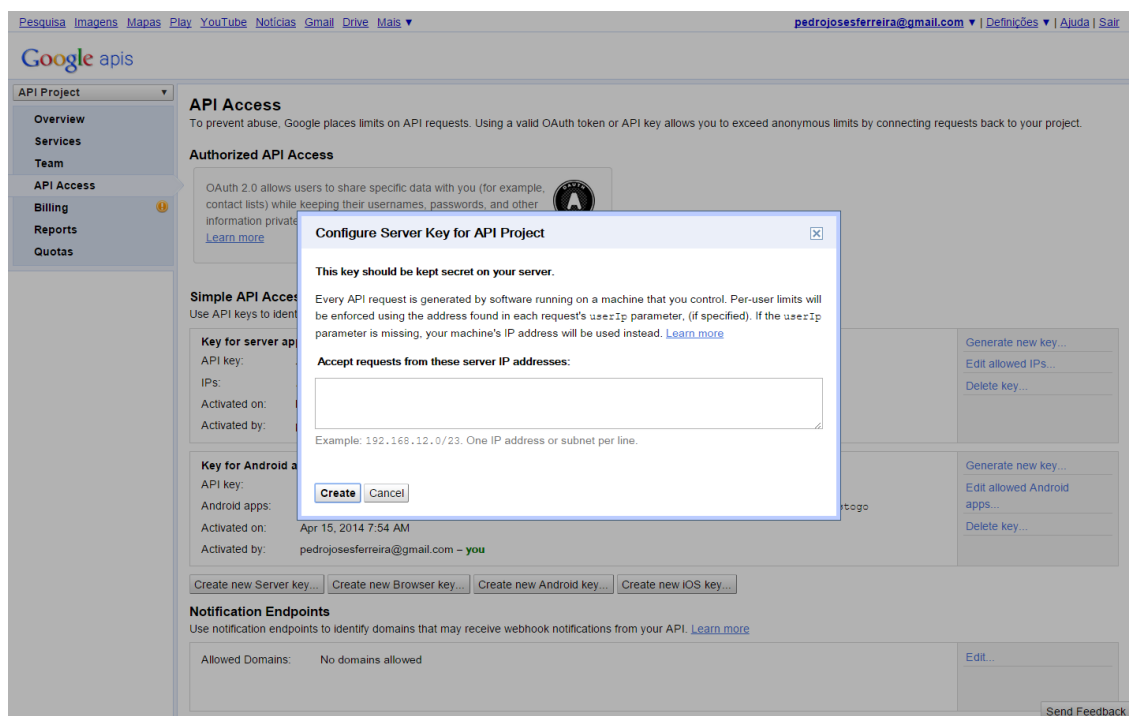


Figura 4.8: Criação de uma ligação às Google API's

4.2 Metodologia de Desenvolvimento

São várias as metodologias de desenvolvimento existentes em Engenharia de *Software*. Desde o Modelo Cascata (também conhecido por Linear ou Clássico), passando pelo

Iterativo e Incremental, não esquecendo a Metodologia Ágil, o *Extreme Programming* e o *Scrum*, entre outras, pode dizer-se que a fase de desenvolvimento desta aplicação foi iniciada com a definição de requisitos não fechada na totalidade, porém, bem definida. Alguns dos requisitos foram sendo alterados com o desenrolar da fase de desenvolvimento, daí que, se existir uma metodologia associada ao tipo de desenvolvimento da aplicação em questão, será o desenvolvimento Iterativo e Incremental. Esta metodologia de desenvolvimento surgiu com o objetivo de dar resposta ao Modelo Cascata que assentava na ideia de que um desenvolvimento era sequencial sem a necessidade de alteração de requisitos. Assim, o Modelo Iterativo e Incremental difere da abordagem clássica uma vez que permite a alteração de requisitos *on the job*, repartindo assim o desenvolvimento em fases que podem ser divididas por análise, implementação e testes, executadas mais do que uma vez se necessário.

4.2.1 MVC

Muito utilizado em Engenharia de *Software*, o MVC é um padrão de desenvolvimento de *software* largamente utilizado hoje em dia. Este modelo tem a capacidade de separar o modelo de dados (*model*) do ecrã com que o utilizador interage (*view*) sendo o controlador o responsável por tomar as ações do utilizador no ecrã, efetuar alterações ao modelo e devolver essas alterações ao ecrã tecnicamente conhecido por *view*.

Embora gerando alguma controvérsia, a implementação de aplicações Android não assenta diretamente na utilização de padrões MVC, no entanto, são fornecidos ao utilizador várias formas de aproximação a este padrão que se caracterizam por ser:

- Definição da interface com o utilizador em vários ficheiros XML;
- Criação de ficheiros de linguagens com base em *resources*;
- Utilização de *inflaters* para estender classes como *ListActivity* ou *TabActivity*;
- Possibilidade de criação de quantas classes de negócio forem necessárias;
- A utilização de utilitários⁸ previamente criados como *DatabaseUtils*, *JsonWriters*, entre outros.

Como referido anteriormente, a comparação de MVC com MVP, em desenvolvimento Android, tem gerado alguma controvérsia pois surgem as dúvidas quanto à classe *Activity* pois esta não estende de nenhuma *View*, no entanto, disponibiliza métodos de

⁸<http://developer.android.com/reference/android/util/package-summary.html>

interação com a mesma como por exemplo *OnCreate*, *OnDestroy*, *OnResume*, entre outros. Dito isto, aqueles que acham que o MVP é o modelo ideal de desenvolvimento para esta plataforma, afirmam-no uma vez que as *views* servem de entrada à aplicação, efetuam o *rendering* dos componentes e comunicam com o *presenter*, que gere a comunicação entre o modelo e a vista.

Na aplicação cliente em questão e seguindo este paradigma, foram definidas várias *packages* de aproximação ao modelo e são elas:

- **pt.estg.ipleiria.meicm.bustogo** - Os *fragments* e *activities* associados à aplicação, isto é, o *presenter*;
- **pt.estg.ipleiria.meicm.exception** - Exceções criadas propositadamente para a aplicação;
- **pt.estg.ipleiria.meicm.others** - Utilitários como detetores de ligação existente, ficheiros de constantes, ferramentas de GPS, entre outros. ;
- **pt.estg.ipleiria.meicm.services** - Serviços de obtenção de alertas e interação com o *Google Places*;
- **pt.estg.ipleiria.meicm.parser** - Descodificação das entradas XML e JSON;
- **pt.ipleiria.meicm.database** - Interação com a base de dados, *queries* de consulta, *update*, *delete*;
- **pt.ipleiria.meicm.object** - Definição do modelo, isto é, dos objetos necessários pelo sistema.

Para além destes, existem ainda, como pode ser visível no Anexo F, o desenho dos ecrãs (*/res/layout*), os menus (*/res/menu*), as *strings* (*/res/values*) e ainda o ficheiro xml das preferências (*/res/xml*).

4.2.2 Modelo de Dados

A modelação de dados de um sistema de base de dados, independentemente do sistema, é importante pois define um conjunto de conceitos para representação dos dados. O modelo de base de dados para a aplicação cliente encontra-se na Figura 4.9, de seguida apresentada.

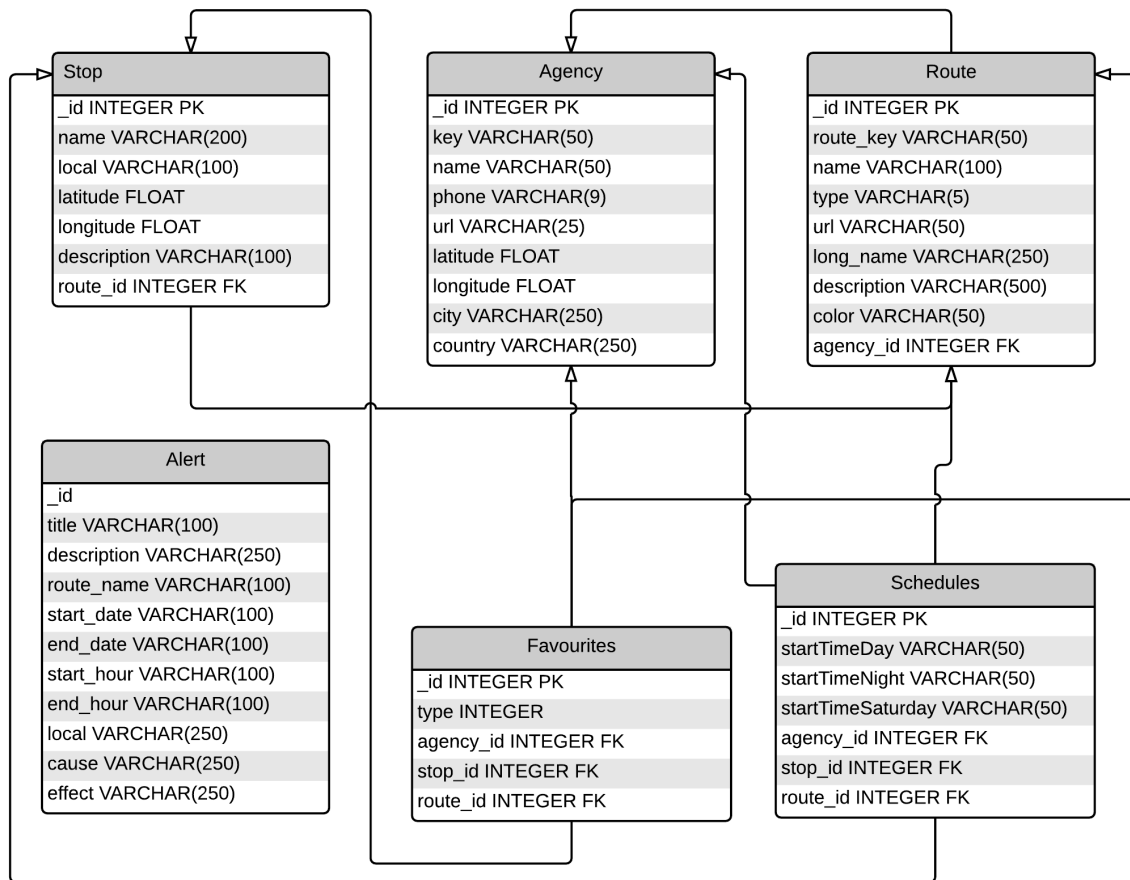


Figura 4.9: Modelo de Dados

Neste, foram criadas seis tabelas de dados, *stop* que contém todas as paragens, *route* que é responsável por registar as rotas existentes, *favourite* em que serão introduzidos ou removidos os favoritos (rotas, paragens ou entidades de transportes) escolhidos pelo utilizador da aplicação. Foi ainda criada uma tabela que regista os dados dos alertas existentes (*alerts*), os horários das paragens (*schedules*) e ainda os dados da empresa de transportes (*agency*).

4.3 Conclusão

A implementação de uma solução que tem como objetivo dar resposta a um problema existente, independentemente da arquitetura, requisitos e plataformas selecionadas, carece sempre de uma análise cuidada das melhores tecnologias. Como tal, e dadas as circunstâncias atuais, a opção por tecnologias de confiança e, sobretudo, gratuitas é

fundamental. Por isso mesmo, a escolha para este sistema foi, para a aplicação móvel cliente, o sistema operativo Android e para a divulgação dos dados por parte das empresas, a conjugação de duas ferramentas cuja utilização tem vindo a aumentar nos últimos tempos, o OpenStreetMap e ainda o GTFS, da Google. Assim sendo e uma vez definidos os requisitos e a arquitetura é necessário então dar início à implementação do sistema seguindo para isso paradigmas utilizados em Engenharia de *Software*, como é caso da metodologia iterativa e incremental.

Capítulo 5

Solução Final

Terminada a implementação da aplicação com base na abordagem proposta é necessário então validar o modo real de funcionamento deste sistema através da conjugação das duas entidades participantes nesta arquitetura, nomeadamente, o cliente e o servidor. Assim sendo, ao longo deste capítulo será descrito o modo de funcionamento da aplicação cliente e as opções tomadas para cada uma das suas funcionalidades. Nesta secção será ainda apresentado e detalhado o resultado dos testes efetuados à aplicação.

5.1 Cliente

Fazendo uma breve referência aos protótipos desenhados e que podem ser visualizados no Anexo B, pode afirmar-se que de acordo com as perspetivas iniciais e em comparação ao resultado obtido, não houve uma diferença tão acentuada. Isto significa que não só o desenho da *interface* acompanhou, de forma geral, os requisitos definidos como também os objetivos propostos foram, maioritariamente, alcançados.

5.1.1 Menu Principal

O Menu Principal define o ponto inicial de interação do utilizador com a aplicação. Este (Figura 5.1) foi desenhado e implementado com o objetivo de garantir uma usabilidade superior dado que os utilizadores fazem uso de uma aplicação deste género em movimento. Este aumento de usabilidade deve-se ao facto deste ser acessível a partir de

qualquer ponto da aplicação, evitando assim o utilizador de sair do contexto atual da sua ação perante a aplicação para consultar todas as outras opções existentes.

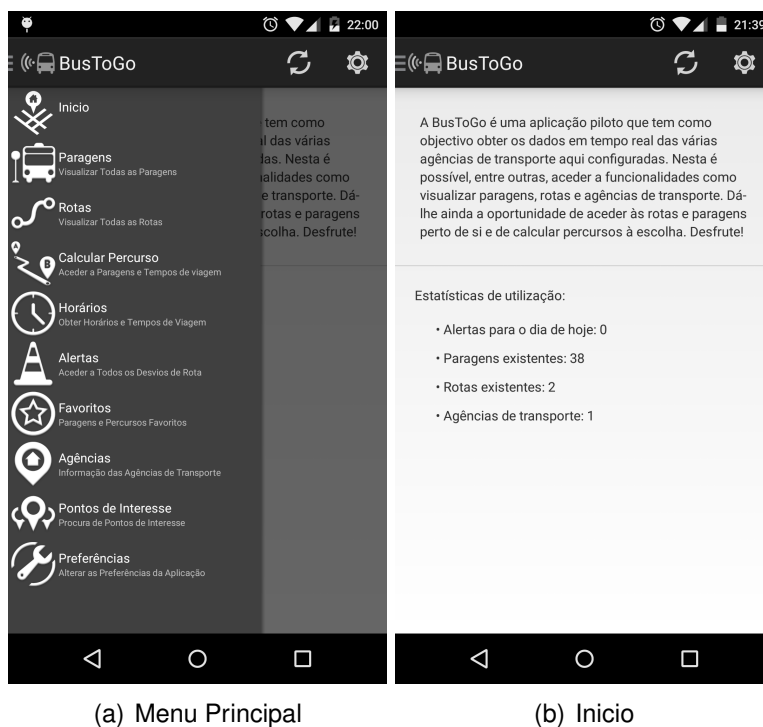


Figura 5.1: Menu Principal e Inicio

Para além deste menu principal, existe ainda uma barra superior que garante mais opções, como o atualizar e o acesso às definições, neste ecrã.

5.1.2 Inicio

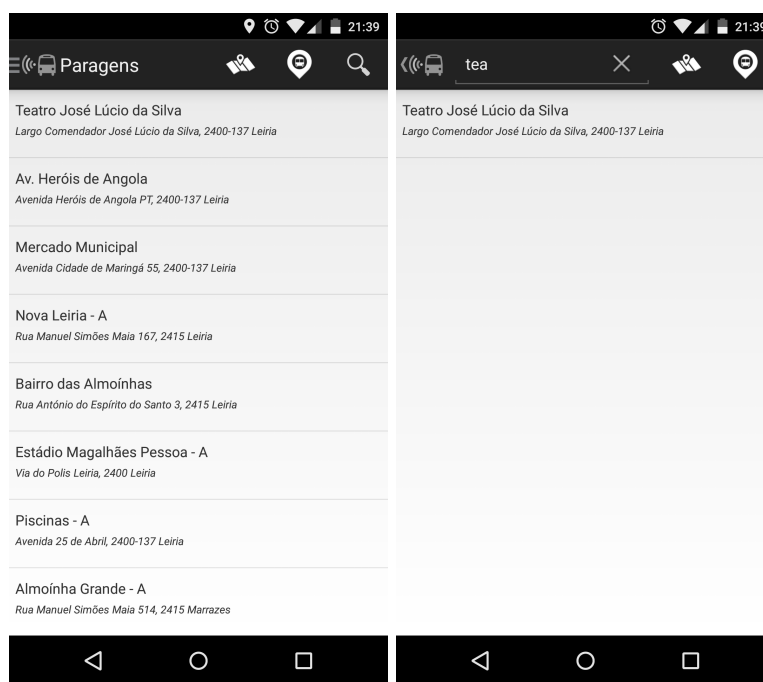
O ecrã de Inicio é o primeiro contato do utilizador com a aplicação. Nesta fase, como visível na figura (Figura 5.1), existe definida uma referência à aplicação como piloto em si. Para além desse texto introdutório, existem ainda estatísticas de dados existentes na mesma, isto é, o número de paragens, número de alertas no momento, o número de rotas e o número de empresas de transporte. Esta informação apenas existe neste ecrã para evitar uma entrada na aplicação diretamente para ecrãs como paragens, rotas ou outro, daí que se tenha decidido a criação deste ecrã de inicio. Numa fase posterior

o texto de referência será substituído por gráficos que traduzirão essas estatísticas em dados visuais mais apelativos.

5.1.3 Paragens

O acesso às paragens é uma das funcionalidades mais importantes numa aplicação deste género. Como tal, foi dado destaque às mesmas tanto no seu posicionamento no menu, como também, nas opções disponíveis para a interação com estas.

Ao ser invocada a opção de consulta de paragens, é automaticamente feita a seleção dos dados destas que se encontram presentes na base de dados. Aí, é mostrado ao utilizador esta lista de paragens, acompanhada da morada das mesmas. A inclusão da morada tem como objetivo tornar mais perceptível ao utilizador de qual é efetivamente a paragem pois nem sempre o nome desta é o mais elucidativo.



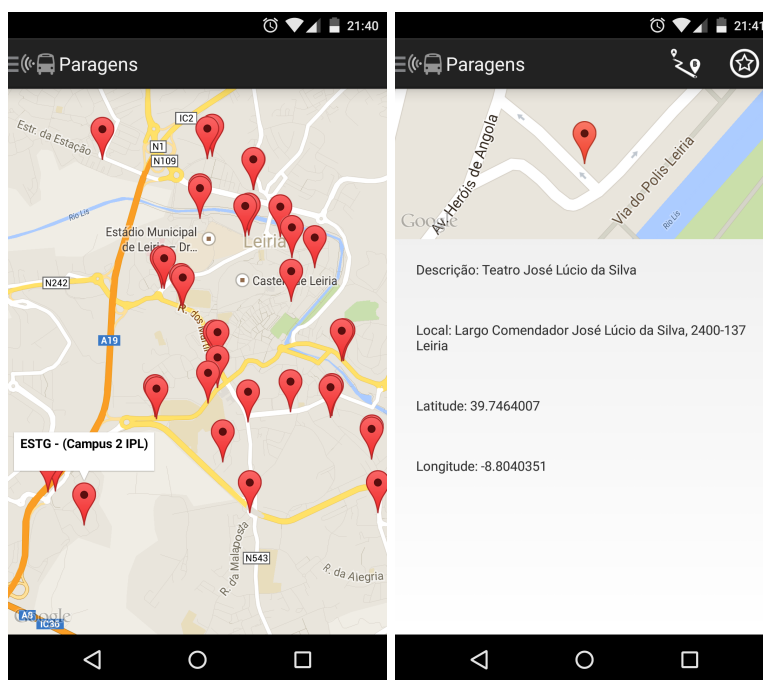
(a) Lista de Paragens

(b) Procura de Paragens

Figura 5.2: Paragens I

Para além disto, neste ecrã, existem três outras opções. A primeira, visível na Figura 5.2, é a procura de paragens que permite ao utilizador introduzir o nome ou local total ou parcial de uma mesma paragem para que o sistema filtre as mesmas e facilite a sua procura. Outras das opções existentes é a listagem das paragens no mapa, caso

o utilizador pretenda. Esta listagem é o enquadramento, em termos visuais, da sua localização no mapa. Porém, e antes de aprofundar o ecrã de detalhe de uma paragem (Figura 5.3), é ainda importante dar destaque a outra das possibilidades que tem objetivo a procura de paragens perto do utilizador, sendo que para isso, este tenha que ter o acesso à localização ativado e ainda, caso pretenda, seleccionar um raio de procura. Para finalizar, existe ainda o ecrã de detalhe das paragens que apresenta não só a localização da paragem no mapa, como também, os detalhes desta que são os mesmos que o ficheiro da referência GTFS alberga, nomeadamente, a descrição ou nome da paragem, a sua localização textual e ainda os seus parâmetros de localização GPS, a latitude e a longitude.



(a) Lista de Paragens no Mapa

(b) Detalhe de Paragens

Figura 5.3: Paragens II

5.1.4 Rotas

As rotas definem uma sequência de paragens até ao seu início. Neste caso, as rotas são listadas por nome com a informação da empresa de transportes a que pertencem (Figura 5.4). Neste ecrã é dada a possibilidade de se procurarem por nome total ou parcial, tal como nas paragens. Seleccionando uma rota na lista é então apresentado

ao utilizador os detalhes desta que são constituídos pela lista de paragens que compõe essa rota bem como o tipo de rota e ainda a empresa de transporte a que pertencem. No ecrã de detalhe é ainda dada a possibilidade ao utilizador de adicionar a rota selecionada aos favoritos.

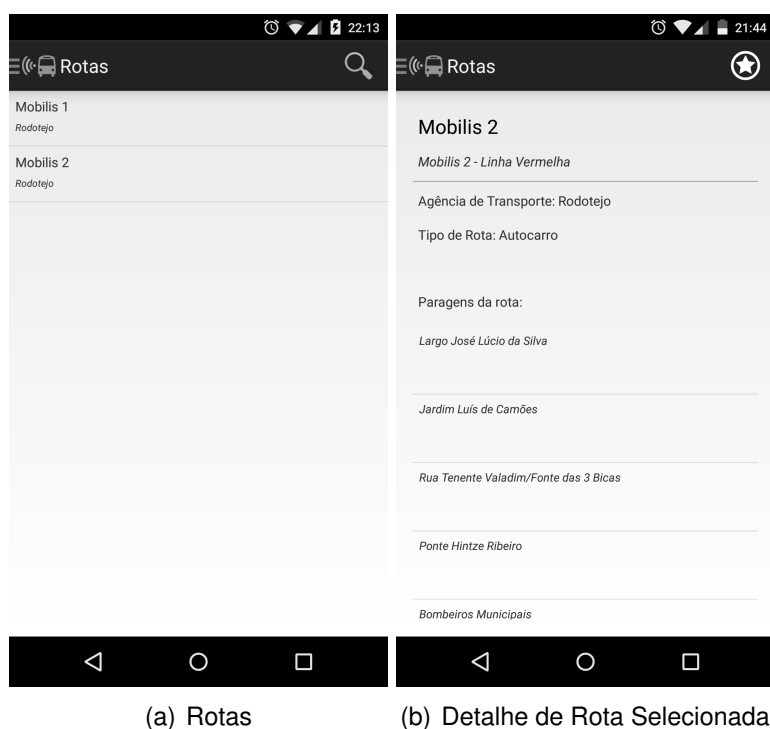
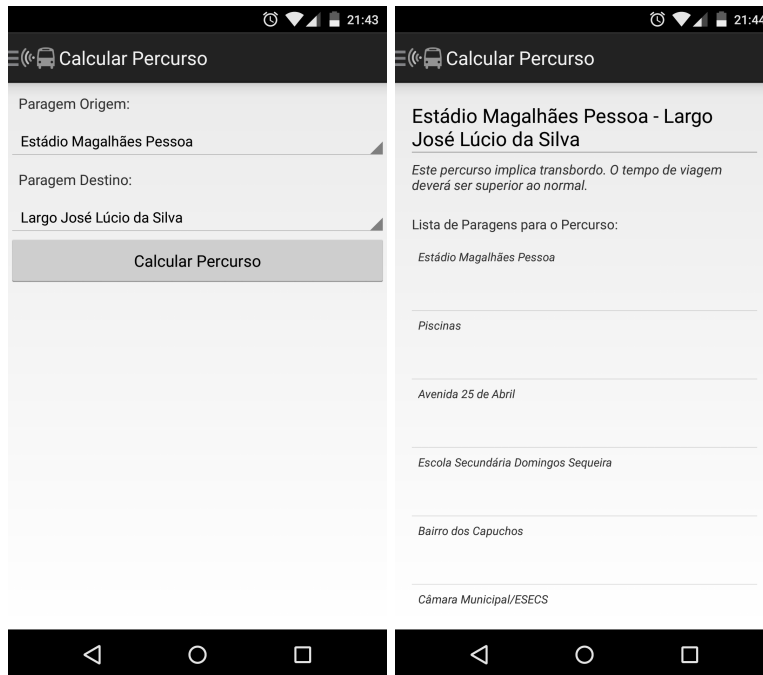


Figura 5.4: Rotas

5.1.5 Calcular Percurso

O calculador de percurso (Figura 5.5) subentende a escolha de um percurso origem e um percurso destino. Esta escolha feita pelo utilizador da aplicação tem o objetivo de dar as paragens intermédias até ao final desse percurso.



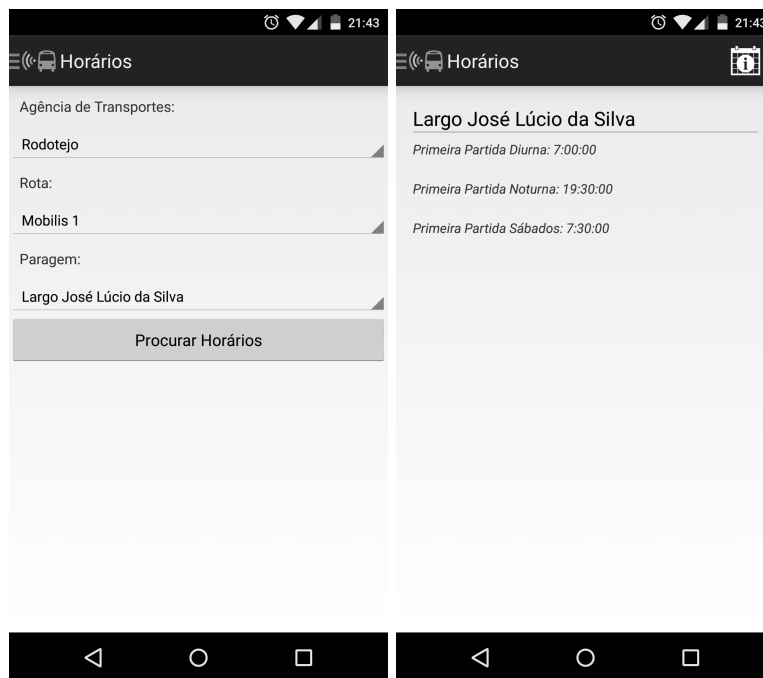
(a) Calcular Percurso

(b) Detalhe de Percurso Calculado

Figura 5.5: Cálculo de Percursos

5.1.6 Horários

Para um utilizador de uma aplicação neste âmbito é necessário também o acesso a informações horárias. Posto isto e limitados pela referência GTFS são apresentados os horários iniciais para cada turno em cada paragem (Figura 5.6) após a seleção de um conjunto de parâmetros. Os parâmetros que conjugados darão acesso aos horários são: a empresa de transportes (pois podem existir mais do que uma), a rota (dependente da empresa de transportes selecionada) e a paragem (também relacionada com a rota selecionada).



(a) Pesquisa de Horário por Paragem (b) Detalhe de Pesquisa de Horário

Figura 5.6: Horários

5.1.7 Alertas

Os alertas foram uma das grandes novidades em aplicações deste género, neste âmbito, devido à adaptação da referência *RealTime*. Assim sendo, foi criado um serviço que periodicamente efetua a procura de novos alertas e os revela ao utilizador. Quando encontrado um, é apresentado ao utilizador na barra de notificações nativa do sistema operativo Android uma referência à existência de novos alertas, sendo estes incluídos numa listagem que pode ser vista na Figura 5.7 e que após uma seleção na lista transpõe os detalhes desse mesmo alerta, que apresenta a hora início e de previsto fim desse alerta, o local, a causa e efeito e ainda a descrição do mesmo.

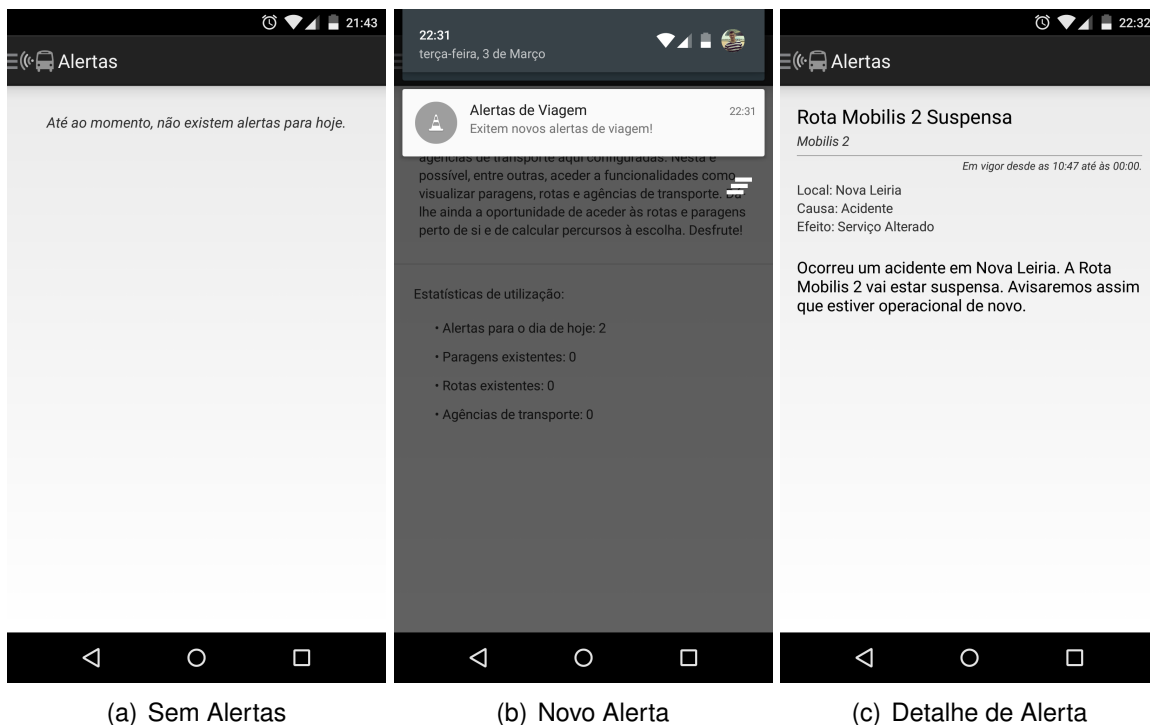


Figura 5.7: Alertas

5.1.8 Favoritos

Os favoritos são uma constante nas aplicações analisadas aquando do início deste estudo. Como tal, optou-se por seguir o mesmo caminho traçado por estas mesmas aplicações para permitirem ao utilizador a adição de rotas, paragens e empresas de transporte a um ecrã de fácil e rápido acesso. Esse ecrã de favoritos, como na Figura 5.8 se pode observar, indica essas mesmas entidades adicionadas aos favoritos podendo ser "escondidas" através da seleção das divisórias existentes. Selecionando também um elemento dos favoritos o utilizador é levado ao detalhe desse mesmo elemento quer ele seja uma rota, uma paragem ou uma empresa de transporte, também nesta aplicação denominada por agência de transporte.

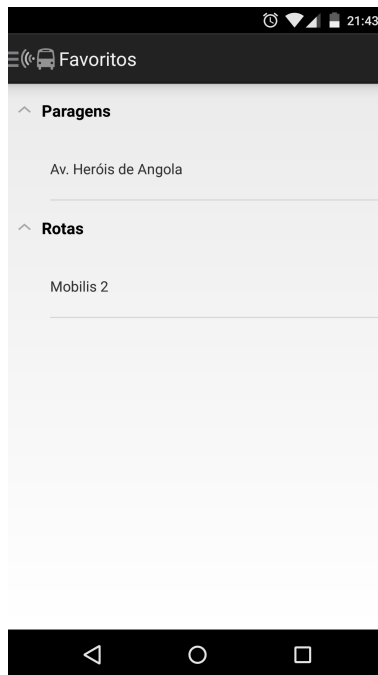


Figura 5.8: Favoritos

5.1.9 Empresas

As empresas de transporte podem ser listadas no ecrã presente na Figura 5.9. Aqui serão visíveis todas as empresas presentes na referência GTFS e os detalhes das mesmas. Esse detalhe incluem a localização no mapa, o telefone e o sítio na Internet. É dada a possibilidade ao utilizador de aceder ao sítio na Internet dessa mesma empresa, efetuar uma chamada para esta ou ainda adicioná-la aos favoritos.

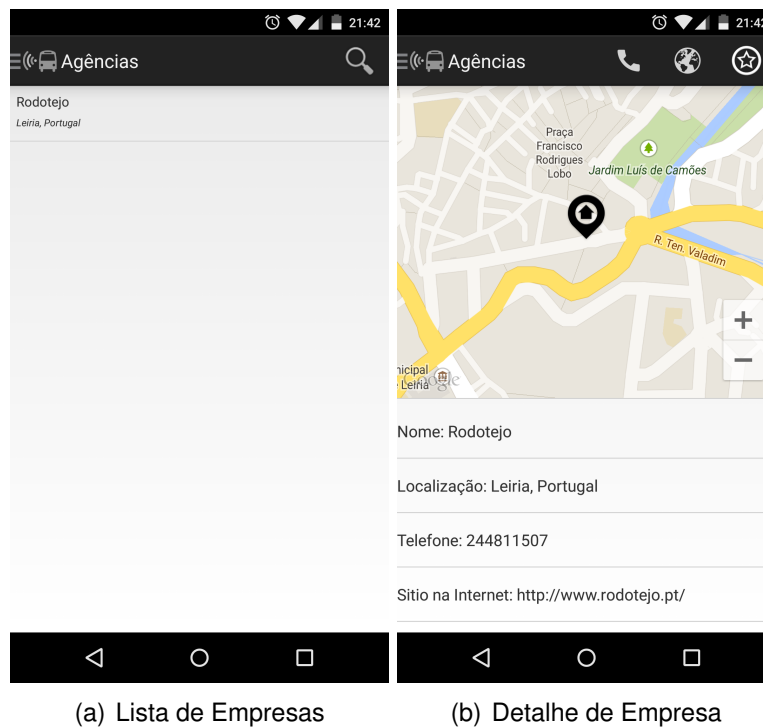


Figura 5.9: Empresas

5.1.10 Pontos de Interesse

Os pontos de interesse (Figura 5.10) foi uma funcionalidade encontrada em algumas das aplicações analisadas. Posto isto e através da integração de serviços da Google, conseguiu-se orientar a aplicação não só aos habitantes da cidade, como também àqueles que não a conhecem e pretendem deslocar-se a sítios de interesse e que, consequentemente, fazem-no de transportes públicos. Estes pontos de interesse são procurados num raio de cinco quilómetros.

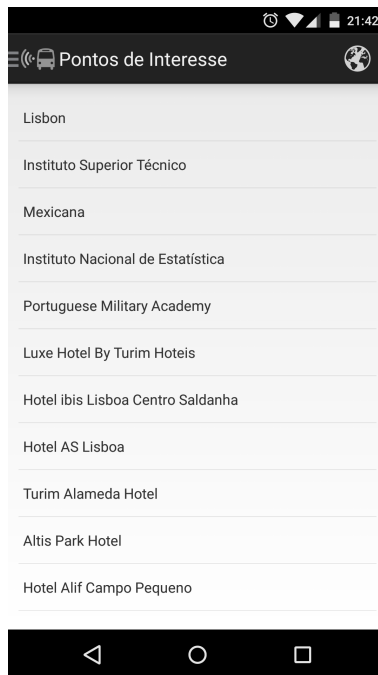


Figura 5.10: Pontos de Interesse

5.1.11 Preferências

Em todas as aplicações deste ou de outro âmbito existe uma possibilidade de editar definições da aplicação a que se dá o nome de preferências. Neste caso, é possível ao utilizador seleccionar se pretende actualizações no arranque da aplicação e o tipo de ligação utilizada para efetuar o *download* dos dados (3G/4G, Wifi, ou 3G/4G e Wifi). É-lhe ainda dada a possibilidade de configurar a procura de alertas de serviço e tempo entre o qual o deve fazer. Para além da sugestão de melhorias (que envia o utilizador diretamente para a aplicação do *Google Mail*), este pode ainda limpar os dados todos existentes na aplicação, como se pode verificar na (Figura 5.11).

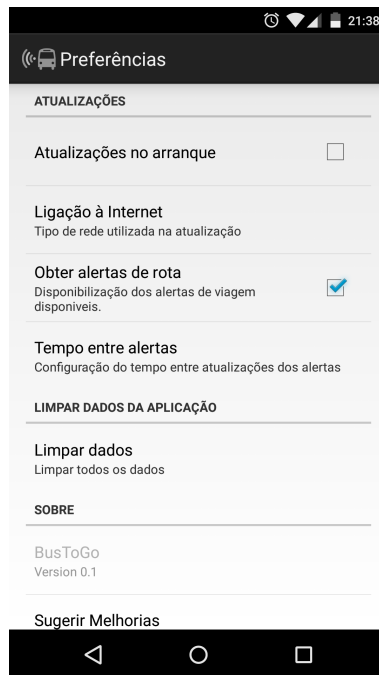


Figura 5.11: Preferências

5.2 Testes

Os testes aplicativos são um passo importante após o estudo e implementação de qualquer sistema pois são estes que garantem a aceitação e o sucesso do mesmo. Estes testes dividem-se em três categorias, funcionalidades, conectividade e carga. Os primeiros são os mais importantes pois é onde são analisadas por utilizadores que desconhecem o sistema, a forma de funcionamento do mesmo e que apuram se estas estão bem pensadas em relação aos objetivos a que se propõem. Por outro lado, tanto os testes de conectividade como os de carga, já não são controlados por um interveniente uma vez que se tratam de funcionalidades e detalhes do próprio sistema.

5.2.1 Funcionalidades

Os testes às funcionalidades foram, para além dos unitários que auxiliaram na correção de problemas e na alteração de questões de usabilidade, escassos tendo em conta o número de possíveis utilizadores que se enquadram neste âmbito. No entanto, os testes foram realizados por cinco pessoas, duas do sexo feminino e três do sexo masculino

com idades compreendidas entre os vinte e cinco e os trinta e oito todas possuidoras de um dispositivo móvel.

Nestes testes, as tarefas pretendidas foram:

- 1) Adicionar a rota "Mobilis 2" aos favoritos;
- 2) Configurar a ligação à Internet utilizada pela aplicação para utilização apenas de *Wifi*;
- 3) Procurar a paragem "Estádio Magalhães Pessoa" e adicioná-la aos favoritos;
- 4) Calcular um percurso com origem em "Estádio Magalhães Pessoa" e fim em "Rotunda ESTG";
- 5) Obter o horário de início da última partida disponível na paragem "Piscinas" da Rota "Mobilis 1";
- 6) Aceder às rotas favoritas e remover a rota "Mobilis 2" dos favoritos;
- 7) Procurar um ponto de interesse;
- 8) Visualizar todas as paragens no mapa.

A não definição de um guião de testes para os utilizadores seguirem não se revelou um problema. No entanto, após um enquadramento do que se pretendia e apresentada a listagem de tarefas propostas, deram-se início aos testes. As tarefas foram selecionadas com o intuito de abordar o maior número de funcionalidades tendo em atenção à morosidade de todo este processo do ponto de vista do utilizador. Assim, a primeira tarefa consistia na adição de uma das duas rotas presentes na aplicação aos favoritos sendo esperado que o utilizador interagisse com os menus das rotas e dos favoritos (de forma a confirmar o sucesso desta adição). De seguida, era pretendido que fosse alterada forma de ligação à Internet através das preferências, quer pelo menu lateral, quer pela página de início. A terceira tarefa consistia na interação tanto com as paragens como com os favoritos, novamente. Aqui era expectável que o utilizador fizesse uso do menu lateral das paragens e a procurasse quer através da sua seleção na lista, quer pela opção de pesquisa por nome. Ainda sobre as paragens, a tarefa seguinte pressupõe o cálculo de um percurso com duas paragens, neste caso, a "Estádio Magalhães Pessoa" como origem e a "Rotunda ESTG" como paragem destino.

A tarefa cinco era das mais complexas existentes nesta aplicação dado que o número de passos a tomar para chegar ao objetivo eram maiores em comparação com todas as outras tarefas. Nesta, o utilizador teria que aceder à opção "Paragens", selecionar a rota, a paragem e a empresa de transportes públicos e procurar pelo horário presente

e indicado na tarefa. De seguida, e após ter adicionado anteriormente a rota "Mobilis 2" era pedido ao utilizador a remoção da mesma. A próxima tarefa e antes de terminar era o utilizador procurar um ponto de interesse por perto sendo que para isso bastava seleccionar a opção "Pontos de Interesse" presente no menu lateral. Por último, a tarefa oitenta pretendia que o utilizador procurasse e visualizasse todas as paragens no mapa.

Resultados

No geral, o entusiasmo em torno da aplicação revelou-se interessante tendo em conta o resultado. Para além dos positivos comentários acerca da localização das opções na barra da esquerda, esta tornou-se num elemento de destaque pois até à data a interação dos utilizadores com esta forma de ter presente o "menu" era nula.

No entanto, e no que toca aos testes realizados, não houve dificuldades de maior nas tarefas propostas, para além da tarefa de aceder ao horário de início da última rota disponível na paragem "Piscinas" e ainda a visualização das paragens no mapa. Em ambas as tarefas surgiram erros repetidos por mais do que um utilizador, como se pode verificar pela Figura 5.12. Na tarefa de aceder ao horário das paragens, alguns utilizadores dirigiram-se em primeiro lugar às paragens em busca da paragem "Piscinas" e tentaram aceder aos horários a partir desta, deixando a recomendação de uma melhoria. Esta situação não era, de todo, expectável uma vez que na barra lateral existe uma opção própria de "Horários". Outra das situações não comuns foi o facto da confusão em torno da visualização das paragens no mapa. Aqui, o utilizador teria que aceder às paragens no menu lateral e seleccionar a opção de paragens no mapa. A confusão em torno desta funcionalidade deveu-se sobretudo à semelhança de ícones entre esta funcionalidade e a de visualizar as paragens por perto.

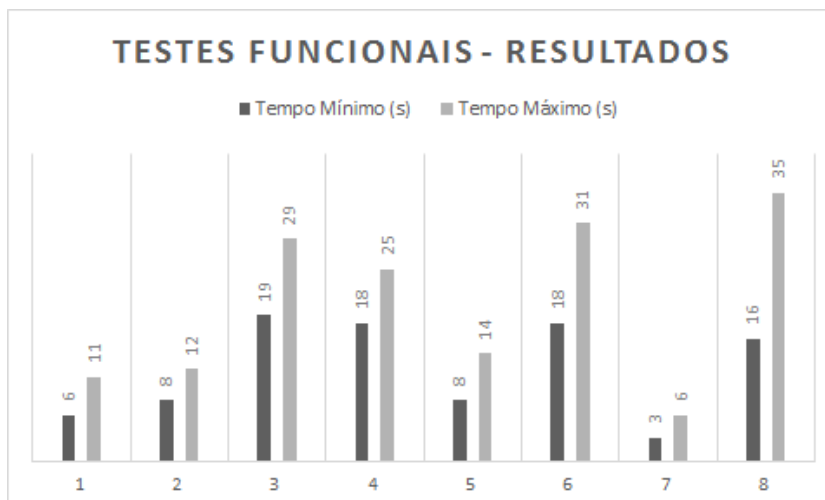


Figura 5.12: Testes Funcionais - Resultados

Como se sabe, a execução de testes com utilizadores é sempre uma mais valia, não só porque estes pensam de forma diferente de quem desenvolve, como também porque descobrem falhas existentes. A aplicação cliente apresentada não foi exceção e ao longo dos testes teve falhas pontuais ao nível de parar de forma repentina a interação do utilizador com esta. Apesar destas falhas (críticas em alguns pontos), surgiram sugestões por parte dos utilizadores que, após os testes, voltaram a pedir para ver e experimentar a aplicação que lhes tinha sido apresentada. Daqui surgiram ideias que serão implementadas futuramente como é o caso das conversões dos dados de início em estatísticas apresentadas graficamente, de forma a tornar mais apelativa a entrada na aplicação. Para além disto, foi sugerida a inclusão, por ordem alfabética das paragens nos *spinners* de escolha. No cálculo do percurso, alguns dos utilizadores esperavam ver o tempo estimado para o mesmo. No que toca a melhorias, foi ainda sugerido que fosse retirada a opção de aceder às definições da *action bar* uma vez presente a mesma opção na barra lateral esquerda.

5.2.2 Conetividade

Um dos requisitos iniciais definidos foi garantir o funcionamento desta aplicação cliente sem ligação à Internet. Como tal, foi também importante apurar o funcionamento desta exposta a algumas condições. Como se sabe, são três os tipos de ligações sobre as quais se pode fazer a descarga dos dados, rede móvel apenas (3G/4G), rede móvel e

wifi ou apenas *wifi*, sendo este parâmetro configurável nas preferências. Posto isto, as condições a que a aplicação foi exposta foram as seguintes:

- Funcionamento sem qualquer ligação - Neste cenário, a aplicação apenas consultará a informação local previamente obtida, correndo o risco de não estar atualizada.
- Funcionamento sem ligação de dados - Neste cenário, a aplicação procurará uma ligação *wifi*, caso não exista, consultará a informação já existente, correndo o risco de não estar atualizada.
- Funcionamento sem *wifi* - A aplicação irá procurar por qualquer outra ligação ativa para atualizar os seus dados. Caso não exista, consultará a informação já existente.
- Funcionamento com ligação de dados e *wifi* - Neste cenário, a aplicação optará pela utilização do sistema *wifi* dadas as suas características e da não limitação de dados.

5.2.3 Carga

Os testes de carga são úteis de forma a verificar o limite dos sistemas em termos de processamento, neste caso, sendo utilizados para verificar o comportamento da aplicação cliente enquanto se efetuava o *download* da informação. Assim sendo, e enquanto este *download* estava a ocorrer, foi testado o acesso às várias funcionalidades como consulta de paragens, rotas, cálculo de rotas, acesso ao sistema de favoritos, entre outros, estando a aplicação a responder dentro da normalidade às funcionalidades requeridas.

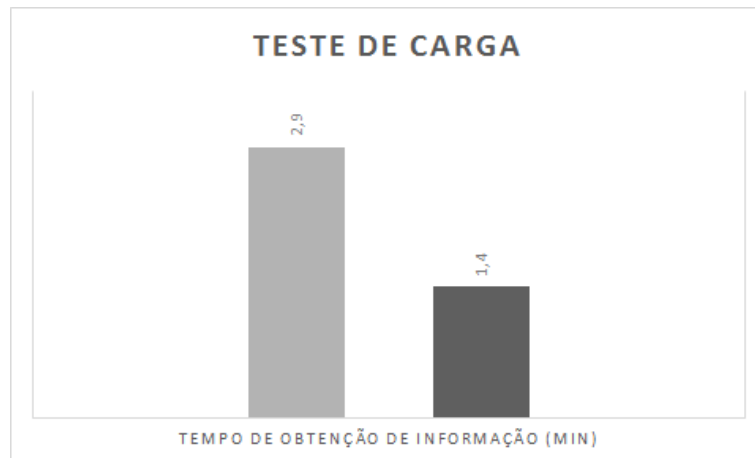


Figura 5.13: Testes de Carga - Tempo de Obtenção de Informação

Ainda assim e como se pode verificar pela Figura 5.12, o *download* da informação estava inicialmente lento (cerca de três minutos para obter toda a informação) tendo, após uma otimização da forma de descodificação desta informação, passado para (cerca de um minuto e quarenta segundos). Embora não esteja ainda nos valores desejáveis, é uma otimização que reduziu o tempo de obtenção dos dados em valores consideráveis.

5.3 Conclusões

O estudo e implementação de um serviço e, posteriormente, a realização de testes ao mesmo são um conjunto de etapas base em qualquer sistema. Como tal, a aplicação cliente apresentada nesta secção foi resultado de um estudo efetuado e presente neste documento que fornece aos utilizadores de transportes públicos, nomeadamente, autocarros, a possibilidade de possuírem informação atualizada em todos os momentos. Embora com algumas limitações, foram realizados três tipos de testes que concluíram que manter a aplicação *clean* e simples de utilizar é uma mais valia para os utilizadores não familiarizados com esta ou mesmo com o sistema operativo móvel em questão, neste caso, o Android. Os três tipos de testes efetuados foram às funcionalidades existentes (o mais importante) como consulta de paragens, remoção de favoritos, acesso às rotas, entre outros. Teste de carga foram também realizados e serviram para apurar o comportamento da aplicação em situação de *stress*, neste caso, enquanto efetuava o *download* da informação quando se navegada pela aplicação. Os testes de conectividade foram também efetuados de forma a avaliar o comportamento da aplicação através

dos diferentes tipos de ligação existentes, 3G/4G e *Wifi* ou ambos e ainda sem qualquer ligação.

Capítulo 6

Conclusão

Esta secção tem por objetivo sumarizar o estudo e implementação de uma solução de resposta a um problema ainda comum no âmbito dos transportes públicos. Hoje em dia, o afastamento das empresas aos seus passageiros através das tecnologias, móveis ou não, é ainda notável, como verificado pelo questionário efetuado às mesmas. Deste modo e de forma a aproveitar a crescente utilização de dispositivos móveis por parte da população, foi estudada, desenhada e implementada uma possível solução de resposta a este problema. Esta solução está orientada a dois objetivos que, apesar de distintos, estão relacionados, isto é, em primeiro lugar, compromete-se a auxiliar as empresas na partilha e disponibilização de informação aos seus passageiros e em segundo lugar a facilitar as deslocações a estes mesmos passageiros através da utilização desta informação.

Estando o foco orientado à utilização de tecnologias *open source*, face às conjunturas atuais, optou-se pela utilização de tecnologias desta natureza. Partindo deste pressuposto, surgiram duas tecnologias que se encaixaram nos requisitos definidos e que, de certa forma são fiáveis uma vez que foram criadas para o âmbito dos transportes públicos. Estas tecnologias ou referências, como também denominadas ao longo deste documento, são conhecidas por GTFS e GTFS *RealTime*. A primeira referência assenta no princípio de criação de um *feed* que é um conjunto de ficheiros de texto com informação estática acerca da empresa de transportes, paragens, rotas, viagens e tempos de paragens. Para além destes, podem e idealmente devem ser adicionados outros de modo a completar o *feed* o mais possível. No caso estudado e abordado para a cidade de Leiria, foi criado um *feed* com o máximo de informação possível sendo que, neste caso, apenas contém as rotas das duas carreiras urbanas do serviço Mobilis. A esta

referência estática foi acrescentada uma outra, denominada de GTFS RT que, como o próprio nome indica, garante informação em tempo real aos passageiros e que se subdivide em três categorias sendo elas alertas de serviço, posições de veículos e atualizações de viagem. Neste caso da cidade de Leiria foi retratado, com esta referência de tempo real, o sistema de alertas que propaga um alerta registado por um funcionário da companhia de transportes através de um *backoffice*. Este *backoffice* tem como função permitir o registo, edição e remoção de um alerta introduzido e que será propagado por toda a rede de utilizadores ativos a utilizarem a aplicação cliente, sendo que aqui é necessária uma ligação de dados.

A aplicação cliente é o segundo interveniente na arquitetura cliente-servidor definida e que foi desenhada e pensada para um utilizador em constante movimento, como é o caso dos passageiros de transportes públicos e, apesar de não se encontrar, à data, na loja da Google *online*, pode ser descarregada¹, instalada e utilizada em qualquer dispositivo com sistema operativo móvel Android. Os dados disponibilizados na versão referida, são as rotas "Mobilis 1" e "Mobilis 2" da empresa Rodotejo, da cidade de Leiria. Para além destas duas tecnologias foi integrada uma outra denominada por *OpenStreetMap* e que tem por base o conceito de *participatory sensing* e que, em consequência disso, tem limitações em termos de validação e fiabilidade de dados, mas que, ao mesmo tempo torna uma solução deste género mais abrangente e prova, sobretudo, que com o auxílio das mais variadas tecnologias se consegue criar uma solução de resposta a um problema nos dias de hoje muito comum, no que diz respeito aos transportes públicos.

6.1 Trabalho Futuro

De forma geral e como qualquer outro sistema serão sempre necessárias alterações e melhorias em todos os sentidos. Não sendo exceção, e começando pelo sistema de gestão da informação, seria interessante existir uma forma de criação e validação de *feeds* estáticos numa ferramenta que as empresas tivessem acesso pois do ponto de vista de primeiras utilizações e sobretudo, para funcionários destas empresas será sempre um desafio a criação de novos ficheiros de informação. Neste sentido, seria igualmente valioso a utilização do *backoffice* para suportar o sistema de configuração e envio de dados para os servidores que integram esta arquitetura e divulgam a informação.

¹<http://pedroferreira.com/utilities/files/BusToGo.apk>

Da mesma forma seria a integração do sistema de posições de veículos e atualizações de viagem em tempo real de forma a tornar a aplicação cliente mais compacta e sobretudo mais "*in real time*". Nesta, para além das funcionalidades existentes, seria importante existirem outras que lhe conferissem mais robustez, como por exemplo, um sistema de histórico de percursos, um sistema de previsão do tempo de chegada a um determinado local ou ainda alertas de saída numa paragem específica.

Bibliografia

- [1] Sapo TeK. Portugueses compram menos telemoveis mas mais smartphones. http://tek.sapo.pt/tek_mobile/equipamentos/portugueses_compram_menos_telemoveis_mas_mais_1323738.html, 2013. [Online; acedido em 19 de Outubro de 2013].
- [2] AAdM. Android e iOS dominam o mercado. <http://abertoatedemadrugada.com/2015/02/android-e-ios-dominam-963-do-mercado.html>, 2015. [Online; acedido em 22 de Fevereiro de 2015].
- [3] Kari Edison Watkins, Brian Ferris, Alan Borning, G. Scott Rutherford, and David Layton. Where is my bus? impact of mobile real-time information on the perceived and actual wait time of transit riders. *Transportation Research Part A: Policy and Practice*, 45(8):839 – 848, 2011.
- [4] Pengfei Zhou, Yuanqing Zheng, and Mo Li. How long to wait?: predicting bus arrival time with mobile phone based participatory sensing. *MobiSys*, pages 379–392, 2012.
- [5] Emiliano De Cristofaro and Claudio Soriente. Participatory privacy: Enabling privacy in participatory sensing. *IEEE Network*, 27(1):32–36, 2013.
- [6] Katie Shilton. Participatory sensing: Building empowering surveillance. *Surveillance and Society*, 8(2), 2010.
- [7] David Alves, Luis M. Martinez, and Jose M. Viegas. Retrieving real-time information to users in public transport networks: An application to the lisbon bus system. *Procedia - Social and Behavioral Sciences*, 54(0):470 – 482, 2012. Proceedings of {EWGT2012} - 15th Meeting of the {EURO} Working Group on Transportation, September 2012, Paris.

- [8] PwC Technology. Consumerization of APIs: Scaling integrations. <http://www.pwc.com/us/en/technology-forecast/2012/issue2/features/feature-consumerization-apis.jhtml>, 2012. [Online; acedido em 15 de Março de 2015].
- [9] Ed Julson. The Growing Imperative of APIs for the Enterprise. <http://www.tibco.com/blog/2015/03/09/15604/>, 2015. [Online; acedido em 25 de Março de 2015].
- [10] Alexander Zipf Pascal Neis. Analyzing the contributor activity of a volunteered geographic information project - the case of openstreetmap. *ISPRS Int. J. Geo-Inf.*, pages 146–165, 27 de Julho de 2012.
- [11] StreetBlog SF. How Google and Portland TriMet Set the Standard for Open Transit Data. <http://pplware.sapo.pt/smartphones-tablets/android-alcanca-recorde-de-81-no-mercado-dos-smartphones/>, 2010. [Online; acedido em 5 de Janeiro de 2015].
- [12] Matthew Roth. How Google and Portland TriMet Set the Standard for Open Transit Data. <http://sf.streetsblog.org/2010/01/05/how-google-and-portlands-trimet-set-the-standard-for-open-transit-data/>, 2010. [Online; acedido em 29 de Dezembro de 2014].
- [13] The Art Of Software. Google Protocol Buffers in Android. <http://wweems.blogspot.pt/2010/09/google-protocol-buffers-in-android.html>, 2010. [Online; acedido em 6 de Julho de 2014].
- [14] Inside Technology. Operating system reviews: Android. <http://www.insidetechology360.com/index.php/operating-system-reviews-android-2-82/>, 2013. [Online; acedido em 26 de Dezembro de 2013].
- [15] Exame Informatica. Android e' o sistema operativo movel mais usado em Portugal. <http://exameinformatica.sapo.pt/noticias/software/2012-01-03-android-e-o-sistema-operativo-movel-mais-usado-em-portugal>, 2012. [Online; acedido em 12 de Novembro de 2013].
- [16] Felix Richter. The Price Gap Between iOS and Android Is Widening. <http://www.statista.com/chart/1903/>

- average-selling-price-of-android-and-ios-smartphones/, 2014. [Online; acessado em 02 de Setembro de 2014].
- [17] Martine Rainho Carlos S. Almeida. Transportes urbanos ganham mais passageiros mas ainda com prejuizo. *Reg. de Leiria*, pages 6–7, 5 de Fevereiro de 2015.
- [18] Siddhartha Bhatia. What is MongoDB and Why todays Enterprise cannot survive without NoSQL database like MongoDB. <http://www.netsolutionsindia.com/blog/what-is-mongodb/>, 2014. [Online; acessado em 14 de Fevereiro de 2015].
- [19] Margaret Rouse. Platform as a Service (PaaS). <http://searchcloudcomputing.techtarget.com/definition/Platform-as-a-Service-PaaS>, 2015. [Online; acessado em 14 de Fevereiro de 2015].
- [20] Pplware. O Android estarã; ainda mais forte em 2015. http://pplware.sapo.pt/smartphones-tablets/windows_phone/o-android-estara-ainda-mais-forte-em-2015/, 2014. [Online; acessado em 11 de Julho de 2014].
- [21] The Verge. Google touts 1 billion active Android users per month. <http://www.theverge.com/2014/6/25/5841924/google-android-users-1-billion-stats>, 2014. [Online; acessado em 29 de Dezembro de 2014].
- [22] Pplware. Smartphones Android venderam 4x mais que iPhones. <http://pplware.sapo.pt/smartphones-tablets/smartphones-android-venderam-4x-mais-que-iphones/>, 2014. [Online; acessado em 15 de Fevereiro de 2014].
- [23] PhoneArena. IDC: Android, Windows Phone see robust growth, iOS plateaus, BlackBerry and others flailing. <http://goo.gl/jrG8K5>, 2013. [Online; acessado em 19 de Outubro de 2013].
- [24] Google. Android, the world's most popular mobile platform. <http://developer.android.com/about/index.html>, 2013. [Online; acessado em 19 de Outubro de 2013].
- [25] CNET. Apple by the numbers, from WWDC 2013. http://news.cnet.com/8301-13579_3-57588547-37/apple-by-the-numbers-from-wwdc-2013/, 2013. [Online; acessado em 23 de Outubro de 2013].

- [26] The Next Web. Microsoft: Windows Phone 160,000 apps see more than 200 million monthly downloads. <http://goo.gl/ksjy1v>, 2013. [Online; acedido em 27 de Outubro de 2013].
- [27] IDC. Android and iOS Combine for 92.3Shipments in the First Quarter While Windows Phone Leapfrogs BlackBerry, According to IDC. <http://www.idc.com/getdoc.jsp?containerId=prUS24108913>, 2013. [Online; acedido em 27 de Outubro de 2013].
- [28] zeenews. Half of all apps on iOS, Android, and Windows Phone are dead: Report. http://zeenews.india.com/business/news/technology/half-of-all-apps-on-ios-android-and-windows-phone-are-dead-report_82794.html, 2013. [Online; acedido em 03 de Outubro de 2013].

Apêndice A

Análise de Aplicações

A.1 Aplicações Existentes

Hoje em dia escolher um *smartphone* implica escolher a cor, o modelo e todas as outras características de modo a agradar a todos os gostos e feitios. Tal como a parte "externa", está também a parte "interna" desse *smartphone*, isto é, o seu sistema operativo. Cada vez mais a oferta é maior, sendo neste momento liderada pela Google com o seu Android, acompanhada de perto pelo iOS da Apple e depois o Windows Phone, da Microsoft.[23] Assim, esta secção apresenta a análise feita às aplicações existentes nas diferentes lojas das diversas plataformas. O objetivo desta análise é retirar o melhor e pior de todas as aplicações de modo a que a solução final, depois de implementada, consiga conciliar o melhor de todas estas e evitar o pior, de forma a que, quando for lançada para o mercado, venha a obter o sucesso esperado.

A.1.1 Android

A Google lançou o Android em 2007 e desde aí que o seu crescimento tem sido exponencial sendo, hoje em dia, a plataforma móvel mais utilizada em todo o mundo.[24] Da análise de aplicações feitas ao *market* da Google, apresentam-se, de seguida, as que mais se destacam. Os pontos que contribuem para que uma aplicação seja destacada na análise efetuada são não só as suas funcionalidades, como também a interface que apresentam.

A primeira das aplicações estudadas é a UK Bus Checker¹ que está disponível nas

¹de FatAttitude Ltd, disponível em <http://www.buschecker.com/>

versões Android, iOS e Windows Phone.

UK Bus Checker

Descrição: A aplicação UK Bus Checker está disponível nas versões Lite e Pro. O principal objetivo desta é de efetuar a monitorização de percursos e tempos de chegada dos autocarros, em tempo real, nas diversas paragens do Reino Unido, de modo a auxiliar o utilizador nas suas deslocações diárias.

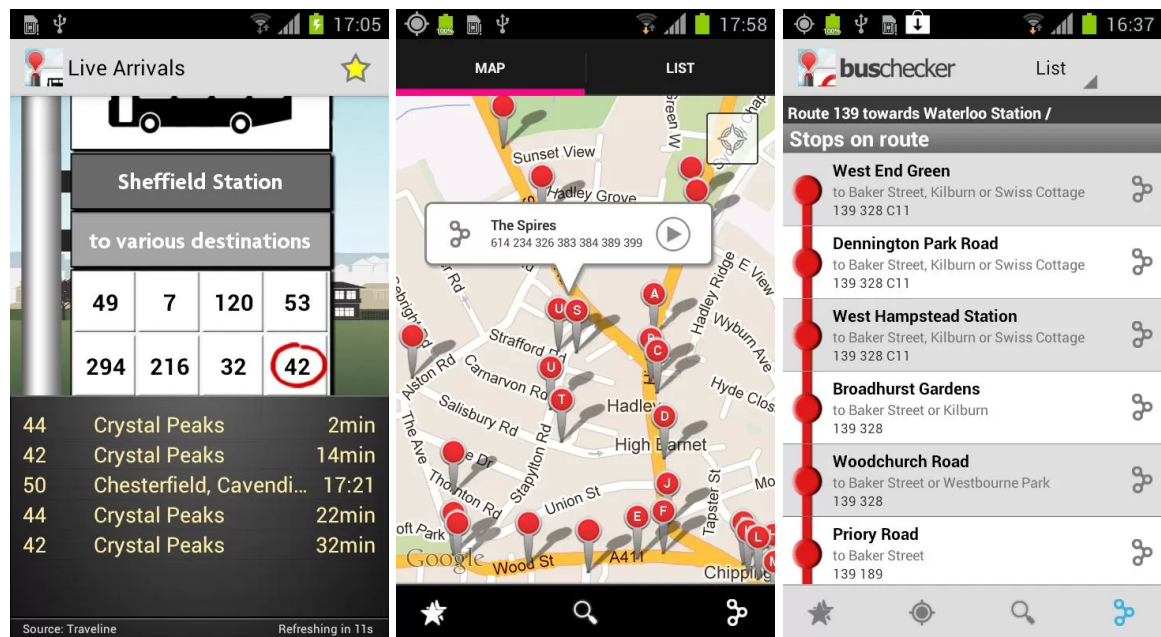
Pontos de destaque:

- Possibilidade de acesso a todos os autocarros existentes em determinada paragem e respetivos tempos de espera;
- Permite visualizar o mapa da rota de cada autocarro;
- Lista as diversas paragens para cada rota;
- Guarda favoritos de paragens;
- Possui um *widget*, configurável, com o tempo dos próximos autocarros para determinada paragem.

Pontos negativos:

- Interface muito confusa;
- Má localização dos botões e respetivas funcionalidades.

Interface:



(a) Tempo de espera numa paragem

(b) Mapa de paragens

(c) Detalhes de uma rota

Figura A.1: Interface da Aplicação UK Bus Checker

Não para o Reino Unido, mas para Londres, a aplicação Live London Bus Tracker,² é mais uma das que se destacam no *market*, devido à sua simplicidade de utilização.

Live London Bus Tracker

Descrição: A aplicação Live London Bus Tracker tal como a UK Bus Checker, dá ao utilizador a possibilidade de saber o tempo para o próximo autocarro chegar a determinada paragem. Nesta, é ainda possível saber as rotas de cada autocarro, bem como a localização de cada uma das paragens.

Pontos de destaque:

- Diversidade de opções no planeamento de uma rota;
- Ecrã de boas-vindas na primeira utilização;
- Boa distribuição das funcionalidades no ecrã;
- Acesso ao tempo que falta para o próximo autocarro em determinada paragem após a seleção da mesma no mapa;

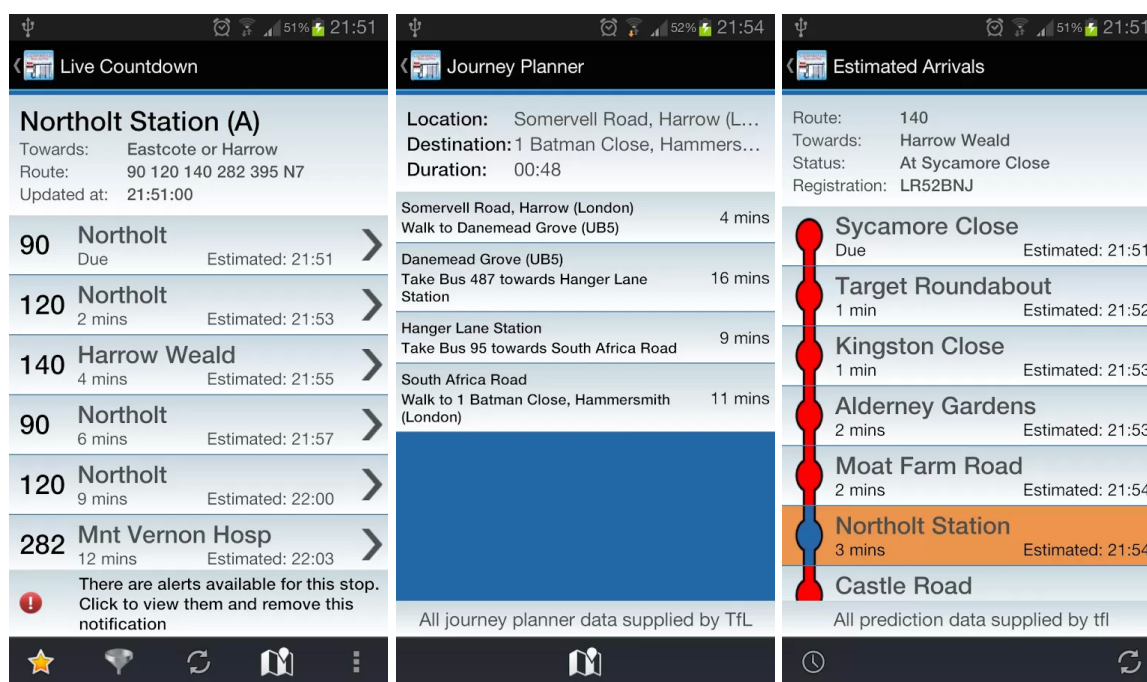
²de AppEffectsUK, disponível em <http://bit.ly/LiveLondonBusTracker>

- Alertas de aproximação de autocarro;
- Pesquisa de paragens com várias alternativas ;
- Definição de uma paragem "home".

Pontos negativos:

- Localização do sistema através da rede de dados não existente.

Interface:



(a) Tempos de chegada

(b) Cálculo de um percurso

(c) Detalhes de uma rota

Figura A.2: Interface da Aplicação Live London Bus Tracker

A seguinte aplicação, New Madrid Metro - Bus - Cercanias ³, apesar da sua vertente de ligação aos transportes públicos da cidade de Madrid, deixa um pouco a desejar pois concentra muita informação mas mal distribuída no ecrã, isto é, nota-se pela sua utilização que não foi dada prioridade à interação com o utilizador. Ainda assim, esta consegue ter muito boa classificação no *market* e os seu número de *downloads* já atingiu valores entre os 100.000 e 500.000. A sua inclusão nesta análise deve-se à componente de autocarros presente.

³de DMP S.L., disponível em <http://bit.ly/NewMadridMetro>

New Madrid Metro - Bus - Cercanias

Descrição: A New Madrid Metro - Bus - Cercanias é uma aplicação que permite aos seus utilizadores controlar os tempos que faltam para os próximos autocarros nas diversas paragens existentes no centro e toda a periferia da cidade de Madrid. A análise desta aplicação foi efetuada devido à componente de autocarros existente, apesar desta possuir informações acerca das redes de metro e comboio.

Pontos de destaque:

- Acesso ao tempo de espera em determinada paragem;
- Procura de rota por número ou por nome;
- Procura do autocarro mais próximo da localização atual;
- Informações e horários de toda a rede de serviços disponíveis.

Pontos negativos:

- Procura de paragens pouco intuitiva;
- Impossibilidade de acesso a todas as paragens, nem mesmo por rota;
- Dependência do número das linhas/rotas e autocarros para efetuar pesquisas.

Interface:

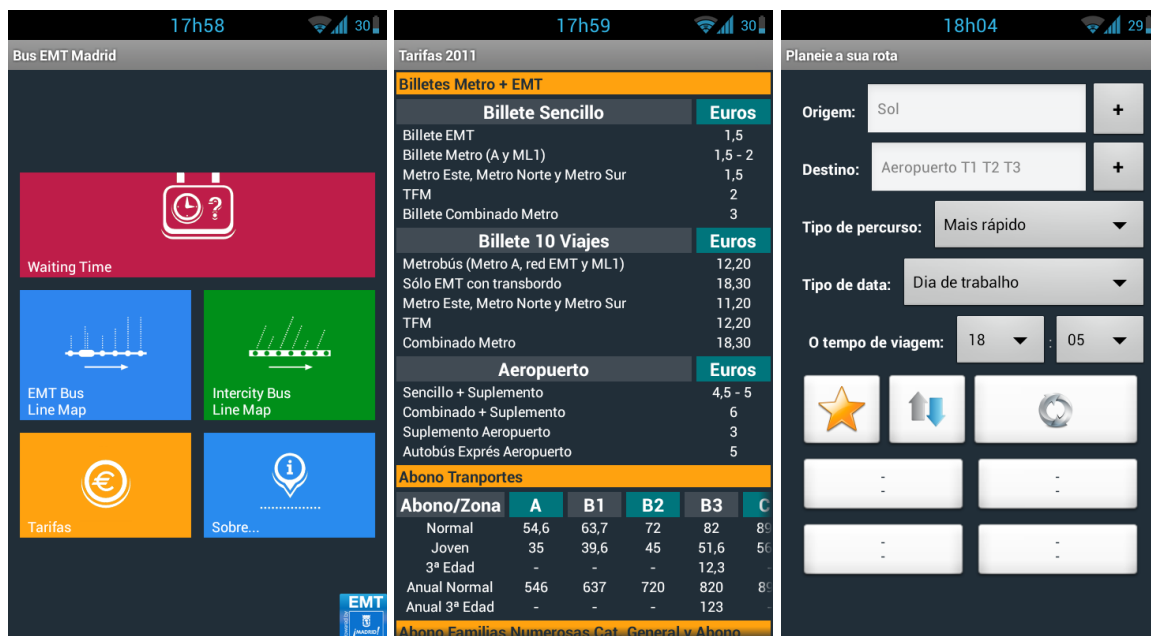


Figura A.3: Interface da Aplicação New Madrid Metro - Bus - Cercanias

A análise de aplicações é importante não só para obter informações acerca das mesmas mas também para ter a noção dos locais onde estas são utilizadas. Das anteriores, notou-se uma forte incidência na Europa, em países como Inglaterra e Espanha, bem como outras pertencentes ao Reino Unido. De seguida, surgem algumas aplicações que, ao estarem em português, mereceram destaque, sendo utilizadas no Brasil nas cidades de Fortaleza, Porto Alegre e São Paulo.

Bus K Fortaleza

Descrição: A aplicação Bus K Fortaleza⁴, e como o próprio nome indica, foi desenvolvida para a cidade de Fortaleza, no Brasil. O objetivo desta aplicação é permitir que os seus utilizadores consultem os itinerários e localização das várias paragens de forma facilitada.

Pontos de destaque:

- Acesso às tarifas das viagens;
- Disponibilização de uma secção de notícias;
- Procura do autocarro mais próximo;
- Alertas de aproximação de autocarro a uma paragem previamente selecionada;
- Interface com as funcionalidades bem distribuídas e fácil acesso.

Pontos negativos:

- O aspeto gráfico;
- Funcionalidade de notícias com muitos *bugs*;
- Poucas funcionalidades tendo em conta aquilo a que se propõe;
- Só há acesso aos detalhes das paragens depois de ser efetuada uma pesquisa.

Interface:

⁴de Adapter Tecnologia, disponível em <http://bit.ly/BusKFortaleza>

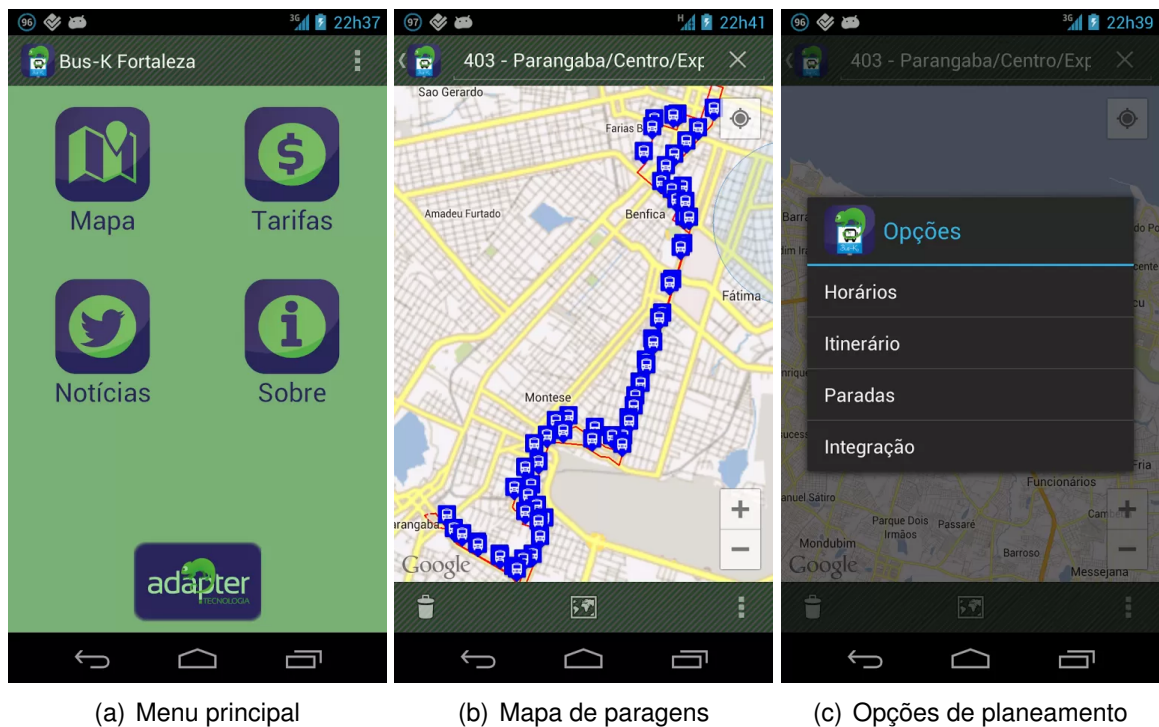


Figura A.4: Interface da Aplicação Bus K Fortaleza

Bus Brazil

Descrição: A aplicação Bus Brazil ⁵ tem como principal objetivo de dar aos seus utilizadores acesso aos horários e itinerários dos autocarros de uma determinada rota, para um determinado destino.

Pontos de destaque:

- Adição de rotas aos favoritos;
- Visualização de horários para determinada rota;
- Possibilidade de acesso aos itinerários das rotas;
- Interface simples e minimalista.

Pontos negativos:

- Para além dos horários, deveria disponibilizar um mapa com rotas;
- Deveria haver a possibilidade de visualizar as paragens no mapa.

Interface:

⁵de Bus Brazil, disponível em <http://bit.ly/BusBrazil>

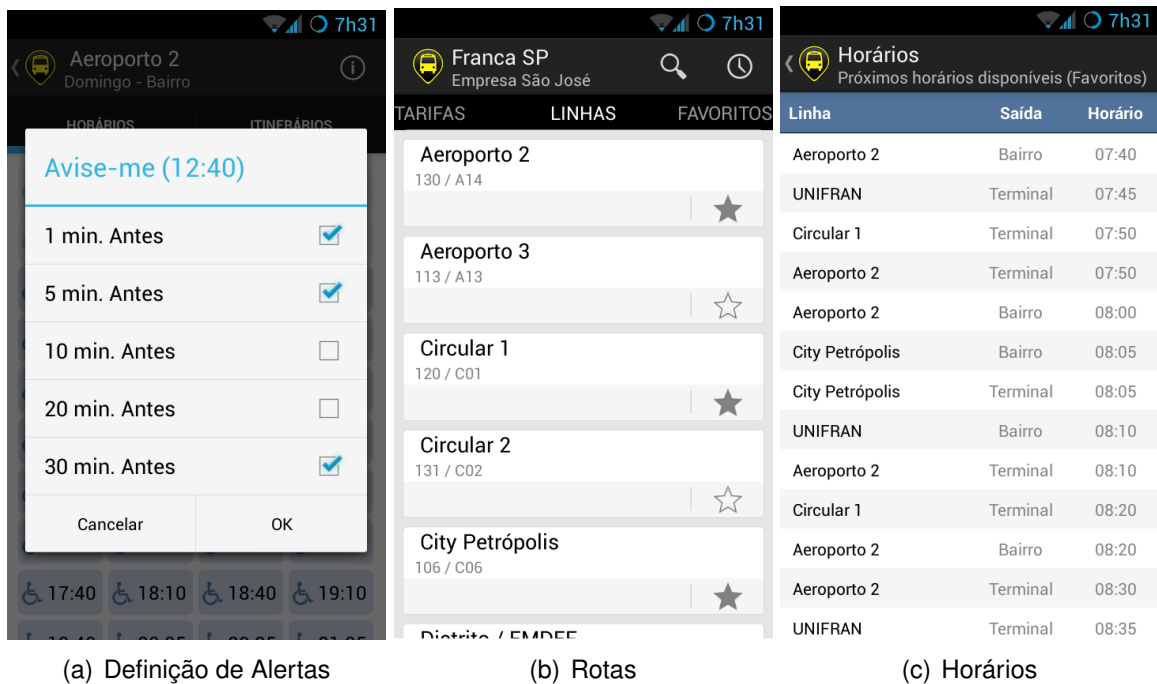


Figura A.5: Interface da Aplicação Bus Brazil

A aplicação Moobly⁶ foi, de todas as analisadas, a que mais se destacou não só pela sua usabilidade como também pela sua forma de disponibilização dos dados.

Moobly

Descrição: Implementada para ser utilizada, maioritariamente, pelos utilizadores de transportes públicos da região de Porto Alegre, no Brasil, a Moobly permite saber quais os melhores autocarros para fazer as deslocações de um ponto origem para um ponto destino e, através disso, consultar as melhores opções para o fazer.

Pontos de destaque:

- Vasto leque de funcionalidades;
- Grande destaque ao mapa da região com as paragens;
- Integração com um sistema de táxis;
- Secção de notícias/atualidade;
- Acesso a informação sobre tarifas, linhas e até mesmo lotação de autocarros;
- Procura de paragens por perto bastante intuitiva;

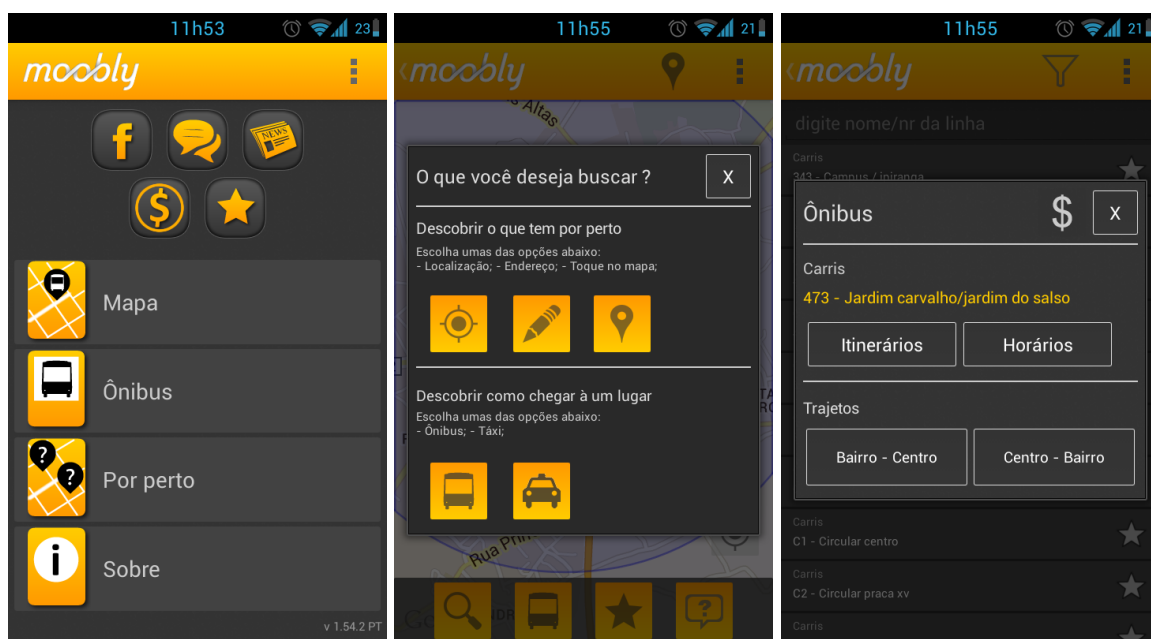
⁶de Ederson Schmidt, disponível em <http://www.moobly.com.br/>

- Possibilidade de filtragem por companhias de serviços;
- Interface bastante bem construída e com acesso fácil a todas as funcionalidades.

Pontos negativos:

- Ter que fazer *login* com uma conta de Facebook ou conta do serviço de modo a aceder e usufruir a funcionalidades "extra".

Interface:



(a) Menu principal

(b) Procura por perto

(c) Detalhes de uma rota

Figura A.6: Interface da Aplicação Moobly

Depois do Brasil e voltando para a Europa, a RATP: Paris, IZI Carris, Sapo Transportes e a MOVE-ME são as aplicações que se seguem. A primeira foi construída para ser utilizada em Paris, tal como o nome indica, enquanto que as três seguintes são portuguesas sendo a sua zona de utilização a grande Lisboa.

Disponível para as três plataformas, a RATP ⁷ (*Régie Autonome des Transports Parisiens*), tal como as anteriores, tem pontos que se destacam e alguns pontos negativos, que de seguida se apresentam.

⁷de RATP, disponível em <http://bit.ly/RATP-Paris>

RATP: Paris

Descrição: Com um objetivo semelhante às aplicações anteriormente analisadas, a RATP: Paris compromete-se a dar aos seus utilizadores o acesso a toda a informação acerca da rede de transportes na zona de Paris, em França.

Pontos de destaque:

- Definição de vários parâmetros no cálculo da rota;
- Acesso ao estado do trânsito.

Pontos negativos:

- Maioria do conteúdo da aplicação é carregado através de *webviews*;
- Definição de alertas não é uma tarefa simples;
- *Crashes* constantes;
- Mapa das linhas dos autocarros muito desordenado;
- Sub-menus confusos;
- É complicado aceder ao tempo que falta para o próximo autocarro passar em terminada paragem.

Interface:

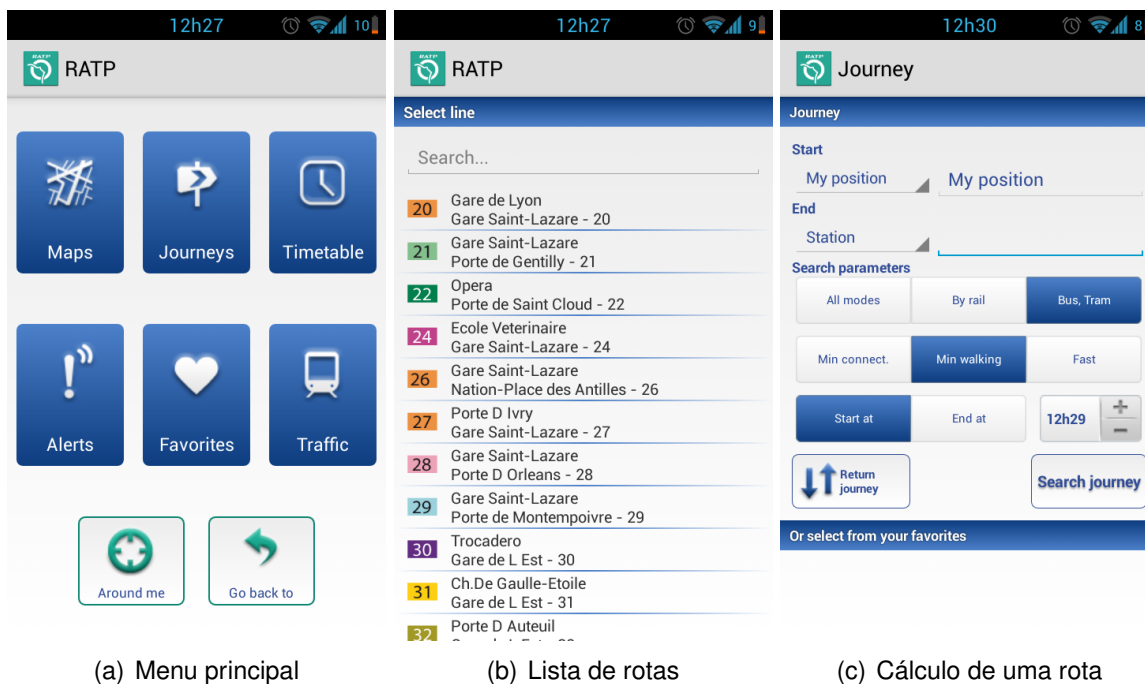


Figura A.7: Interface da Aplicação RATP: Paris

As duas próximas aplicações pertencem a empresas portuguesas e são utilizadas em Portugal, para consulta de autocarros e rotas existentes na cidade de Lisboa. Em comparação com algumas aplicações analisadas anteriormente neste documento, tanto a "IZI Carris"⁸ como a "Sapo Transportes"⁹ ficam à quem daquilo que é esperado das aplicações deste género e, apesar de ser elevada a quantidade de trabalho investido nas mesmas, fica-se com a sensação que estas carecem de algumas funcionalidades básicas, comuns noutras aplicações, como por exemplo, uma calculadora de percurso, no caso da "IZI Carris" ou a disponibilização do tempo para o próximo autocarro na "Sapo Transportes".

IZI Carris

Descrição: Aplicação desenvolvida para a rede de transportes de Lisboa, a Carris. A IZI Carris permite a que todos os seus utilizadores saibam os autocarros que fazem paragens em determinados pontos, bem como de todas as rotas existentes para os vários destinos da cidade.

Pontos de destaque:

- Possibilidade de acesso, em determinada paragem, ao tempo que falta para o próximo autocarro chegar;
- Visualização das rotas dos autocarros;
- Adição de paragens aos favoritos.

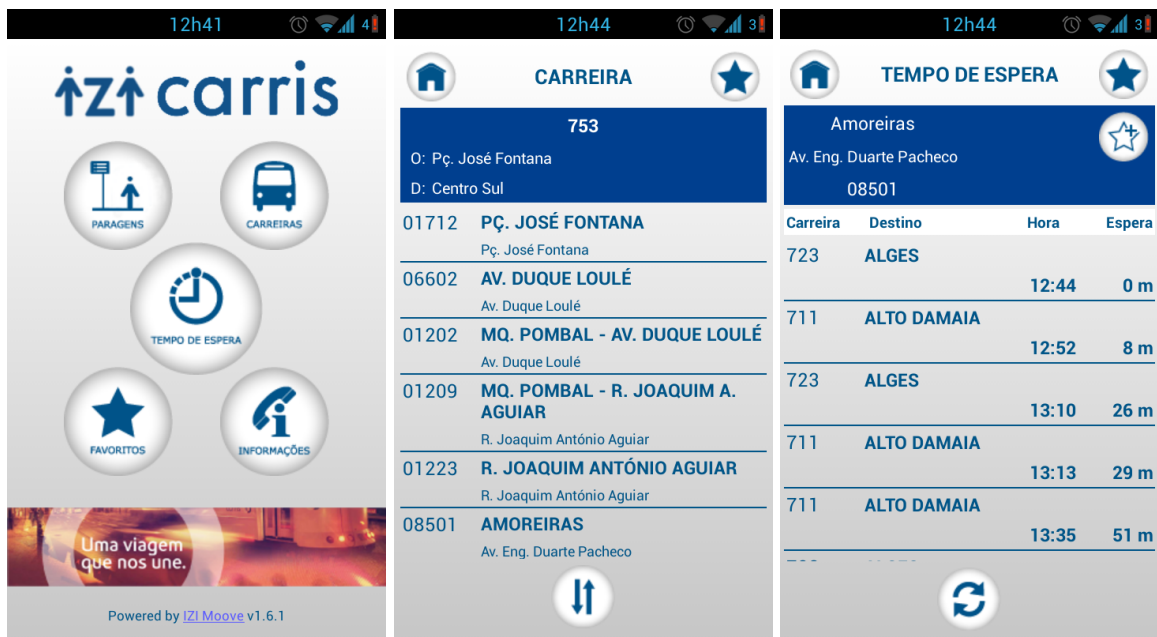
Pontos negativos:

- Em certos ecrãs, perde-se a noção onde se está;
- Devia conter os tempos médios da chegada dos autocarros, em modo *offline*.

Interface:

⁸de IZIMOOVE, disponível em <http://bit.ly/IZICarris>

⁹de PT Comunicações, disponível em <http://mobile.sapo.pt/app/transportes>



(a) Menu principal

(b) Lista de rotas

(c) Cálculo de uma rota

Figura A.8: Interface da Aplicação IZI Carris

Sapo Transportes

Descrição: A aplicação Sapo Transportes é uma aplicação portuguesa criada pela PT Comunicações. Esta aplicação permite aos seus utilizadores acederem a informação de transportes da Área Metropolitana de Lisboa, sejam eles, autocarro, barco, comboio ou metro.

Pontos de destaque:

- Adição de paragens aos favoritos;
- Agregação de vários tipos de transporte;
- Possibilidade de escolha da(s) companhia(s) que disponibilizam o serviço;
- Acesso aos horários dos próximos autocarros nas paragens.

Pontos negativos:

- Dependência de rede para acesso à informação;
- *Crashes* constantes tornando a aplicação muito lenta;
- Interface confusa;
- É notório um enorme *delay* desde a interação com a aplicação e resposta desta;
- Não existência de um modo *offline*.

Interface:

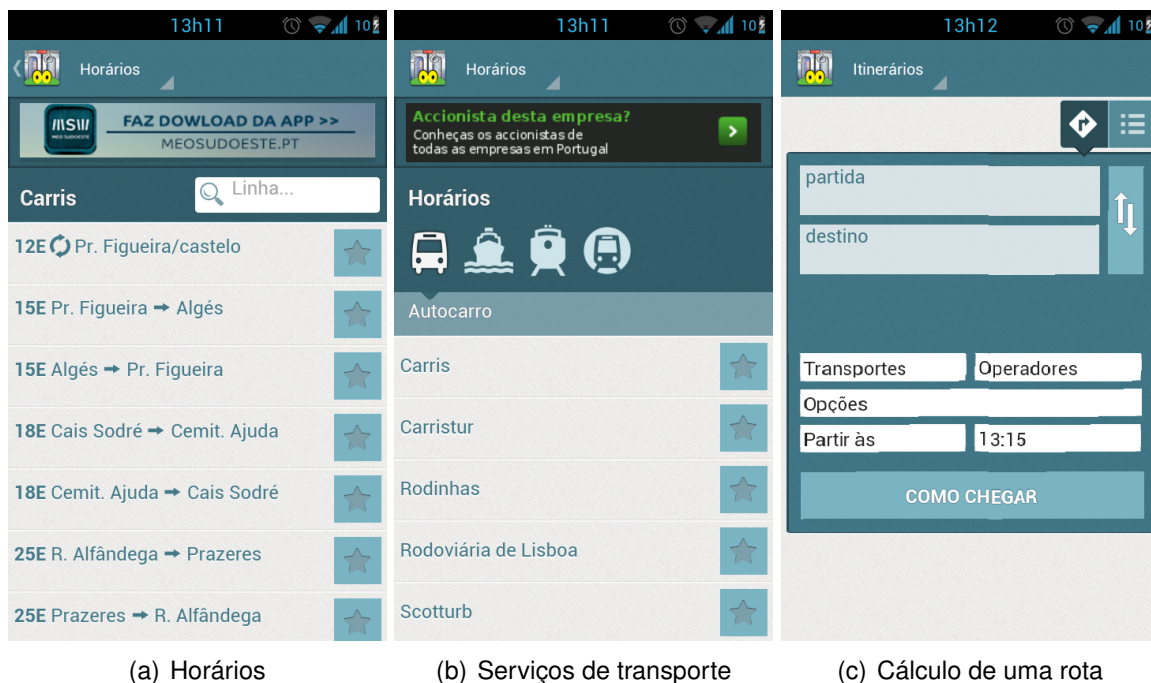


Figura A.9: Interface da Aplicação Sapo Transportes

NUS NextBus

Descrição: A aplicação NUS NextBus¹⁰ foi desenvolvida com o objetivo de auxiliar os passageiros de autocarros de Singapura.

Pontos de destaque:

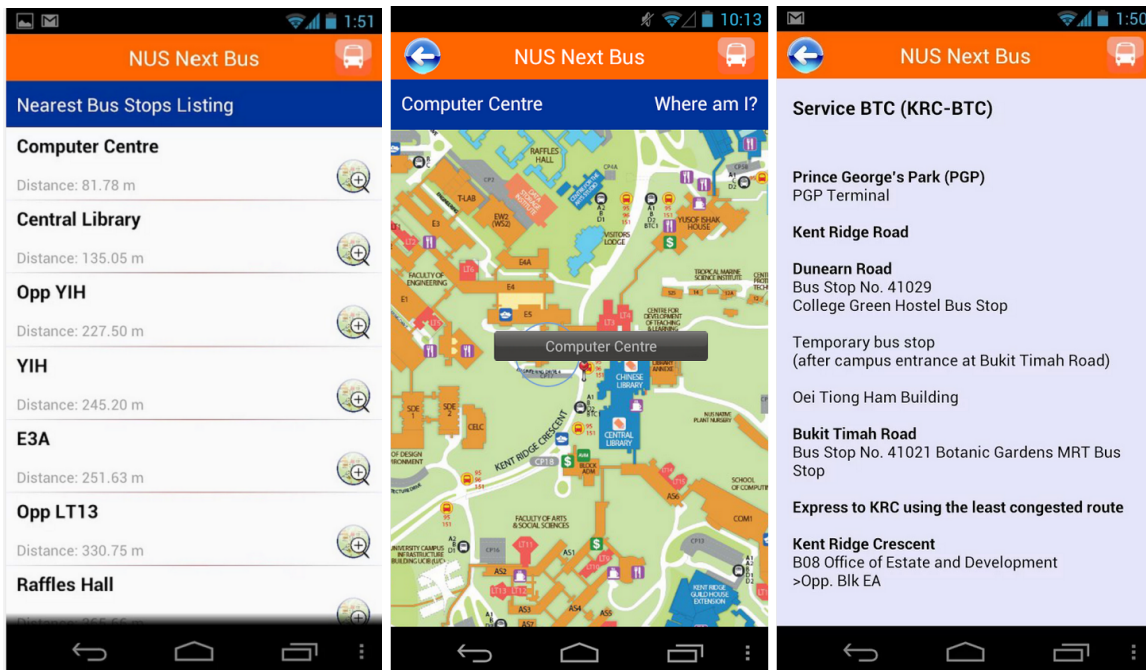
- Possibilidade de acesso às paragens mais próximas e visualizá-las no mapa;
- Acesso ao tempo que falta para o próximo autocarro.

Pontos negativos:

- Esquema de cores muito "agressivo";
- Mapa de paragens e localização muito confuso.

Interface:

¹⁰de National University of Singapore, disponível em <http://bit.ly/ANextBus>



(a) Paragens mais próximas

(b) Mapa da localização atual

(c) Secção de notícias

Figura A.10: Interface da Aplicação NUS NextBus

Na presente análise foi estabelecida como prioridade a inclusão de aplicações portuguesas fossem ou não de destaque, pois além do que já existe lá fora, é importante perceber o que já existe a funcionar em Portugal para corrigir o pior e melhorar o que já está de acordo com o que uma aplicação deste género exige. Posto isto, a análise da aplicação MOVE-ME, que se segue, foi fundamental para as conclusões retiradas sobre as aplicações utilizadas em Portugal, contrariando aquilo que foi referido aquando da análise das aplicações IZI Carris e Sapo Transportes, ambas da grande Lisboa. Assim sendo, a aplicação que se segue está disponível não só para a cidade de Lisboa, como também para a cidade do Porto, podendo ser utilizada tanto em *smartphones* Android como também, iOS.

MOVE-ME

Descrição: Destinada a todos aqueles que fazem uso de transportes públicos, esta aplicação apresenta-se com uma interface um diferente do habitual (nas aplicações deste género), sendo que aqui se optou pelos botões grandes muito devido à forma em como vai ser utilizada, isto é, na maioria das vezes, em movimento. No que toca às suas

funcionalidades, as que contém funcionam como o esperado, embora no cômputo geral sejam escassas.

Pontos de destaque:

- Procura de todas as paragens existentes através da introdução de um raio de procura;
- Disponibilização de pontos de interesse;
- Possibilidade de utilização em duas grandes cidades.

Pontos negativos:

- Interface mal construída;
- Escassas funcionalidades.

Interface:

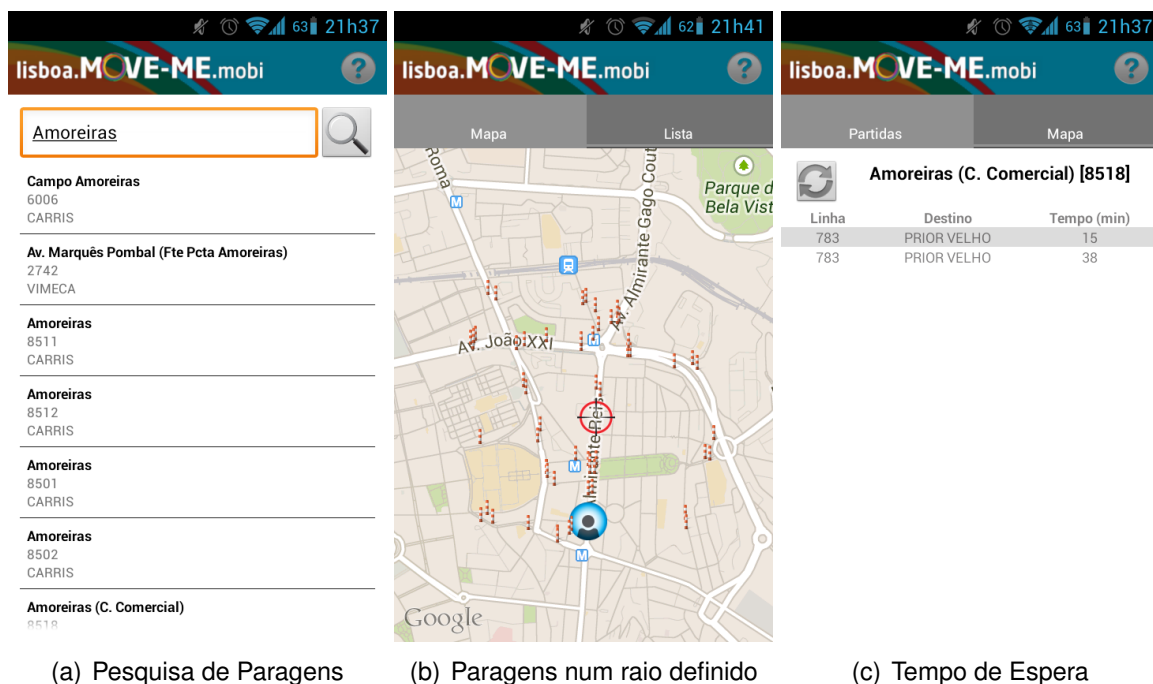


Figura A.11: Interface da aplicação MOVE-ME

A.1.2 iOS

A "era" *smartphone* começou com o iOS da Apple em 2007. Até hoje, a Apple já vendeu mais de 600 Milhões de dispositivos com iOS, e tem mais de 900.000 aplicações na sua *store*.^[25] Dada a sua importância no mercado e no mundo tecnológico é importante ter

em consideração as aplicações existentes nesta plataforma, uma vez que, tal como no Android, é muita a oferta existente. Assim, esta secção é onde se encontram listadas as aplicações que mais se destacaram do lote de todas as testadas, aquando da sua análise.

London Bus Live Countdown

Descrição: A aplicação London Bus Live Countdown¹¹ foi criada para todos aqueles utilizadores que necessitam de saber o próximo autocarro em determinada paragem. Dá a possibilidade de saber a localização atual com um mapa de paragens por perto e ainda todos os tempos de demora da viagem planeada.

Pontos de destaque:

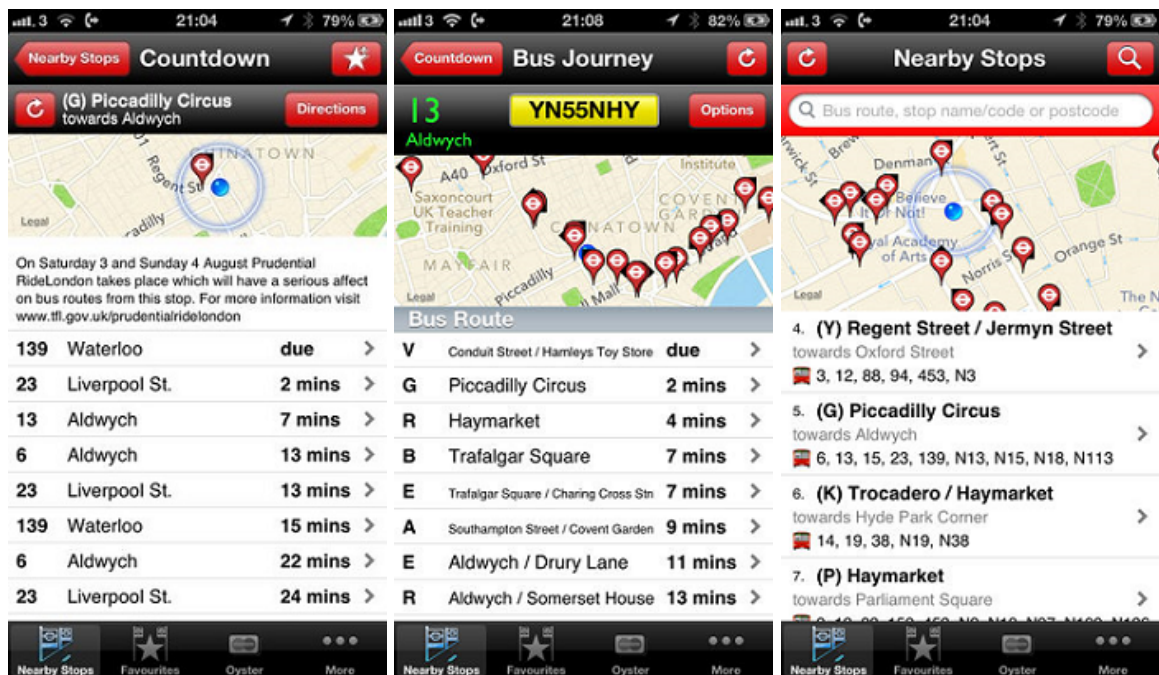
- Adição de favoritos;
- Disponibilização de um mapa de paragens próximas;
- Visualização de "desvios" de rota;
- Possibilidade de ver estado do trânsito;
- Calculadora de percursos.

Pontos negativos:

- Listagem das paragens mais próximas confusa.

Interface:

¹¹de Polydiam, disponível em <http://bit.ly/LondonBusLiveCountdown>



(a) Paragens mais próximas

(b) Tempo de Espera

(c) Percurso

Figura A.12: Interface da Aplicação Live London Countdown

Como se pode verificar, também para iOS, a quantidade de aplicações disponíveis nesta área para o Reino Unido, Inglaterra inclusive, são bastante elevadas. Depois da Live London Countdown, a NextBuses¹², de seguida apresentada, é mais uma das aplicações disponíveis para a cidade de Londres que, tal como a maioria das outras, permite ao seu utilizador o controlo sobre a movimentação dos autocarros, os seus tempos e ainda as suas rotas. A NextBuses apenas se encontra disponível para iOS e na língua inglesa e já conta com inúmeras classificações positivas.

NextBuses

Descrição: NextBuses é uma aplicação que permite aos seus utilizadores ter acesso ao tempo que falta para os próximos autocarros na paragem que estes pretendam. Dá ainda a possibilidade de aceder ao tempo entre paragens para a mesma rota.

Pontos de destaque:

- Adição de favoritos;

¹²de Malcolm Barclay, disponível em <http://bit.ly/NextBuses>

- Disponibilização de um mapa de paragens próximas;
- Possibilidade de procura de paragens por rua.

Pontos negativos:

- Carece de funcionalidades.

Interface:

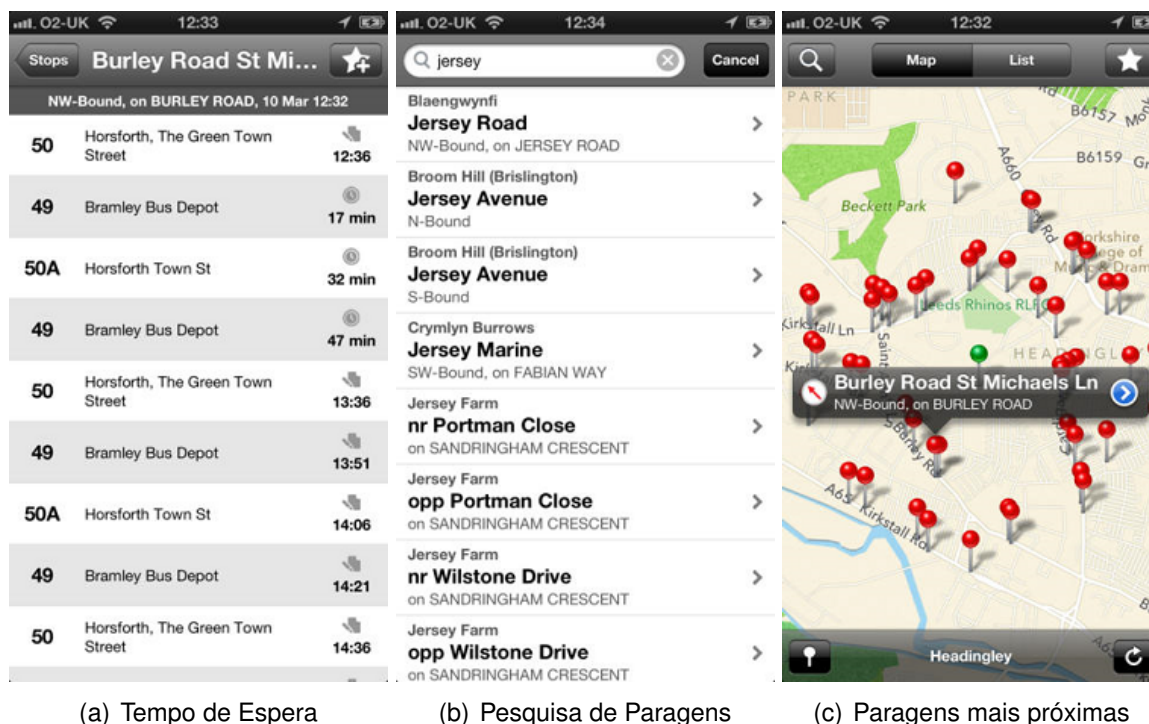


Figura A.13: Interface da Aplicação NextBuses

iNextBus

Descrição: iNextBus¹³ está disponível para iOS e dá a possibilidade dos seus utilizadores saberem os tempos para os próximos autocarros nas diversas paragens. Além de permitir visualizar a rota de determinado autocarro no mapa, tem ainda como funcionalidade adicional a adição de favoritos, sejam eles paragens ou rotas.

Pontos de destaque:

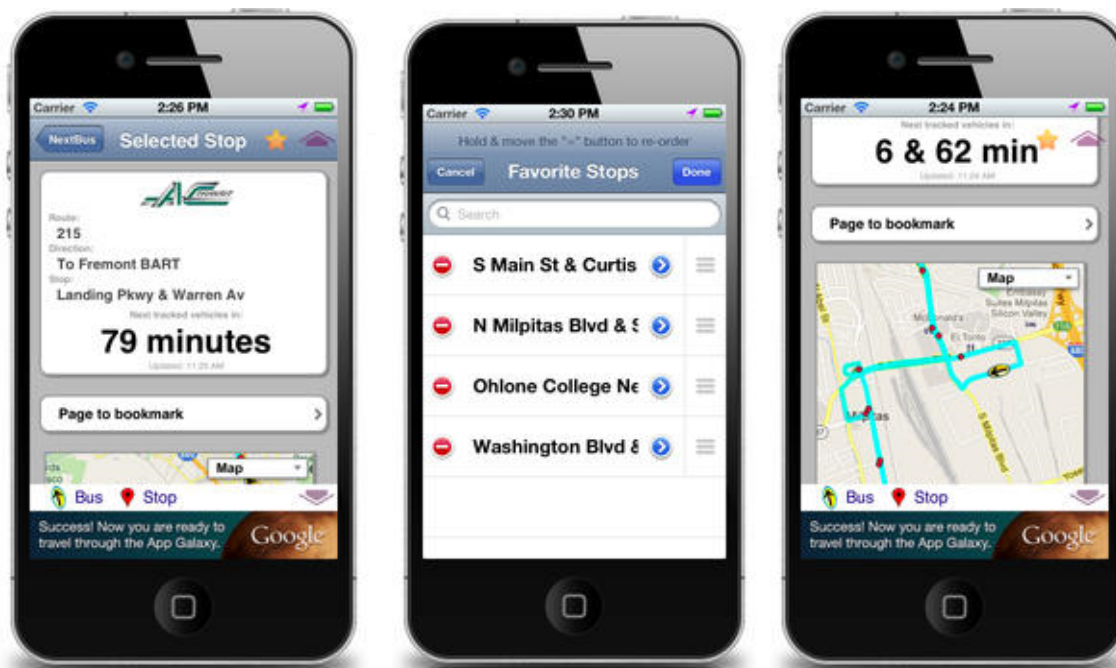
- Adição de favoritos;
- Interface bem construída e de fácil utilização;
- Possibilidade de visualizar a rota do autocarro no mapa.

¹³de Idemfactor Solutions, disponível em <http://bit.ly/iNextBus>

Pontos negativos:

- A informação deveria estar mais repartida por página.

Interface:



(a) Rota do autocarro no mapa

(b) Gestão de Favoritos

(c) Tempo para o próximo auto-carro

Figura A.14: Interface da Aplicação iNextBus

InTime NextBus

Descrição: Esta aplicação, InTime NextBus ¹⁴, permite aos seus utilizadores ter acesso ao tempo que falta para os próximos autocarros na paragem que estes pretendam bem como a interação com o mapa de autocarros e paragens existentes.

Pontos de destaque:

- Definição de alertas;
- Visualização do autocarro em deslocação no mapa;
- Possibilidade de visualizar todas as paragens no mapa.

Pontos negativos:

- Interface muito confusa;

¹⁴de TnT Brain GbR, disponível em <http://bit.ly/InTimeNextBus>

- Informação muito concentrada por ecrã.

Interface:



(a) Rota do autocarro no mapa

(b) Paragens

(c) Paragens guardadas como favoritos

Figura A.15: Interface da Aplicação InTime NextBus

Um dos pontos de exclusão de aplicações no momento da sua análise foi a *interface* logo depois do número de funcionalidades e qualidade das mesmas. Do lote de aplicações analisadas, no conjunto das plataformas, a que se segue, de nome SG NextBus¹⁵, foi uma das que mais se destacou pela qualidade com que foi pensada, desenhada e construída, notando-se aqui uma forte "preocupação" na interação desta com o utilizador porém, um dos seus pontos negativos é a utilização de demasiadas animações na transição de ecrãs, que torna a aplicação não tão agradável nos aparelhos iOS mais antigos e, logicamente, com mais carência de requisitos de *hardware*.

SG NextBus

Descrição: Com um visual bastante atrativo, a aplicação SG NextBus permite aos seus utilizadores ter acesso ao tempo que falta para os próximos autocarros em determinada paragem. Esta foi construída para auxiliar os utilizadores de transportes públicos na

¹⁵de Hon Cheng Muh, disponível em <http://bit.ly/SGNextBus>

cidade de Singapura.

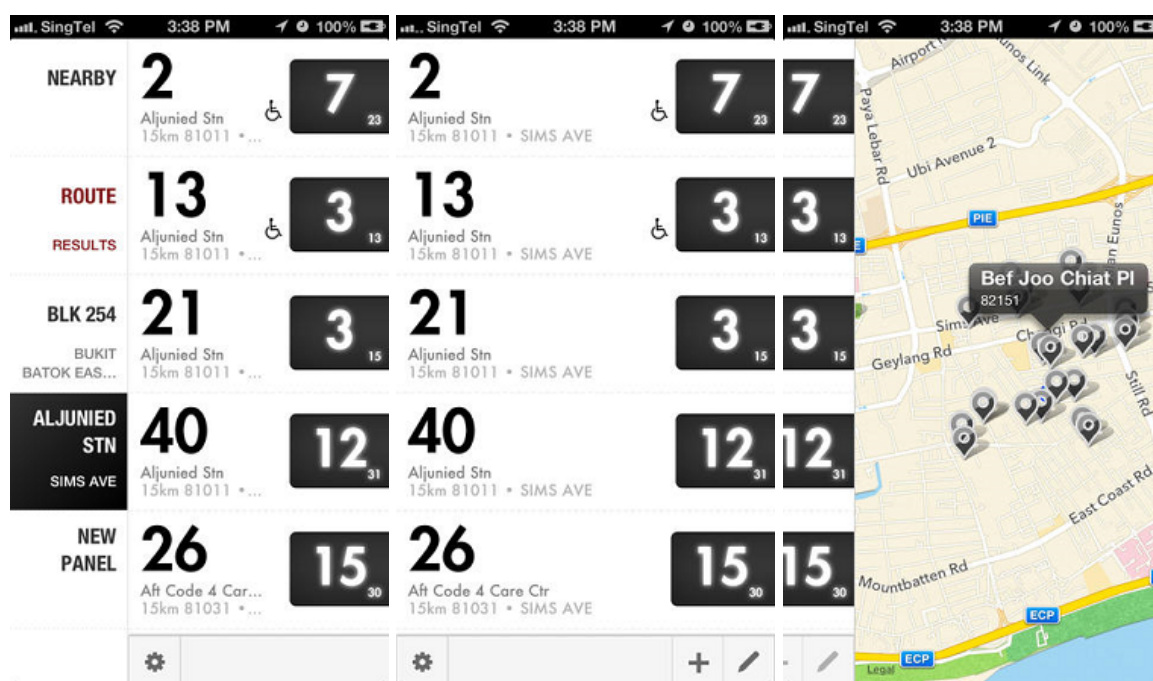
Pontos de destaque:

- Tempo de espera para o próximo autocarro;
- Possibilidade de visualizar todas as paragens no mapa;
- Acesso às paragens mais próximas e respetivos tempos;
- Informação acerca dos autocarros que permitem o transporte de pessoas que se deslocam através de cadeira de rodas.

Pontos negativos:

- Demasiadas animações na transição de ecrãs.

Interface:



(a) Menu Principal

(b) Autocarros e Tempos

(c) Mapa de Paragens

Figura A.16: Interface da Aplicação SG NextBus

Algumas das aplicações que têm sido analisadas ao longo deste documento estão disponíveis para mais que uma plataforma. Mais um exemplo disso é a NUS NextBus¹⁶, abaixo analisada, que está disponível para as versões Android e iOS.

¹⁶de National University of Singapore, disponível em <http://bit.ly/NUSNextBus>

NUS NextBus

Descrição: NUS NextBus é uma aplicação que faz uso da informação de transportes públicos a circular, nomeadamente autocarros, e que dá ao utilizador a percepção das paragens mais próximas e dos tempos que faltam para os próximos autocarros chegarem a essas mesmas paragens.

Pontos de destaque:

- Apresentação do tempo de espera muito bem conseguida;
- Visualização das paragens mais próximas e indicação da sua distância ao ponto atual.

Pontos negativos:

- Interface mal conseguida para algumas funcionalidades;
- Má escolha de cores.

Interface:

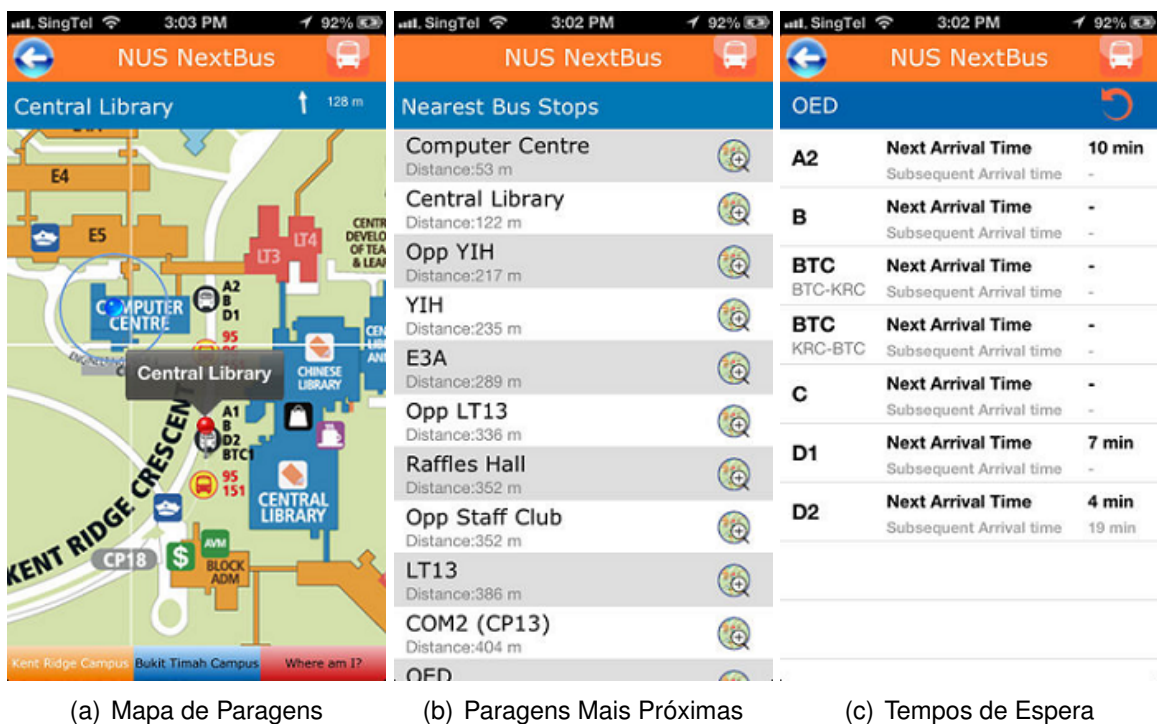


Figura A.17: Interface da Aplicação NUS NextBus

RATP: Paris

Descrição: RATP: Paris ¹⁷ está disponível para os três sistemas operativos móveis mais conhecidos e utilizados, Android, iOS e Windows Phone. Como tal, esta permite que os seus utilizadores saibam o tempo que falta para os próximos autocarros, metros ou comboio, bem como lhes dá a possibilidade de calcular percursos a partir de um ponto origem para um ponto destino.

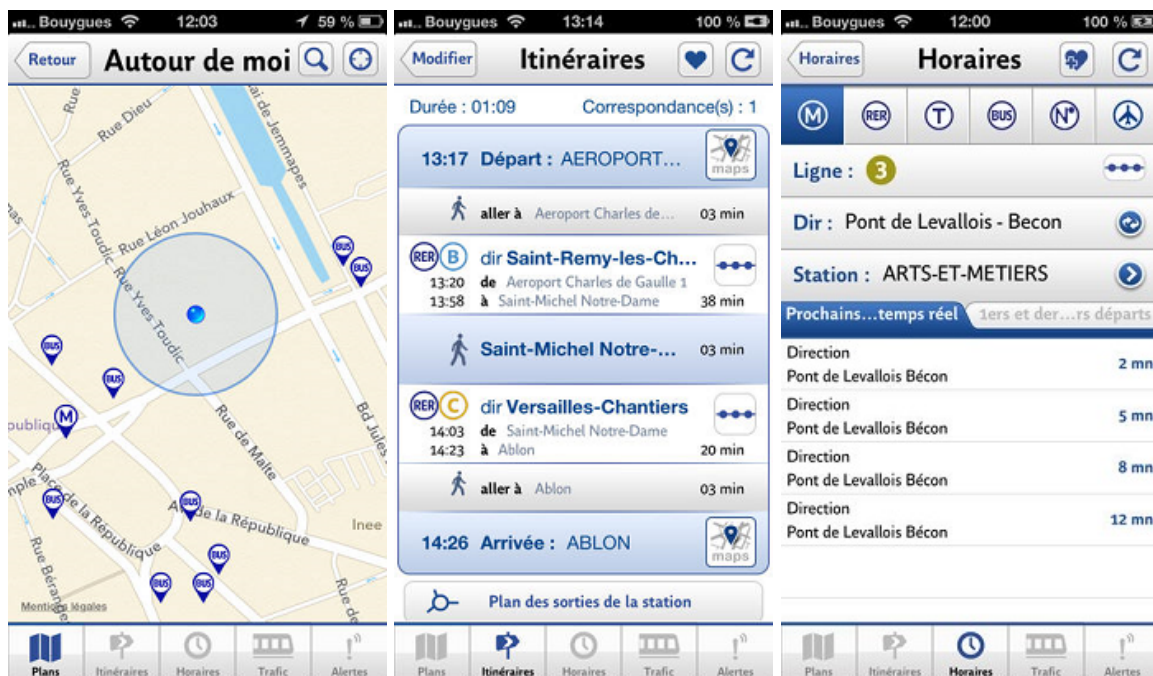
Pontos de destaque:

- Sistema de favoritos;
- Agregação de vários serviços de transporte;
- Definição de alertas;
- Acesso ao estado do trânsito.

Pontos negativos:

- Agregação de muita informação por ecrã o que torna um pouco confusa a sua leitura;

Interface:



(a) Mapa de Paragens

(b) Rotas

(c) Horários

Figura A.18: Interface da Aplicação RATP:Paris

¹⁷de Regie Autonome des Transports Parisien, disponível em <http://bit.ly/iRATPParis>

UK Bus Checker

Descrição: A aplicação UK Bus Checker¹⁸ está disponível apenas como versão paga. O principal objetivo desta é de efetuar a monitorização de percursos e tempos de chegada dos autocarros, em tempo real, nas diversas paragens do Reino Unido, de modo a auxiliar o utilizador nas suas deslocações diárias.

Pontos de destaque:

- Possibilidade de acesso a todos os autocarros existentes em determinada paragem e respetivos tempos de espera;
- Permite visualizar o mapa da rota de cada autocarro;
- Lista as diversas paragens para cada rota;
- Guarda favoritos de paragens;
- Acesso aos horários das várias rotas;
- Procura de paragens por perto;
- Disponibilização da paragens em lista ou no mapa.

Pontos negativos:

- Ter que pagar para usufruir de funcionalidades extra.

Interface:

¹⁸de FatAttitude Ltd, disponível em <http://www.buschecker.com/>

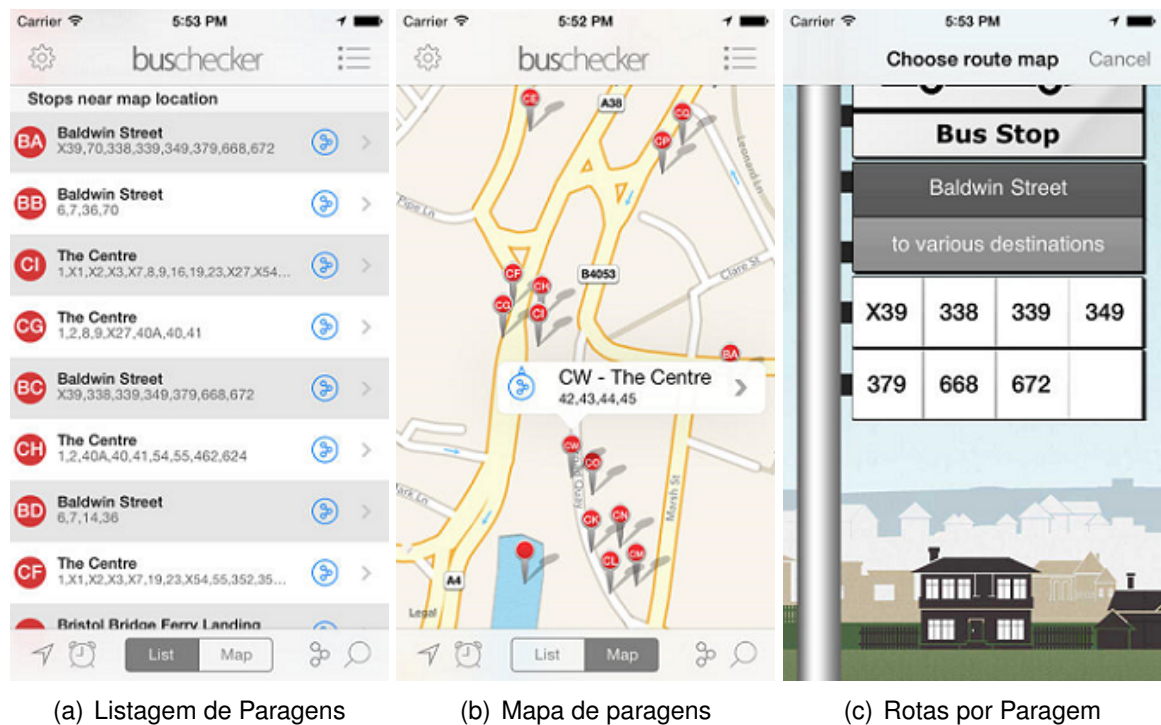


Figura A.19: Interface da Aplicação UK Bus Checker

MOVE-ME

Descrição: Também presente na plataforma Android, esta aplicação está disponível para iOS e é ideal para os utilizadores de transportes públicos da cidade de Lisboa e do Porto, uma vez que esta está presente para utilização nestas duas cidades. Apesar da interface não ser aquilo que se esperava, numa aplicação deste género e para esta plataforma, não deixa de estar funcional e cumprir aquilo a que se compromete, embora, o número total de funcionalidades da mesma deixe um pouco a desejar.

Pontos de destaque:

- Procura de todas as paragens existentes através da introdução de um raio de procura;
- Disponibilização de pontos de interesse;
- Possibilidade de utilização em duas grandes cidades.

Pontos negativos:

- Interface mal construída;
- Escassas funcionalidades.

A.1.3 Windows Phone

O Windows Phone¹⁹ é, de todas as plataformas neste capítulo apresentadas, a mais recente. Projetado e desenvolvido pela Microsoft para suceder ao existente e em crescente extinção Windows Mobile, é hoje um sistema móvel "único", isto é, tanto utilizado por empresas como por particulares, ao contrário do que se verificava com o seu antecessor. Foi lançado em Outubro de 2010 acompanhado de uma *interface* nova, a "Metro", e em nada a ver com os sistemas já existentes no mercado. Além disso, este novo sistema foi concebido para ser integrado com serviços Microsoft e de terceiros com o objectivo de ter o mínimo de requisitos de *hardware*, para que possa correr em todos os dispositivos.

Uma vez que a utilização desta plataforma tem vindo a crescer cada vez mais,[26][27] achou-se importante a inclusão das aplicações pertencentes a esta no presente estudo.

Countdown

Descrição: A aplicação Countdown²⁰ para a plataforma Windows Phone que fornece a informação sobre o tempo de partida e chegada dos autocarros em todas as paragens da cidade de Londres.

Pontos de destaque:

- Adição de favoritos;
- Disponibilização de um mapa de paragens próximas.

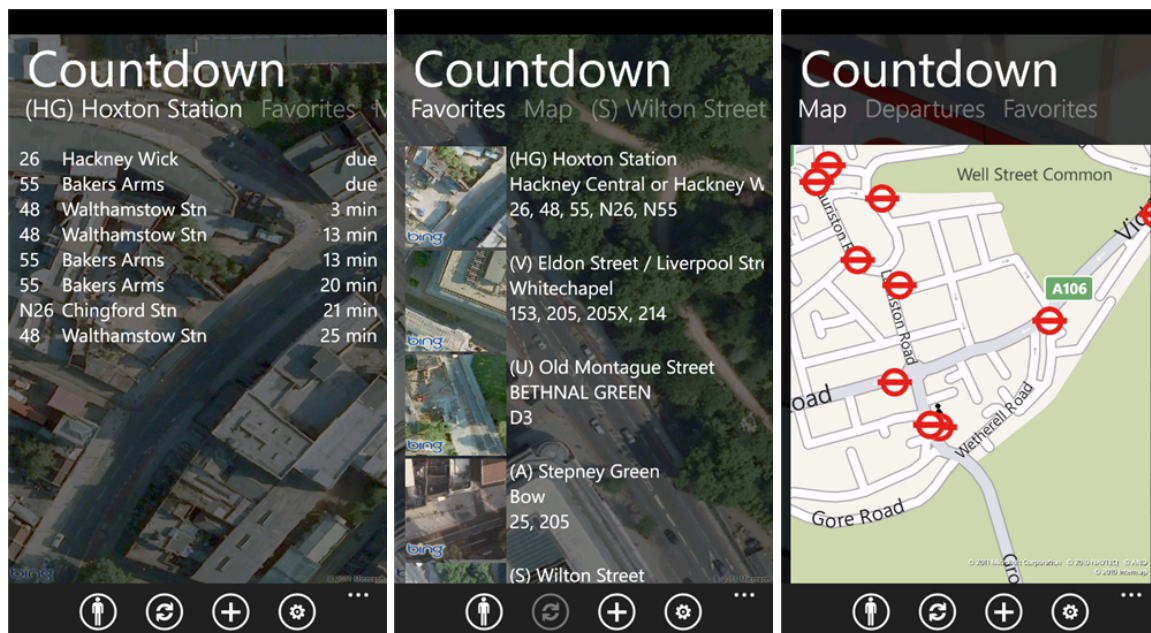
Pontos negativos:

- Mapa da localização como fundo.

Interface:

¹⁹de Microsoft, disponível em <http://www.windowsphone.com/pt-pt/>

²⁰de Web Methods Ltd., disponível em <http://bit.ly/WPCountdown>



(a) Tempo de espera para os próximos autocarros

(b) Sistema de Favoritos

(c) Mapa de Paragens

Figura A.20: Interface da Aplicação Countdown

Depois de analisada para a Android e também iOS, a RATP:Paris²¹ aparece agora disponível para Windows Phone com um aspeto um pouco diferente das versões já analisadas, mas com o mesmo número de funcionalidades. Contudo, os pontos negativos desta aplicação continuam os mesmos nas várias versões das diferentes plataformas, uma falha que, apesar de permitir a sua utilização diária, deveriam ser revistas de modo a proporcionar ao utilizador uma melhor experiência de utilização.

RATP: Paris

Descrição: Tal como na aplicação Android e iOS atrás apresentadas, a RATP: Paris versão Windows Phone tem os mesmos objetivos e são eles o auxílio dos seus utilizadores nas deslocações através de transportes públicos na cidade de Paris, em França.

Pontos de destaque:

- Procura de autocarro/rota por número;
- Sistema de favoritos;
- Horários de circulação;

²¹de Beguin Conseil, disponível em <http://bit.ly/RATPWP>

- Detalhe do tempo em cada paragem.

Pontos negativos:

- Informação "dispersa" pelos ecrãs.

Interface:



(a) Tempo para o Próximo Auto-carro

(b) Sistema de Favoritos

(c) Pesquisa de Autocarros

Figura A.21: Interface da Aplicação RATP

Buses Due

Descrição: Construída para ser utilizada na cidade de Londres, a aplicação Buses Due²² regista os tempos que faltam para os próximos autocarros, permitindo assim a que todos os seus utilizadores poupem algum tempo nas suas deslocações diárias.

Pontos de destaque:

- Sistema de favoritos com bastante informação útil;
- Possibilidade de seleção de uma paragem no mapa e ter acesso aos tempos para os próximos autocarros;
- Paragens com informação não só das horas no momento, como também do tempo que falta para os próximos autocarros;

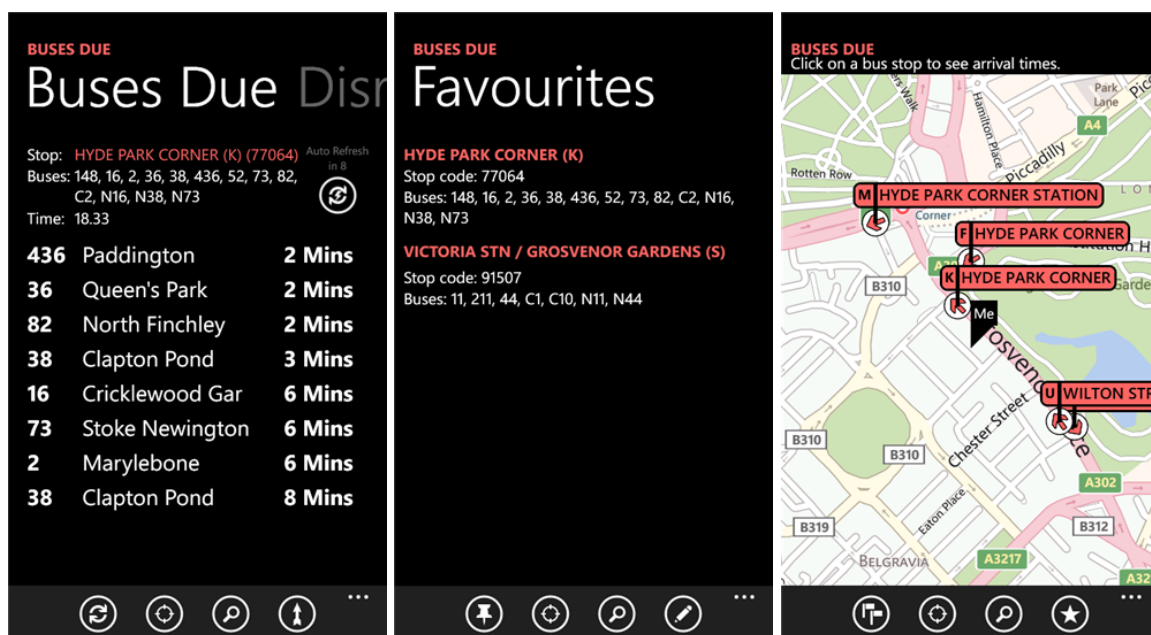
²²de Lama Apps, disponível em <http://bit.ly/BusesDue>

- Disponibilização de todos os autocarros para uma mesma paragem;
- Atualização automática dos tempos.

Pontos negativos:

- Por vezes, perde-se a noção de onde estamos na aplicação.

Interface:



(a) Tempo para o Próximo Auto-carro

(b) Sistema de Favoritos

(c) Mapa de Paragens

Figura A.22: Interface da Aplicação Buses Due

BH Buses

Descrição: BH Buses²³ é uma aplicação que permite aos seus utilizadores procurar por todas as rotas de autocarros da cidade de Brighton e Hove, no Reino Unido. Além desta funcionalidade, faz-se acompanhar ainda de uma secção de notícias, horários e apresenta os detalhes destas no mapa da aplicação.

Pontos de destaque:

- Apresentação do número das rotas com o destino e o tempo que falta;
- Possibilidade de rota por número;
- Listagem de paragens ordenada alfabeticamente;

²³de Netscape Limited, disponível em <http://bit.ly/BHBuses>

- As rotas com mesmo número são facilmente.

Pontos negativos:

- Esquema de cores da aplicação.

Interface:

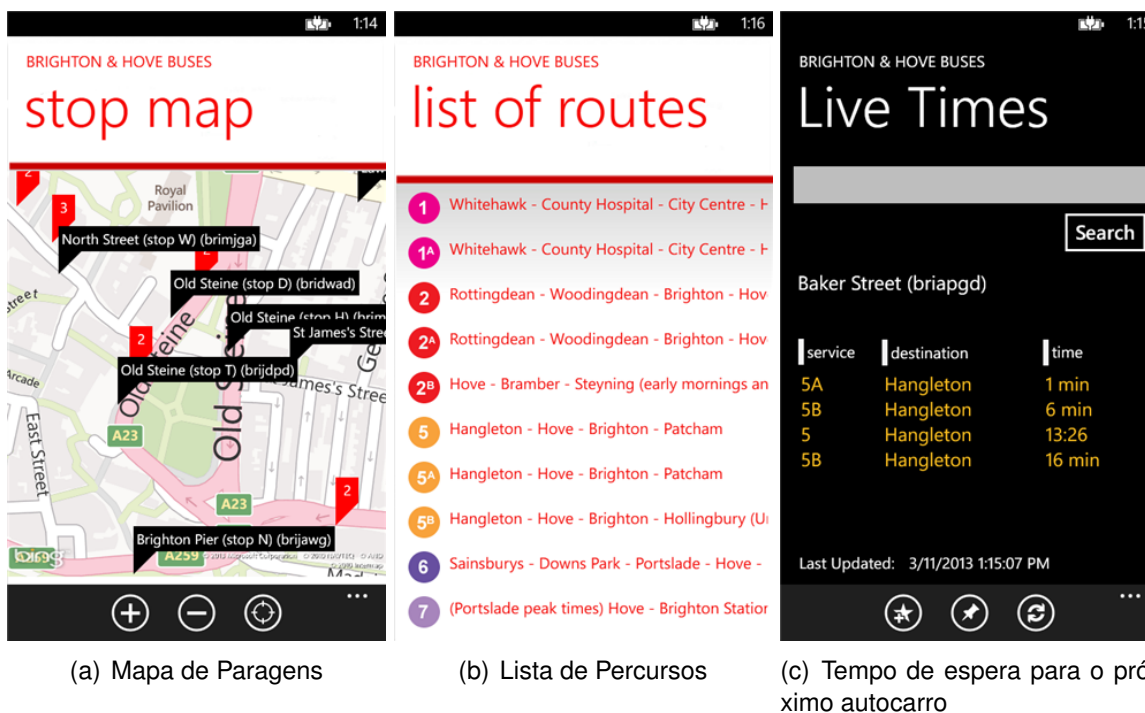


Figura A.23: Interface da Aplicação BH Buses

Deja-bus

Descrição: Déjà-bus²⁴ é uma aplicação para Windows Phone implementada para os utilizadores de transportes públicos da cidade de Monterrey, no México.

Pontos de destaque:

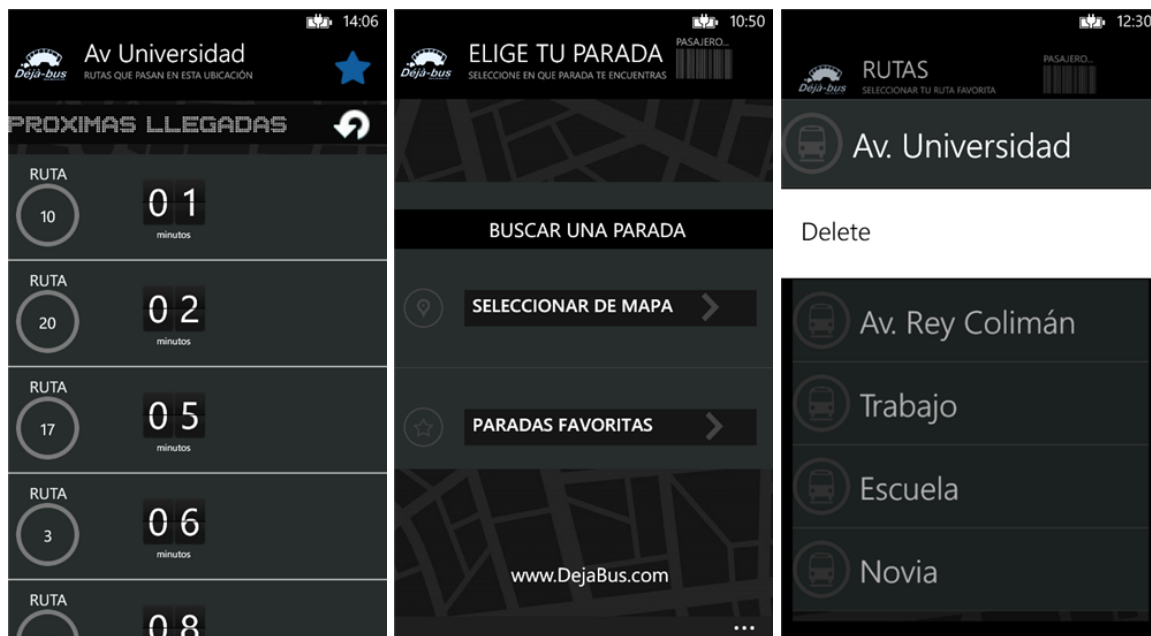
- Interface muito bem construída com indicação da rota e tempo de chegada do autocarro;
- Procura de paragens através do mapa;
- Adição de favoritos.

Pontos negativos:

- Atualização dos tempos de chegada é feita manualmente.

²⁴de dejabus, disponível em <http://bit.ly/DejaBus>

Interface:



(a) Tempo de Chegada dos Próximos Autocarros

(b) Procura de Paragens

(c) Percursos Favoritos

Figura A.24: Interface da Aplicação Déjà-bus

De todas as aplicações Windows Phone testadas, a próxima, Next Bus UK Live!²⁵ foi a que mais se destacou devido à sua aparência e também número e qualidade das funcionalidades que apresenta.

Next Bus UK Live!

Descrição: Tal como o próprio nome indica, esta aplicação foi desenvolvida com o intuito de auxiliar os utilizadores de transportes públicos do Reino Unido nas suas deslocações diárias.

Pontos de destaque:

- Interface muito bem construída;
- Disponibilização dos tempos de espera;
- A forma de interação com os favoritos;
- As várias formas de pesquisas.

²⁵de dans, disponível em <http://bit.ly/NextBusUK>

Pontos negativos:

- Não ser gratuita;

Interface:

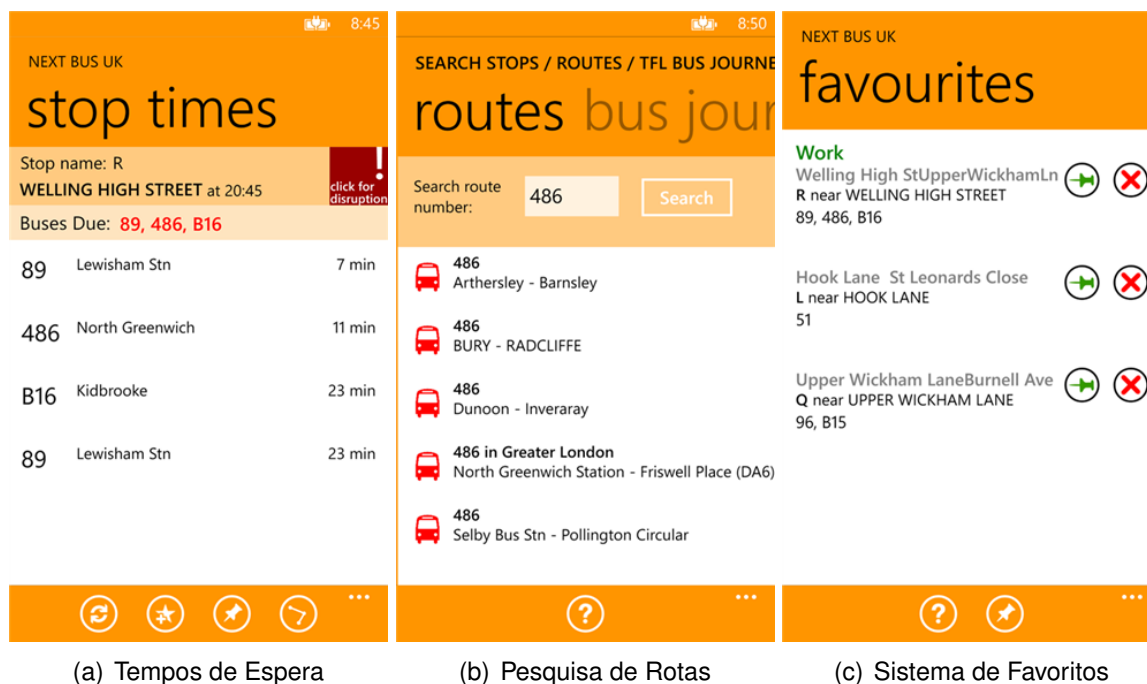


Figura A.25: Interface da Aplicação Next Bus UK Live!

UK Bus Checker

Descrição: Tal como a anterior, a aplicação UK Bus Checker²⁶ foi pensada e implementada a contar com os passageiros de autocarros do Reino Unido.

Pontos de destaque:

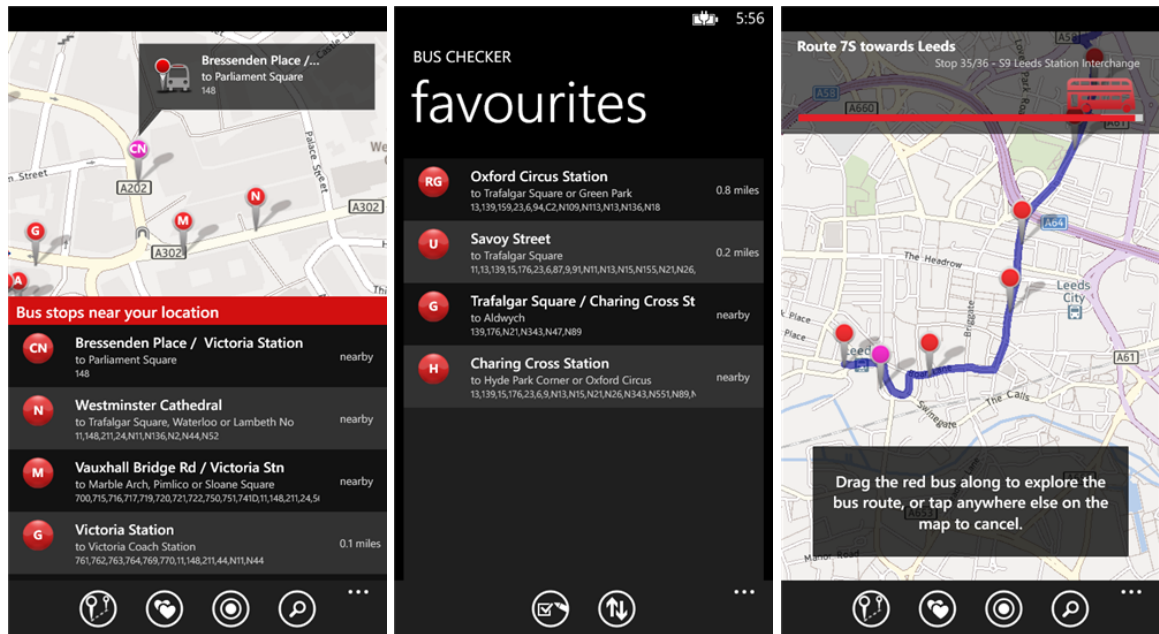
- Disponibilização das paragens mais próximas;
- Toda a interface gráfica;
- Sistema de procura;
- Favoritos.

Pontos negativos:

- A versão *lite* não possui o melhor da aplicação.

Interface:

²⁶de FatAttitude, disponível em <http://www.buschecker.com/?app=UK>



(a) Paragens mais próximas

(b) Sistema de Favoritos

(c) Rota no Mapa

Figura A.26: Interface da Aplicação London Bus Checker

Bus Nearby

Descrição: Para os passageiros de Dublin, na Irlanda, a Bus Nearby²⁷ auxilia os seus utilizadores nas deslocações, dando-lhes o tempo para o próximo autocarro, permitindo a consulta das paragens no mapa fazendo com que seja possível adicioná-las aos favoritos.

Pontos de destaque:

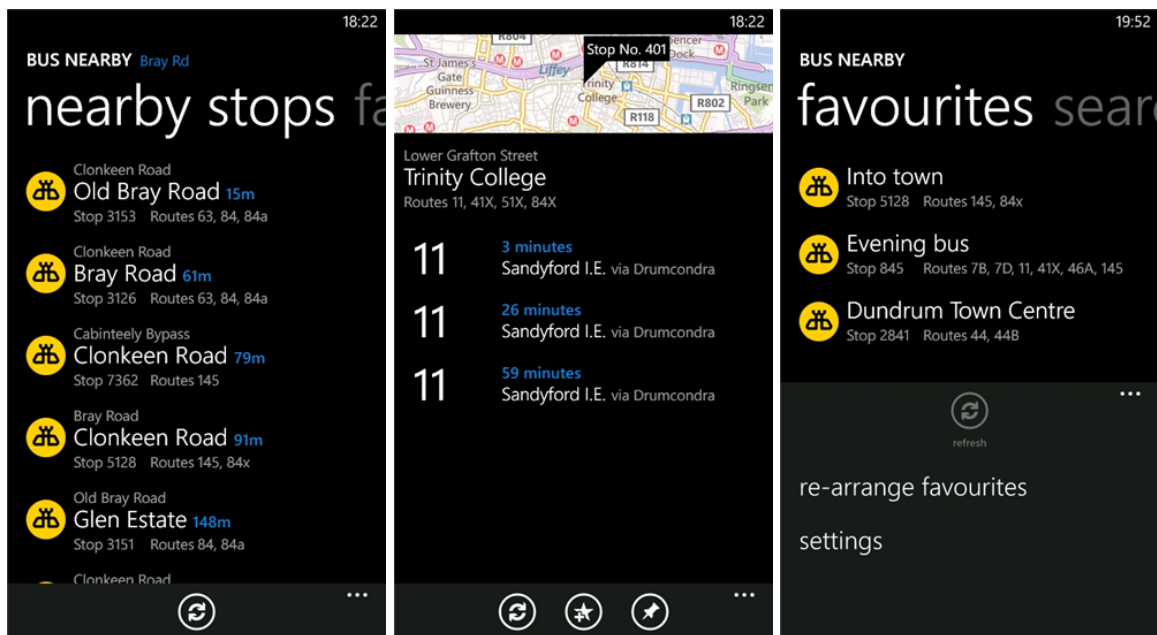
- Possibilidade de ordenação dos favoritos;
- Aspeto gráfico;
- Disponibilização do conteúdo no ecrã;

Pontos negativos:

- O mapa de paragens disponibilizado é de tamanho reduzido.

Interface:

²⁷de Neil Turner, disponível em <http://bit.ly/BusNearby>



(a) Paragens mais próximas

(b) Tempo de Espera

(c) Sistema de Favoritos

Figura A.27: Interface da Aplicação Bus Nearby

A.1.4 Análise Geral

Após a análise de um conjunto de aplicações existentes nas diversas lojas *online* chegou-se à conclusão que a maioria destas partilham um vasto número de funcionalidades, isto é, dado aquilo a que se propõem, conseguem ter, não só as funcionalidades em si, como também modos de funcionamento semelhantes. Isto é refletido nas três plataformas abordadas, uma vez que, apesar de serem diferentes todas assentam sobre os mesmos princípios de disponibilização de dados e interação com o utilizador, utilizando as mesmas *guidelines* e boas práticas no desenvolvimento de aplicações deste tipo. Abaixo, apresenta-se uma tabela resumo com as funcionalidades em comum resultante da análise do conjunto de aplicações selecionadas.

| | Funcionalidades | | | | | | |
|-------------------------|-------------------------------|-----------|-------------------------|-----------------------------------|----------------------|-----------------------------|---------|
| | Tempos de Espera nas paragens | Favoritos | Calculadora de percurso | Visualizar paragens mais próximas | Pesquisa de Paragens | Visualizar Paragens no Mapa | Alertas |
| Live London Bus Tracker | ✓ | | ✓ | | ✓ | | |
| UK Bus Checker | ✓ | | | | ✓ | ✓ | |
| New Metro Madrid | ✓ | | ✓ | | | | |
| Bus K Fortaleza | | | | | | ✓ | ✓ |
| Bus Brazil | | ✓ | | | | | ✓ |
| Moobly | ✓ | ✓ | ✓ | ✓ | ✓ | | ✓ |
| RATP: Paris | ✓ | ✓ | ✓ | | ✓ | ✓ | ✓ |
| IZI Carris | ✓ | ✓ | | | ✓ | | ✓ |
| Sapo Transportes | | ✓ | ✓ | | | | ✓ |
| NUS NextBus | ✓ | | | ✓ | | ✓ | |
| Live London Countdown | ✓ | ✓ | ✓ | | ✓ | | |
| NextBuses | ✓ | ✓ | | ✓ | ✓ | | |
| iNextBus | ✓ | ✓ | | | | | |
| InTime NextBus | ✓ | ✓ | | | | ✓ | ✓ |
| SG NextBus | ✓ | | | ✓ | | ✓ | |
| NUS NextBus | ✓ | | | ✓ | | ✓ | |
| Countdown | ✓ | ✓ | | ✓ | | ✓ | |
| Buses Due | ✓ | ✓ | | | | | |
| BH Buses | ✓ | | | | | | |
| Deja-Bus | ✓ | ✓ | | ✓ | ✓ | ✓ | |
| Next Bus UK Live! | ✓ | ✓ | | | ✓ | ✓ | |
| UK Bus Checker | ✓ | ✓ | ✓ | | ✓ | ✓ | |
| Bus Nearby | ✓ | ✓ | | ✓ | ✓ | | |
| MOVE-ME | ✓ | | ✓ | ✓ | ✓ | ✓ | |

| | | Funcionalidades | | | | | | |
|----------------|-------------------------|-------------------------------|----------------------------|-------------------|------------------------------|-------------------------|----------|------------------------------------|
| | | Horários e Preços das Viagens | Vários Tipos de Transporte | Pesquisa por Rota | Visualizar Paragens por Rota | Visualizar Rota no Mapa | Notícias | Pesquisa do autocarro mais próximo |
| Aplicações | Live London Bus Tracker | | | | | ✓ | | |
| | UK Bus Checker | | | | ✓ | ✓ | | |
| | New Metro Madrid | ✓ | ✓ | ✓ | | | | ✓ |
| | Bus K Fortaleza | ✓ | | | | | ✓ | ✓ |
| | Bus Brazil | ✓ | | | | ✓ | | |
| | Moobly | ✓ | ✓ | | | ✓ | ✓ | |
| | RATP: Paris | | ✓ | | | | | |
| | IZI Carris | | | ✓ | ✓ | | | |
| | Sapo Transportes | ✓ | ✓ | ✓ | ✓ | | | |
| | NUS NextBus | | | | | | | |
| | Live London Countdown | | | ✓ | | | | |
| | NextBuses | | | | | | | |
| | iNextBus | | | | | ✓ | | |
| | InTime NextBus | | | | | ✓ | | |
| | SG NextBus | | | | | | | |
| | NUS NextBus | | | | | | | |
| | Countdown | | | | | | | |
| | Buses Due | | | ✓ | ✓ | | | |
| | Deja-Bus | | | | | | | |
| | Next Bus UK Live! | | | ✓ | ✓ | ✓ | | |
| UK Bus Checker | | | | ✓ | ✓ | | | |
| Bus Nearby | | | | | | | | |
| MOVE-ME | | | | ✓ | ✓ | | | |

Tabela A.2: Outras funcionalidades

Tal como na tabela 2.1, a tabela 2.2 é apresentada na ordem sobre as quais foram testadas e documentadas as aplicações. Assim, nesta tabela constam todas as funcionalidades que apesar de serem importantes para a solução que se propõe, são pouco comuns no conjunto de aplicações testadas. Dessas funcionalidades destacam-se a possibilidade de visualizar um rota no mapa ou ainda a visualização do conjunto de paragens nessa mesma rota. Além destas, a disponibilização de uma secção de notícias e de informações de viagem como por exemplo horários e preços de viagens também são opções viáveis quando o objetivo é a construção de uma aplicação o mais completa possível.

Estatísticas de Utilização

A análise das aplicações permite retirar várias conclusões, não só acerca das funcionalidades mais utilizadas e mais ou menos comuns, como também acerca de vários factores que estão indiretamente relacionados com estas mesmas aplicações. Estes factores prendem-se com a característica do negócio implícita nas mesmas. A seguinte tabela apresenta as aplicações, o seu rendimento, se gratuita ou paga, a plataforma a que pertence, a classificação dada pelos seus utilizadores (calculada pelo número de utilizadores que deram *feedback*) e ainda a compatibilidade com os diversos sistemas operativos, dentro da mesma plataforma.

Tabela A.3: Estatísticas das aplicações analisadas

| Aplicação | Gratuita | Plataforma | Rating | Compatibilidade |
|-------------------------|-----------------|-------------------|---------------|------------------------|
| UK Bus Checker | Não | Android | 4/5 (115) | 2.2 - Atual |
| London Live Bus Tracker | Sim | Android | 4,5/5 (7686) | 1.6 - Atual |
| New Madrid Metro Bus | Sim | Android | 4,5/5 (451) | 2.2 - Atual |
| Bus K Fortaleza | Sim | Android | 4,7/5 (195) | 2.2 - Atual |
| Bus Brazil | Sim | Android | 4,8/5 (5) | 2.2 - Atual |
| Moobly | Sim | Android | 4,7/5 (686) | 1.6 - Atual |
| RATP: Paris | Sim | Android | 3,6/5 (6 960) | 2.1 - Atual |

Continua na próxima página

Tabela A.3 – em continuação da página anterior

| Aplicação | Gratuita | Plataforma | Rating | Compatibilidade |
|---------------------------|-----------------|-------------------|-------------------------|------------------------|
| IZI Carris | Sim | Android | 3,3/5 ⁽²¹⁴⁾ | 2.1 - Atual |
| Sapo Transportes | Sim | Android | 3,0/5 ⁽⁹³⁾ | 2.2 - Atual |
| NUS NextBus | Sim | Android | 2,3/5 ⁽³⁸⁾ | 2.2 - Atual |
| MOVE-ME | Sim | Android | 3,2/5 ⁽⁷⁶⁾ | 2.2 - Atual |
| London Bus Live Countdown | Sim | iOS | N/D | 4.3 - Atual |
| NextBuses | Não | iOS | 3,0/5 ⁽¹²¹⁴⁾ | 6.0 - Atual |
| iNextBus | Não | iOS | N/D | 5.0 - Atual |
| InTime NextBus | Não | iOS | N/D | 4.3 - Atual |
| SG NextBus | Sim | iOS | N/D | 5.0 - Atual |
| NUS NextBus | Sim | iOS | N/D | 5.1 - Atual |
| RATP: Paris | Sim | iOS | 3.5/5 ⁽¹⁹⁾ | 4.3 - Atual |
| UK Bus Checker | Não | iOS | 4/5 ⁽²²⁴⁾ | 5.0 - Atual |
| MOVE-ME | Sim | iOS | 2.5/5 ⁽²²⁾ | 4.3 - Atual |
| Countdown | Sim | Windows Phone | 3/5 ⁽²⁾ | 7.5 - 8 |
| Deja -Bus | Sim | Windows Phone | N/D | 7.5 - 8 |
| RATP: Paris | Sim | Windows Phone | 5/5 ⁽¹⁾ | 7.5 - 8 |
| Buses Due | Sim | Windows Phone | 2,5/5 ⁽²⁾ | 7.5 - 8 |
| BH Buses | Sim | Windows Phone | 2/5 ⁽¹⁾ | 7.5 - 8 |
| Next Bus UK Live! | Não | Windows Phone | N/D | 7.5 - 8 |
| UK Bus Checker | Não | Windows Phone | 4/5 ⁽⁷⁰⁾ | 7.5 - 8 |

Continua na próxima página

Tabela A.3 – em continuação da página anterior

| Aplicação | Gratuita | Plataforma | Rating | Compatibilidade |
|------------------|-----------------|-------------------|---------------|------------------------|
| Bus Nearby | Sim | Windows Phone | N/D | 7.5 - 8 |

Da tabela 2.3, ordenada pela ordem em que foram analisadas as aplicações, conseguem obter-se diversas conclusões. Começando pelo número de aplicações analisadas pertencentes às diversas *stores* consegue perceber-se a dimensão das mesmas, isto é, destacando-se a Play, da Google e a App Store, da Apple, com o maior número de aplicações presentes para o tema em questão, enquanto que no que respeita à Windows Store, da Microsoft, a dificuldade em encontrar aplicações do género foi mais elevada, muito devido ao facto desta ser a mais recente das três. [28] Quanto ao número de aplicações pagas analisadas os números não deixam dúvida que das seleccionadas para iOS, metade não são gratuitas. No que toca ao sistema Android, apenas uma das aplicações não é paga mas apresenta uma versão Lite que, embora com um número de funcionalidades mais reduzido, é gratuita.

Apêndice B

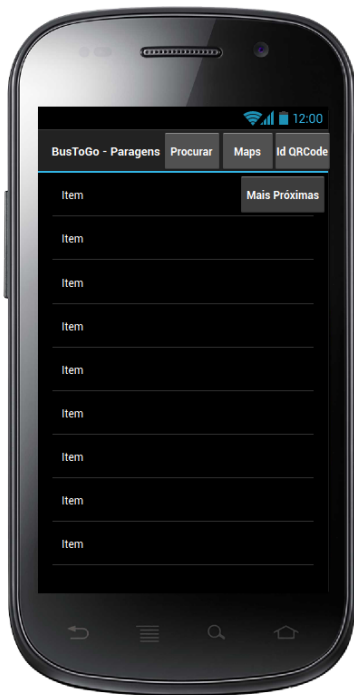
Protótipos da Solução



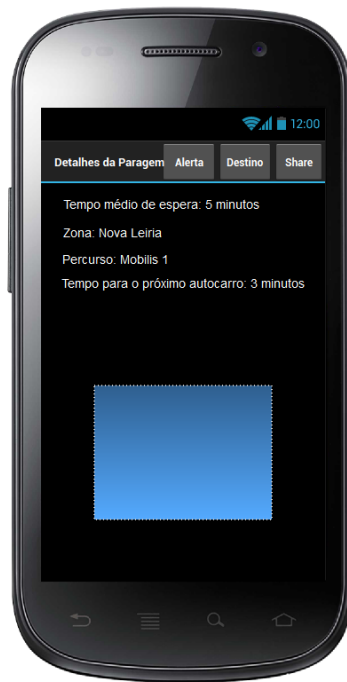
(a) Barra de Acção

(b) Menu Principal

(c) Cálculo de Percurso



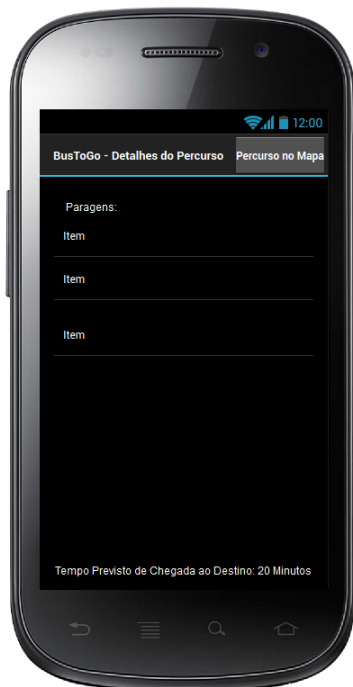
(d) Paragens



(e) Detalhes de Paragens



(f) Percurso



(g) Detalhes de Percurso



(h) Preferências

Apêndice C

Especificação GTFS

Tabela C.1: *agency.txt*

| Nome | Descrição | Obrigatório |
|------------------------|---|-------------|
| <i>agency_id</i> | Identificador único de cada agência de transporte. É opcional para <i>feeds</i> com dados de uma agência apenas. | Não |
| <i>agency_name</i> | Nome da agência de transporte. | Sim |
| <i>agency_url</i> | URL da agência de transporte. Deverá ser completo, isto é, contendo o protocolo (http/https) associado ao mesmo. | Sim |
| <i>agency_timezone</i> | Fuso horário da agência de transporte. Deverá ser definido de acordo com especificação: http://goo.gl/STCxIG | Sim |
| <i>agency_lang</i> | Código ISO 639-1 de duas letras que identificam o idioma utilizado pela empresa de transportes. | Não |
| <i>agency_phone</i> | Número de telefone da empresa | Não |

Continua na próxima página

Tabela C.1 – em continuação da página anterior

| Nome | Descrição | Obrigatório |
|------------------------|---|--------------------|
| <i>agency_fare_url</i> | URL que direciona o passageiro para a compra dos bilhetes de transporte. Deverá ser completo, isto é, contendo o protocolo (http/https) associado ao mesmo. | Não |

Tabela C.2: stops.txt

| Nome | Descrição | Obrigatório |
|-----------------------|--|--------------------|
| <i>stop_id</i> | Identificador único de cada paragem. | Sim |
| <i>stop_code</i> | Texto que identifica uma paragem. | Não |
| <i>stop_name</i> | Nome da paragem. | Sim |
| <i>stop_desc</i> | Descrição de uma paragem. | Não |
| <i>stop_lat</i> | Latitude de uma paragem. | Sim |
| <i>stop_long</i> | Longitude de uma paragem. | Não |
| <i>zone_id</i> | Define a zona da tarifa de uma paragem. | Não |
| <i>stop_url</i> | URL com informação de uma paragem específica. Deverá ser completo, isto é, contendo o protocolo (http/https) associado ao mesmo. | Não |
| <i>location_type</i> | Tipo de localização. Se for a 0 ou em branco: paragem. Caso seja 1 significa que é um terminal rodoviário. | Não |
| <i>parent_station</i> | Caso esteja num terminal rodoviário, identifica a "paragem-mãe" ao qual está associada. | Não |

Continua na próxima página

Tabela C.2 – em continuação da página anterior

| Nome | Descrição | Obrigatório |
|----------------------------|---|--------------------|
| <i>stop_timezone</i> | Fuso horário sob o qual a paragem está inserida. | Não |
| <i>wheelchair_boarding</i> | Identifica se é possível ou não o embarque de passageiros em cadeira de rodas. 0 ou vazio - Sem informações. 1 - Indica que alguns veículos permitem o embarque de passageiros nesta paragem. 2 - Não é possível o embarque de pessoas em cadeira de rodas. | Não |

Tabela C.3: routes.txt

| Nome | Descrição | Obrigatório |
|-------------------------|--|--------------------|
| <i>route_id</i> | Identificador único de cada rota. | Sim |
| <i>agency_id</i> | Identificador da agência de transporte associada à rota. | Não |
| <i>route_short_name</i> | Nome abreviado de cada rota. | Sim |
| <i>route_long_name</i> | Nome completo de uma rota. | Sim |
| <i>route_desc</i> | Descrição de uma rota. | Não |
| <i>route_type</i> | Tipo de rota. 0 - Veículo sobre carris à superfície. 1 - Metro. 2 - Comboio. 3 - Autocarro. 4 - Barco. 5 - Teleférico de rua. 6 - Teleférico Suspenso. 7 - Ascensor. | Sim |
| <i>route_url</i> | URL com informação de uma rota específica. Deverá ser completo, isto é, contendo o protocolo (http/https) associado ao mesmo. | Não |

Continua na próxima página

Tabela C.3 – em continuação da página anterior

| Nome | Descrição | Obrigatório |
|-------------------------|---|--------------------|
| <i>route_color</i> | Cor de fundo associada à rota. Normalmente utilizada para ser representada no mapa. | Não |
| <i>route_text_color</i> | Cor de texto associada à rota. Normalmente utilizada para ser representada no mapa. | Não |

Tabela C.4: trips.txt

| Nome | Descrição | Obrigatório |
|------------------------|--|--------------------|
| <i>route_id</i> | Identificador da rota à qual a viagem pertence. | Sim |
| <i>service_id</i> | Identifica um conjunto de datas em que o serviço está disponível. | Sim |
| <i>trip_id</i> | Identificador da viagem. | Sim |
| <i>trip_headsign</i> | Descritivo do destino da viagem. | Não |
| <i>trip_short_name</i> | Nome de uma viagem. | Não |
| <i>direction_id</i> | Valor que identifica as direções das viagens. 0 - Viagem numa única direção. Só ida ou só volta. 1 - Viagem na direção oposta. Só ida ou só volta. | Não |
| <i>block_id</i> | Identifica a que quadro pertence a viagem, isto é, se o mesmo veículo pode fazer duas viagens sequenciais. | Não |
| <i>shape_id</i> | Forma da viagem. Valor definido no ficheiro <i>shapes.txt</i> | Não |

Continua na próxima página

Tabela C.4 – em continuação da página anterior

| Nome | Descrição | Obrigatório |
|------------------------------|---|--------------------|
| <i>wheelchair_accessible</i> | Identifica se é possível ou não o embarque de passageiros em cadeira de rodas. 0 ou vazio - Sem informações. 1 - Indica que alguns veículos permitem o embarque de passageiros nesta paragem. 2 - Não é possível o embarque de pessoas em cadeira de rodas. | Não |

Tabela C.5: stop_times.txt

| Nome | Descrição | Obrigatório |
|-----------------------|--|--------------------|
| <i>trip_id</i> | Identificador único da viagem. | Sim |
| <i>arrival_time</i> | Horário de chegada a uma paragem específica. | Sim |
| <i>departure_time</i> | Horário de partida de uma paragem específica. | Sim |
| <i>stop_id</i> | Identificador da paragem ao qual a viagem pertence. | Sim |
| <i>stop_sequence</i> | Ordem da paragem na sequência da rota. | Sim |
| <i>stop_headsign</i> | Identificação do destino da viagem para os passageiros. | Não |
| <i>pickup_type</i> | Tipo de embarque em cada paragem. 0 - Embarque normal. 1 - Sem embarque disponível. 2 - É necessário o contato com a agência para agendar embarque. 3 - Combinar com o motorista o embarque. | Não |

Continua na próxima página

Tabela C.5 – em continuação da página anterior

| Nome | Descrição | Obrigatório |
|----------------------------|--|--------------------|
| <i>drop_off_type</i> | Tipo de desembarque em cada paragem. 0 - Desembarque normal. 1 - Sem desembarque disponível. 2 - É necessário o contato com a agência para agendar desembarque. 3 - Combinar com o motorista o desembarque | Não |
| <i>shape_dist_traveled</i> | Quando definido o ficheiro <i>shapes.txt</i> , indica a distância do primeiro ponto ao ponto atual. | Não |

Tabela C.6: calendar.txt

| Nome | Descrição | Obrigatório |
|----------------------------|--|--------------------|
| <i>service_id</i> | Identificador do serviço. | Sim |
| <i>monday</i> | Indicador se o serviço é válido às segundas-feiras. 0 - Não está disponível em algumas. 1 - Disponível em todas. | Sim |
| <i>tuesday</i> | Indicador se o serviço é válido às terças-feiras. 0 - Não está disponível em algumas. 1 - Disponível em todas. | Sim |
| <i>wednesday</i> | Indicador se o serviço é válido às quartas-feiras. 0 - Não está disponível em algumas. 1 - Disponível em todas. | Sim |
| <i>thursday</i> | Indicador se o serviço é válido às quintas-feiras. 0 - Não está disponível em algumas. 1 - Disponível em todas. | Sim |
| Continua na próxima página | | |

Tabela C.6 – em continuação da página anterior

| Nome | Descrição | Obrigatório |
|-------------------|--|--------------------|
| <i>friday</i> | Indicador se o serviço é válido às sextas-feiras. 0 - Não está disponível em algumas. 1 - Disponível em todas. | Sim |
| <i>saturday</i> | Indicador se o serviço é válido aos sábados. 0 - Não está disponível em alguns. 1 - Disponível em todos. | Sim |
| <i>sunday</i> | Indicador se o serviço é válido aos domingos. 0 - Não está disponível em alguns. 1 - Disponível em todos. | Sim |
| <i>start_date</i> | Data de início do serviço. | Sim |
| <i>end_date</i> | Data de fim do serviço. | Sim |

Tabela C.7: calendar_dates.txt

| Nome | Descrição | Obrigatório |
|-----------------------|--|--------------------|
| <i>service_id</i> | Identificador do serviço. | Sim |
| <i>date</i> | Data em que a disponibilidade do serviço é diferente do normal. | Sim |
| <i>exception_type</i> | Tipo de exceção. 1 - O serviço foi adicionado para a data especificada. 2 - O serviço foi removido na data especificada. | Sim |

Tabela C.8: fare_attributes.txt

| Nome | Descrição | Obrigatório |
|--------------------------|--|--------------------|
| <i>fare_id</i> | Identificador de uma tarifa. | Sim |
| <i>price</i> | Preço de uma tarifa. | Sim |
| <i>currency_type</i> | Moeda usada para pagar a tarifa. | Sim |
| <i>payment_method</i> | Forma como a tarifa é paga. 0 - Tarifa paga a bordo. 1 - Tarifa paga antes do embarque. | Sim |
| <i>transfers</i> | Número de transbordos possíveis com uma tarifa. 0 - Não são permitidos transbordos com esta tarifa. 1 - Apenas é possível um transbordo. 2 - É possível efetuar dois transbordos. A vazio, número ilimitado de transbordos com uma tarifa. | Sim |
| <i>transfer_duration</i> | Duração, em segundos, até o final de uma tarifa ativa. | Não |

Tabela C.9: fare_rules.txt

| Nome | Descrição | Obrigatório |
|----------------------------|--|--------------------|
| <i>fare_id</i> | Identificador de uma tarifa. | Sim |
| <i>route_id</i> | Rota sob a qual a tarifa é aplicada. | Não |
| <i>origin_id</i> | Identificador da zona de origem sob a qual a tarifa é aplicada. | Não |
| <i>destination_id</i> | Identificador da zona de destino sob a qual a tarifa é aplicada. | Não |
| Continua na próxima página | | |

Tabela C.9 – em continuação da página anterior

| Nome | Descrição | Obrigatório |
|--------------------|--------------------------------|--------------------|
| <i>contains_id</i> | Associa uma tarifa a uma zona. | Não |

Tabela C.10: shapes.txt

| Nome | Descrição | Obrigatório |
|----------------------------|---|--------------------|
| <i>shape_id</i> | Identificador de uma forma. | Sim |
| <i>shape_pt_lat</i> | Latitude de uma forma. | Sim |
| <i>shape_pt_lon</i> | Longitude de uma forma. | Sim |
| <i>shape_pt_sequence</i> | Sequência de uma forma num conjunto de formas. | Sim |
| <i>shape_dist_traveled</i> | Distância percorrida desde o início de uma forma ao ponto atual da mesma. | Não |

Tabela C.11: frequencies.txt

| Nome | Descrição | Obrigatório |
|----------------------------|--|--------------------|
| <i>trip_id</i> | Identificador da viagem ao qual a frequência se aplica. | Sim |
| <i>start_time</i> | Hora de início do serviço com a frequência especificada. | Sim |
| Continua na próxima página | | |

Tabela C.11 – em continuação da página anterior

| Nome | Descrição | Obrigatório |
|---------------------|--|--------------------|
| <i>end_time</i> | Hora de fim do serviço com a frequência especificada. | Sim |
| <i>headway_secs</i> | Tempo, em segundos, durante o qual a frequência está "ativa". | Sim |
| <i>exact_times</i> | Indicador se as viagens baseadas em frequência são programadas com exatidão. 0 - Não são programadas com exatidão. 1 - Programadas com exatidão. | Não |

Tabela C.12: transfers.txt

| Nome | Descrição | Obrigatório |
|--------------------------|--|--------------------|
| <i>from_stop_id</i> | Identificador da paragem onde se inicia a ligação de rotas. | Sim |
| <i>to_stop_id</i> | Identificador da paragem onde termina a ligação de rotas. | Sim |
| <i>transfer_type</i> | Tipo de transbordo possível. 0 - Ponto recomendado para efetuar o transbordo. 1 - Ponto de transbordo entre rotas. 2 - Tempo de espera necessário entre transbordos. 3 - Não existe transbordo possível. | Sim |
| <i>min_transfer_time</i> | Tempo mínimo de espera até ser feito o transbordo. | Não |

Tabela C.13: *feed_info.txt*

| Nome | Descrição | Obrigatório |
|----------------------------|---|--------------------|
| <i>feed_publisher_name</i> | Nome completo da entidade que publica o <i>feed</i> . | Sim |
| <i>feed_publisher_url</i> | Site da entidade que publica o <i>feed</i> . | Sim |
| <i>feed_lang</i> | Idioma padrão do texto utilizado no <i>feed</i> . | Sim |
| <i>feed_start_date</i> | Data de início a partir do qual o <i>feed</i> é válido. | Não |
| <i>feed_end_date</i> | Data de fim da validade do <i>feed</i> . | Não |
| <i>feed_version</i> | Versão atual do <i>feed</i> . | Não |

Apêndice D

GTFS - Realtime

Tabela D.1: FeedMessage

| Nome | Descrição | Tipo |
|---------------|--------------------------------------|----------------------|
| <i>header</i> | Informação e mensagem do <i>feed</i> | Elemento Obrigatório |
| <i>entity</i> | Conteúdo do <i>feed</i> | Elemento Repetido |

Tabela D.2: FeedMessage

| Nome | Descrição | Tipo |
|------------------------------|---|----------------------|
| <i>gtfs_realtime_version</i> | Versão do <i>feed</i> | Elemento Obrigatório |
| <i>incrementality</i> | Versão a incrementar do <i>feed</i> | Elemento Opcional |
| <i>timestamp</i> | Hora a que o conteúdo do <i>feed</i> foi criado | Elemento Opcional |

Tabela D.3: FeedHeader

| Nome | Descrição | Tipo |
|------------------------------|--|----------------------|
| <i>gtfs_realtime_version</i> | Versão do <i>feed</i> | Elemento Obrigatório |
| <i>incrementality</i> | Enumeração com os valores <i>FULL_DATASET</i> que substitui todos os dados do <i>feed</i> anterior ou <i>DIFFERENTIAL</i> que apenas substitui dados parciais num futuro próximo (ainda não implementado). | Elemento Opcional |
| <i>timestamp</i> | Hora a que o conteúdo do <i>feed</i> foi criado | Elemento Opcional |

Tabela D.4: FeedEntity

| Nome | Descrição | Tipo |
|--------------------|---|----------------------|
| <i>id</i> | Identificador exclusivo do <i>feed</i> para esta entidade. | Elemento Obrigatório |
| <i>is_deleted</i> | Booleano que indica se uma entidade deve ou não ser excluída. | Elemento Opcional |
| <i>trip_update</i> | Informação acerca dos atrasos em tempo real. | Elemento Opcional |
| <i>vehicle</i> | Dados sobre a posição do veículo em tempo real. | Elemento Opcional |

Continua na próxima página

Tabela D.4 – em continuação da página anterior

| Nome | Descrição | Tipo |
|--------------|--|-------------------|
| <i>alert</i> | Informação sobre os alertas em tempo real. | Elemento Opcional |

Tabela D.5: TripUpdate

| Nome | Descrição | Tipo |
|-------------------------|--|----------------------|
| <i>trip</i> | Identificador da viagem à qual o elemento se aplica. | Elemento Obrigatório |
| <i>vehicle</i> | Informação sobre o veículo que opera a viagem. | Elemento Opcional |
| <i>stop_time_update</i> | Atualização de viagem. | Elemento Opcional |
| <i>timestamp</i> | Momento em que o tempo real do veículo é medido. | Elemento Opcional |

Tabela D.6: StopTimeEvent

| Nome | Descrição | Tipo |
|----------------------------|---|-------------------|
| <i>delay</i> | O atraso do veículo. Positivo se o veículo estiver realmente atrasado. Negativo se estiver adiantado em relação ao horário. | Elemento Opcional |
| Continua na próxima página | | |

Tabela D.6 – em continuação da página anterior

| Nome | Descrição | Tipo |
|--------------------|-------------------------------|-------------------|
| <i>time</i> | Horário do evento. | Elemento Opcional |
| <i>uncertainty</i> | Previsão do atraso do evento. | Elemento Opcional |

Tabela D.7: StopTimeUpdate

| Nome | Descrição | Tipo |
|------------------------------|--|-------------------|
| <i>stop_sequence</i> | Valor da sequência da paragem definida no ficheiro <i>stop_times.txt</i> | Elemento Opcional |
| <i>stop_id</i> | Identificador da paragem que está definida no ficheiro <i>stops.txt</i> | Elemento Opcional |
| <i>arrival</i> | Horário de chegada. | Elemento Opcional |
| <i>departure</i> | Horário de partida. | Elemento Opcional |
| <i>schedule_relationship</i> | Relação de partida. <i>Scheduled</i> em que o veículo segue o programação definida. <i>Skipped</i> A paragem é ignorada. <i>No_Data</i> Sem informação para a paragem. | Elemento Opcional |

Tabela D.8: *VehiclePosition*

| Nome | Descrição | Tipo |
|------------------------------|--|-------------------|
| <i>trip</i> | Viagem que o veículo está a efetuar. | Elemento Opcional |
| <i>vehicle</i> | Informações acerca do veículo que está a realizar a viagem. | Elemento Opcional |
| <i>position</i> | Posição atual do veículo. | Elemento Opcional |
| <i>current_stop_sequence</i> | O índice de sequencia da paragem atual. | Elemento Opcional |
| <i>stop_id</i> | Identificador da paragem atual. | Elemento Opcional |
| <i>current_status</i> | O estado atual do veículo face à paragem. | Elemento Opcional |
| <i>timestamp</i> | Momento atual da posição do veículo. | Elemento Opcional |
| <i>congestion_level</i> | Estado da congestão do veículo. <i>Incoming_At</i> O veículo a chegar à paragem. <i>Stopped_At</i> O veículo está na paragem. <i>In_Transit_To</i> O veículo está em trânsito. | Elemento Opcional |

Tabela D.9: *VehiclePosition*

| Nome | Descrição | Tipo |
|------------------------------|---|-------------------|
| <i>trip</i> | Viagem que o veículo está a efetuar. | Elemento Opcional |
| <i>vehicle</i> | Informações acerca do veículo que está a realizar a viagem. | Elemento Opcional |
| <i>position</i> | Posição atual do veículo. | Elemento Opcional |
| <i>current_stop_sequence</i> | O índice de sequencia da paragem atual. | Elemento Opcional |
| <i>stop_id</i> | Identificador da paragem atual. | Elemento Opcional |
| <i>current_status</i> | O estado atual do veículo face à paragem. <i>Incoming_At</i> O veículo a chegar à paragem. <i>Stopped_At</i> O veículo está na paragem. <i>In_Transit_To</i> O veículo está em trânsito. | Elemento Opcional |
| <i>timestamp</i> | Momento atual da posição do veículo. | Elemento Opcional |
| <i>congestion_level</i> | Estado da congestão do trânsito. <i>Unknown_Congestion_Level</i> - Estado de trânsito desconhecido. <i>Running_Smoothly</i> - A correr normalmente. <i>Stop_And_Go</i> - "Pára-Arranca". <i>Congestion</i> - Congestionado. <i>Severe_Congestion</i> - Muito congestionado. | Elemento Opcional |

Tabela D.10: Alert

| Nome | Descrição | Tipo |
|------------------------|---|-------------------|
| <i>active_period</i> | Horário durante o qual o horário deve ser mostrado. | Elemento Repetido |
| <i>informed_entity</i> | Entidades a informar sobre o alerta. | Elemento Repetido |
| <i>cause</i> | Elemento causador do alerta. <i>Unknown_Cause</i> - Causa desconhecida. <i>Other_Cause</i> - Outra Causa. <i>Technical_Problem</i> - Problema Técnico. <i>Strike</i> - Greve. <i>Demonstration</i> - Protesto. <i>Accident</i> - Acidente. <i>Holiday</i> - Férias. <i>Weather</i> - Problema meteorológico. <i>Maintenance</i> - Manutenção. <i>Construction</i> - Construção. <i>Police_Activity</i> - Atividade Policial. <i>Medical_Emergency</i> - Emergência Médica | Elemento Opcional |
| <i>effect</i> | Efeito do alerta <i>No_Service</i> - Sem serviço. <i>Reduced_Service</i> - Serviço Reduzido. <i>Significant_Delays</i> - Atrasos significativos. <i>Detour</i> - Desvio. <i>Adicional_Service</i> - Serviço Adicional. <i>Modified_Service</i> - Serviço alterado. <i>Other_Effect</i> - Outro Efeito. <i>Unknown_Effect</i> - Efeito Desconhecido. <i>Stop_Moved</i> - Paragem modificada. | Elemento Opcional |
| <i>url</i> | URL que fornece mais informação acerca do alerta. | Elemento Opcional |
| <i>header_text</i> | Cabeçalho do alerta. | Elemento Opcional |

Continua na próxima página

Tabela D.10 – em continuação da página anterior

| Nome | Descrição | Tipo |
|-------------------------|----------------------|-------------------|
| <i>description_text</i> | Descrição do alerta. | Elemento Opcional |

Tabela D.11: TimeRange

| Nome | Descrição | Tipo |
|--------------|--------------------|-------------------|
| <i>start</i> | Horário de início. | Elemento Opcional |
| <i>end</i> | Horário de fim. | Elemento Opcional |

Tabela D.12: Position

| Nome | Descrição | Tipo |
|----------------------------|---|----------------------|
| <i>latitude</i> | Latitude da Posição. | Elemento Obrigatório |
| <i>longitude</i> | Longitude da Posição. | Elemento Obrigatório |
| <i>bearing</i> | Direção em graus. (Norte, Sul, Este e Oeste). | Elemento Opcional |
| <i>odometer</i> | Distância percorrida. | Elemento Opcional |
| Continua na próxima página | | |

Tabela D.12 – em continuação da página anterior

| Nome | Descrição | Tipo |
|--------------|------------------------|-------------------|
| <i>speed</i> | Velocidade do veículo. | Elemento Opcional |

Tabela D.13: TripDescriptor

| Nome | Descrição | Tipo |
|------------------------------|---|-------------------|
| <i>trip_id</i> | Identificador da viagem. | Elemento Opcional |
| <i>route_id</i> | Identificador da rota. | Elemento Opcional |
| <i>start_time</i> | Horas de início da viagem. | Elemento Opcional |
| <i>start_date</i> | Data inicial da viagem. | Elemento Opcional |
| <i>schedule_relationship</i> | Relação entre a viagem e a programação da mesma. <i>Scheduled</i> - Viagem de acordo com a programação. <i>Added</i> - Viagem adicionada às já existentes. <i>Unscheduled</i> - Sem programação de viagem. <i>Canceled</i> - Viagem eliminada da programação. <i>Replacement</i> - Uma viagem substitui parte da programação. | Elemento Opcional |

Tabela D.14: *VehicleDescriptor*

| Nome | Descrição | Tipo |
|----------------------|--|-------------------|
| <i>id</i> | Identificador do veículo. | Elemento Opcional |
| <i>label</i> | Informação sobre o veículo mostrada ao utilizador. | Elemento Opcional |
| <i>license_plate</i> | Matrícula do veículo. | Elemento Opcional |

Tabela D.15: *EntitySelector*

| Nome | Descrição | Tipo |
|-------------------|---------------------------------------|-------------------|
| <i>agency_id</i> | Identificador da agência de viagens. | Elemento Opcional |
| <i>route_id</i> | Identificador da rota. | Elemento Opcional |
| <i>route_type</i> | Tipo de Rota. (Autocarro, Metro, ...) | Elemento Opcional |
| <i>trip</i> | Detalhes da viagem do veículo. | Elemento Opcional |
| <i>stop_id</i> | Identificador da paragem. | Elemento Opcional |

Tabela D.16: *TranslatedString*

| Nome | Descrição | Tipo |
|--------------------|---|-------------------|
| <i>translation</i> | Tradução dos elementos do GTFS RT. <i>Text</i> - Texto a traduzir. <i>Language</i> - Código da língua a traduzir. | Elemento Opcional |

Apêndice E

Especificação GTFS - Mobilis

E.0.5 *agency.txt*

agency_id,agency_name,agency_url,agency_timezone,agency_phone
RDT,Rodotejo,http://www.rodotejo.pt/,Europe/Lisbon,244811507

E.0.6 *calendar.txt*

service_id,monday,tuesday,wednesday,thursday,friday,saturday,
sunday,start_date,end_date
FULLW,1,1,1,1,1,1,1,20140101,20151231
WE,0,0,0,0,0,1,0,20140101,20151231

E.0.7 *calendar_dates.txt*

service_id,date,exception_type
FULLW,20140101,2
FULLW,20140418,2
FULLW,20140420,2
FULLW,20140501,2
FULLW,20140610,2
FULLW,20140815,2
FULLW,20141208,2
FULLW,20141225,2

FULLW,20150101,2
FULLW,20150403,2
WE,20150425,2
FULLW,20150501,2
FULLW,20150610,2
WE,20150815,2
FULLW,20151208,2
FULLW,20151225,2

E.0.8 *fare_attributes.txt*

fare_id,price,currency_type,payment_method,transfers,transfer_duration
p,4.80,EUR,1,0,
u,1.25,EUR,0,0,
d,2.00,EUR,0,0,
s,7.00,EUR,1,0,
m,15.55,EUR,1,0,
r,11.65,EUR,1,0,

E.0.9 *fare_rules.txt*

fare_id,route_id,origin_id,destination_id,contains_id

E.0.10 *feed_info.txt*

feed_publisher_name,feed_publisher_url,feed_lang,feed_start_date,
feed_end_date,feed_version
Rodoviária do Tejo,<http://www.rodotejo.pt/>,pt,20140101,20151231,

E.0.11 *frequencies.txt*

```
trip_id,start_time,end_time,headway_secs
MOB1D,7:00:00,18:56:00,1020
MOB1N,19:30:00,22:30:00,3600
MOB1S,7:30:00,12:50:00,2400
MOB2D,7:09:00,19:05:00,1020
MOB2N,20:00:00,23:00:00,3600
MOB2S,7:50:00,13:10:00,2400
```

E.0.12 *frequencies.txt*

```
trip_id,start_time,end_time,headway_secs
MOB1D,7:00:00,18:56:00,1020
MOB1N,19:30:00,22:30:00,3600
MOB1S,7:30:00,12:50:00,2400
MOB2D,7:09:00,19:05:00,1020
MOB2N,20:00:00,23:00:00,3600
MOB2S,7:50:00,13:10:00,2400
```

E.0.13 *trips.txt*

```
route_id,service_id,trip_id,trip_headsign,direction_id,block_id,shape_id
MOB1,FULLW,MOB1D,,
MOB1,FULLW,MOB1N,,
MOB1,FULLW,MOB1S,,
MOB2,FULLW,MOB2D,,
MOB2,FULLW,MOB2N,,
MOB2,FULLW,MOB2S,,
```

E.0.14 routes.txt

route_id,agency_id,route_short_name,route_long_name,route_desc,
route_type,route_url,route_color,route_text_color
MOB1,RDT,Mobilis 1,Mobilis 1 - Linha Verde,,3,,
MOB2,RDT,Mobilis 2,Mobilis 2 - Linha Vermelha,,3,,

E.0.15 shapes.txt

shape_id,shape_pt_lat,shape_pt_lon,shape_pt_sequence,shape_dist_traveled

E.0.16 stop_times.txt

trip_id,arrival_time,departure_time,stop_id,stop_sequence,stop_headsign,
pickup_type,drop_off_time,shape_dist_traveled
MOB1D,7:00:00,7:00:00,1_LJL,1,,
MOB1D,7:01:00,7:01:00,2_AHA,2,,
MOB1D,7:03:00,7:03:00,3_MM_1,3,,
MOB1D,7:05:00,7:05:00,4_NL_1,4,,
MOB1D,7:07:00,7:07:00,5_AG_1,5,,
MOB1D,7:11:00,7:11:00,6_BA,6,,
MOB1D,7:14:00,7:14:00,7_EMP_1,7,, MOB1D,7:15:00,7:15:00,8_PIS_1,8,,
MOB1D,7:16:00,7:16:00,9_A25A_1,9,,
MOB1D,7:19:00,7:19:00,10_ESDS_1,10,,
MOB1D,7:22:00,7:22:00,11_BC_1,11,,
MOB1D,7:23:00,7:23:00,12_CML_1,12,,
MOB1D,7:24:00,7:24:00,13_PM_1,13,,
MOB1D,7:26:00,7:26:00,14_ESTG,14,,
MOB1D,7:27:00,7:27:00,15_RESTG,15,,
MOB1D,7:31:00,7:31:00,16_PE,16,,
MOB1D,7:33:00,7:33:00,17_SAP_1,17,,
MOB1D,7:34:00,7:34:00,18_BS_1,18,,
MOB1D,7:36:00,7:36:00,19_AMP_1,19,,
MOB1D,7:37:00,7:37:00,20_ASE_1,20,,

MOB1D,7:38:00,7:38:00,21_ANSF_2,21,,,,
MOB1D,7:40:00,7:40:00,22_GUI_1,22,,,,
MOB1D,7:42:00,7:42:00,23_EPL,23,,,,
MOB1D,7:44:00,7:44:00,24_SR1,24,,,,
MOB1D,7:45:00,7:45:00,25_SR2_RF,25,,,,
MOB1D,7:46:00,7:46:00,26_SR3_CETL,26,,,,
MOB1D,7:48:00,7:48:00,27_HPVI,27,,,,
MOB1D,7:49:00,7:49:00,28_HPCE,28,,,,
MOB1D,7:50:00,7:50:00,29_ECM1,29,,,,
MOB1D,7:52:00,7:52:00,30_BM1,30,,,,
MOB1D,7:53:00,7:53:00,31_PHR1,31,,,,
MOB1D,7:55:00,7:55:00,32_JLC1,32,,,,
MOB1N,19:30:00,19:30:00,1_LJL,1,,,,
MOB1N,19:31:00,19:31:00,2_AHA,2,,,,
MOB1N,19:33:00,19:33:00,3_MM_1,3,,,,
MOB1N,19:35:00,19:35:00,4_NL_1,4,,,,
MOB1N,19:37:00,19:37:00,5_AG_1,5,,,,
MOB1N,19:41:00,19:41:00,6_BA,6,,,,
MOB1N,19:44:00,19:44:00,7_EMP_1,7,,,,
MOB1N,19:45:00,19:45:00,8_PIS_1,8,,,,
MOB1N,19:46:00,19:46:00,9_A25A_1,9,,,,
MOB1N,19:49:00,19:49:00,10_ESDS_1,10,,,,
MOB1N,19:52:00,19:52:00,11_BC_1,11,,,,
MOB1N,19:53:00,19:53:00,12_CML_1,12,,,,
MOB1N,19:54:00,19:54:00,13_PM_1,13,,,,
MOB1N,19:56:00,19:56:00,14_ESTG,14,,,,
MOB1N,19:57:00,19:57:00,15_RESTG,15,,,,
MOB1N,20:01:00,20:01:00,16_PE,16,,,,
MOB1N,20:03:00,20:03:00,17_SAP_1,17,,,,
MOB1N,20:04:00,20:04:00,18_BS_1,18,,,,
MOB1N,20:06:00,20:06:00,19_AMP_1,19,,,,
MOB1N,20:07:00,20:07:00,20_ASE_1,20,,,,
MOB1N,20:08:00,20:08:00,21_ANSF_2,21,,,,
MOB1N,20:10:00,20:10:00,22_GUI_1,22,,,,
MOB1N,20:12:00,20:12:00,23_EPL,23,,,,

MOB1N,20:14:00,20:14:00,24_SR1,24,,,,
MOB1N,20:15:00,20:15:00,25_SR2_RF,25,,,,
MOB1N,20:16:00,20:16:00,26_SR3_CETL,26,,,,
MOB1N,20:18:00,20:18:00,27_HPVI,27,,,,
MOB1N,20:19:00,20:19:00,28_HPCE,28,,,,
MOB1N,20:20:00,20:20:00,29_ECM1,29,,,,
MOB1N,20:22:00,20:22:00,30_BM1,30,,,,
MOB1N,20:23:00,20:23:00,31_PHR1,31,,,,
MOB1N,20:25:00,20:25:00,32_JLC1,32,,,,
MOB1S,7:30:00,7:30:00,1_LJL,1,,,,
MOB1S,7:31:00,7:31:00,2_AHA,2,,,,
MOB1S,7:33:00,7:33:00,3_MM_1,3,,,,
MOB1S,7:35:00,7:35:00,4_NL_1,4,,,,
MOB1S,7:37:00,7:37:00,5_AG_1,5,,,,
MOB1S,7:41:00,7:41:00,6_BA,6,,,,
MOB1S,7:44:00,7:44:00,7_EMP_1,7,,,,
MOB1S,7:45:00,7:45:00,8_PIS_1,8,,,,
MOB1S,7:46:00,7:46:00,9_A25A_1,9,,,,
MOB1S,7:49:00,7:49:00,10_ESDS_1,10,,,,
MOB1S,7:52:00,7:52:00,11_BC_1,11,,,,
MOB1S,7:53:00,7:53:00,12_CML_1,12,,,,
MOB1S,7:54:00,7:54:00,13_PM_1,13,,,,
MOB1S,7:56:00,7:56:00,14_ESTG,14,,,,
MOB1S,7:57:00,7:57:00,15_RESTG,15,,,,
MOB1S,8:01:00,8:01:00,16_PE,16,,,,
MOB1S,8:03:00,8:03:00,17_SAP_1,17,,,,
MOB1S,8:04:00,8:04:00,18_BS_1,18,,,,
MOB1S,8:06:00,8:06:00,19_AMP_1,19,,,,
MOB1S,8:07:00,8:07:00,20_ASE_1,20,,,,
MOB1S,8:08:00,8:08:00,21_ANSF_2,21,,,,
MOB1S,8:10:00,8:10:00,22_GUI_1,22,,,,
MOB1S,8:12:00,8:12:00,23_EPL,23,,,,
MOB1S,8:14:00,8:14:00,24_SR1,24,,,,
MOB1S,8:15:00,8:15:00,25_SR2_RF,25,,,,
MOB1S,8:16:00,8:16:00,26_SR3_CETL,26,,,,

MOB1S,8:18:00,8:18:00,27_HPVI,27,,,,
MOB1S,8:19:00,8:19:00,28_HPCE,28,,,,
MOB1S,8:20:00,8:20:00,29_ECM1,29,,,,
MOB1S,8:22:00,8:22:00,30_BM1,30,,,,
MOB1S,8:23:00,8:23:00,31_PHR1,31,,,,
MOB1S,8:25:00,8:25:00,32_JLC1,32,,,,
MOB2D,7:09:00,7:09:00,1_LJL,1,,,,
MOB2D,7:12:00,7:12:00,32_JLC1,2,,,,
MOB2D,7:13:00,7:13:00,34_F3B,3,,,,
MOB2D,7:14:00,7:14:00,35_PHR2,4,,,,
MOB2D,7:15:00,7:15:00,36_BM2,5,,,,
MOB2D,7:17:00,7:17:00,27_HPVI,6,,,,
MOB2D,7:18:00,7:18:00,28_HPCE,7,,,,
MOB2D,7:19:00,7:19:00,39_ECM2,8,,,,
MOB2D,7:21:00,7:21:00,40_SR1_CETL,9,,,,
MOB2D,7:23:00,7:23:00,41_SR2_RF,10,,,,
MOB2D,7:24:00,7:24:00,42_SR3_RPP,11,,,,
MOB2D,7:26:00,7:26:00,43_GUI_2,12,,,,
MOB2D,7:28:00,7:28:00,44_ANSF_2,13,,,,
MOB2D,7:29:00,7:29:00,45_ASE_1,14,,,,
MOB2D,7:30:00,7:30:00,46_AMP_2,15,,,,
MOB2D,7:31:00,7:31:00,47_BS_2,16,,,,
MOB2D,7:32:00,7:32:00,48_SAP_2,17,,,,
MOB2D,7:35:00,7:35:00,63_RESTG_2,18,,,,
MOB2D,7:36:00,7:36:00,14_ESTG,19,,,,
MOB2D,7:38:00,7:38:00,51_PM_2,20,,,,
MOB2D,7:39:00,7:39:00,52_CML_2,21,,,,
MOB2D,7:40:00,7:40:00,53_BC_2,22,,,,
MOB2D,7:43:00,7:43:00,54_ESDS_2,23,,,,
MOB2D,7:44:00,7:44:00,55_A25A2,24,,,,
MOB2D,7:46:00,7:46:00,56_PIS_2,25,,,,
MOB2D,7:47:00,7:47:00,57_EMP_2,26,,,,
MOB2D,7:50:00,7:50:00,6_BA,27,,,,
MOB2D,7:53:00,7:53:00,59_AG_2,28,,,,
MOB2D,7:55:00,7:55:00,60_NL_2,29,,,,

MOB2D,7:57:00,7:57:00,61_MM_2,30,,,,
MOB2D,8:00:00,8:00:00,62_LCM,31,,,,
MOB2N,20:00:00,20:00:00,1_LJL,1,,,,
MOB2N,20:03:00,20:03:00,32_JLC1,2,,,,
MOB2N,20:04:00,20:04:00,34_F3B,3,,,,
MOB2N,20:05:00,20:05:00,35_PHR2,4,,,,
MOB2N,20:06:00,20:06:00,36_BM2,5,,,,
MOB2N,20:08:00,20:08:00,27_HPVI,6,,,,
MOB2N,20:09:00,20:09:00,28_HPCE,7,,,,
MOB2N,20:10:00,20:10:00,39_ECM2,8,,,,
MOB2N,20:12:00,20:12:00,40_SR1_CETL,9,,,,
MOB2N,20:13:00,20:13:00,41_SR2_RF,10,,,,
MOB2N,20:14:00,20:14:00,42_SR3_RPP,11,,,,
MOB2N,20:16:00,20:16:00,43_GUI_2,12,,,,
MOB2N,20:18:00,20:18:00,44_ANSF_2,13,,,,
MOB2N,20:19:00,20:19:00,45_ASE_1,14,,,,
MOB2N,20:20:00,20:20:00,46_AMP_2,15,,,,
MOB2N,20:21:00,20:21:00,47_BS_2,16,,,,
MOB2N,20:22:00,20:22:00,48_SAP_2,17,,,,
MOB2N,20:25:00,20:25:00,63_RESTG_2,18,,,,
MOB2N,20:26:00,20:26:00,14_ESTG,19,,,,
MOB2N,20:28:00,20:28:00,51_PM_2,20,,,,
MOB2N,20:29:00,20:29:00,52_CML_2,21,,,,
MOB2N,20:30:00,20:30:00,53_BC_2,22,,,,
MOB2N,20:33:00,20:33:00,54_ESDS_2,23,,,,
MOB2N,20:34:00,20:34:00,55_A25A2,24,,,,
MOB2N,20:36:00,20:36:00,56_PIS_2,25,,,,
MOB2N,20:37:00,20:37:00,57_EMP_2,26,,,,
MOB2N,20:40:00,20:40:00,6_BA,27,,,,
MOB2N,20:43:00,20:43:00,59_AG_2,28,,,,
MOB2N,20:45:00,20:45:00,60_NL_2,29,,,,
MOB2N,20:47:00,20:47:00,61_MM_2,30,,,,
MOB2N,20:50:00,20:50:00,62_LCM,31,,,,
MOB2S,7:50:00,7:50:00,1_LJL,1,,,,
MOB2S,7:53:00,7:53:00,32_JLC1,2,,,,

MOB2S,7:54:00,7:54:00,34_F3B,3,,,,
 MOB2S,7:55:00,7:55:00,35_PHR2,4,,,,
 MOB2S,7:56:00,7:56:00,36_BM2,5,,,,
 MOB2S,7:58:00,7:58:00,27_HPVI,6,,,,
 MOB2S,7:59:00,7:59:00,28_HPCE,7,,,,
 MOB2S,8:00:00,8:00:00,39_ECM2,8,,,,
 MOB2S,8:02:00,8:02:00,40_SR1_CETL,9,,,,
 MOB2S,8:03:00,8:03:00,41_SR2_RF,10,,,,
 MOB2S,8:04:00,8:04:00,42_SR3_RPP,11,,,,
 MOB2S,8:06:00,8:06:00,43_GUI_2,12,,,,
 MOB2S,8:08:00,8:08:00,44_ANSF_2,13,,,,
 MOB2S,8:09:00,8:09:00,45_ASE_1,14,,,,
 MOB2S,8:10:00,8:10:00,46_AMP_2,15,,,,
 MOB2S,8:11:00,8:11:00,47_BS_2,16,,,,
 MOB2S,8:12:00,8:12:00,48_SAP_2,17,,,,
 MOB2S,8:15:00,8:15:00,63_RESTG_2,18,,,,
 MOB2S,8:16:00,8:16:00,14_ESTG,19,,,,
 MOB2S,8:18:00,8:18:00,51_PM_2,20,,,,
 MOB2S,8:19:00,8:19:00,52_CML_2,21,,,,
 MOB2S,8:20:00,8:20:00,53_BC_2,22,,,,
 MOB2S,8:23:00,8:23:00,54_ESDS_2,23,,,,
 MOB2S,8:24:00,8:24:00,55_A25A2,24,,,,
 MOB2S,8:26:00,8:26:00,56_PIS_2,25,,,,
 MOB2S,8:27:00,8:27:00,57_EMP_2,26,,,,
 MOB2S,8:30:00,8:30:00,6_BA,27,,,,
 MOB2S,8:33:00,8:33:00,59_AG_2,28,,,,
 MOB2S,8:35:00,8:35:00,60_NL_2,29,,,,
 MOB2S,8:37:00,8:37:00,61_MM_2,30,,,,
 MOB2S,8:40:00,8:40:00,62_LCM,31,,,,

E.0.17 stops.txt

stop_id,stop_name,stop_desc,stop_lat,stop_lon,zone_id,stop_url
 1_LJL,Largo José Lúcio da Silva,,39.746458,-8.804027,,

2_AHA,Avenida Herois de Angola,,39.747928,-8.804115,,
3_MM_1,Mercado Municipal/Av.Cidade Maringá,,39.749772,-8.806574,,
4_NL_1,Nova Leiria,,39.752397,-8.808659,,
5_AG_1,Almoinha Grande,,39.754286,-8.811902,,
6_BA,Bairro das Almoínhas,,39.754148,-8.819742,,
7_EMP_1,Estádio Magalhães Pessoa,,39.750760,-8.812511,,
8_PIS_1,Piscinas,,39.749694,-8.809147,,
9_A25A_1,Avenida 25 de Abril,,39.746837,-8.815306,,
10_ESDS_1,Escola Secundária Domingos Sequeira,,39.745674,-8.813779,,
11_BC_1,Bairro dos Capuchos,,39.74256,-8.81132,,
12_CML_1,Câmara Municipal/ESECS,,39.741083,-8.811194,,
13_PM_1,Porto Moniz,,39.739442,-8.815778,,
14_ESTG,ESTG - (Campus 2 IPL),,39.735103,-8.823308,,
15_RESTG,Rotunda ESTG,,39.735139,-8.823366,,
16_PE,Prisão Escola,,39.733957,-8.808903,,
17_SAP_1,Centro de Saúde Gorjão Henriques,,39.736867,-8.810832,,
18_BS_1,Bairro dos Sargentos,,39.739103,-8.808943,,
19_AMP_1,Avenida Marquês de Pombal,,39.739662,-8.806069,,
20_ASE_1,Avenida Nossa Senhora de Fátima (Senhora da Encarnação),,
39.739419,-8.802866,,
21_ANSF_2,Avenida Nossa Senhora de Fátima,,39.73727,-8.800124,,
22_GUI_1,Guimarota,,39.734118,-8.799379,,
23_EPL,Escola Profissional/ISLA,,39.732053,-8.795058,,
24_SR1,São Romão - Rua Flor. Pedro Lopes,,39.732142,-8.791447,,
25_SR2_RF,São Romão - Rua da Fonte,,39.735125,-8.792585,,
26_SR3_CETL,São Romão - R. Esperança CETL,,39.736079,-8.793785,,
27_HPVI,Hospital - Visitas,,39.743936,-8.793741,,
28_HPCE,Hospital - Consultas Externas,,39.742486,-8.794478,,
29_ECM1,Escola Correia Mateus,,39.745023,-8.792563,,
30_BM1,Bombeiros Municipais,,39.741486,-8.799895,,
31_PHR1,Ponte Hintze Ribeiro,,39.74263,-8.801994,,
32_JLC1,Jardim Luís de Camões,,39.745196,-8.806288,,
34_F3B,Rua Tenente Valadim/Fonte das 3 Bicas,,39.743158,-8.80624,,
35_PHR2,Ponte Hintze Ribeiro,,39.742655,-8.80208,,
36_BM2,Bombeiros Municipais,,39.741381,-8.799744,,

39_ECM2,Escola Correia Mateus,,39.744492, -8.792245,,
40_SR1_CETL,São Romão - R. Esperança CETL,,39.735991,-8.793875,,
41_SR2_RF,São Romão - Rua da Fonte,,39.735133,-8.793164,,
42_SR3_RPP,Rua Coronel Pereira Pascoal,,39.733633,-8.796187,,
43_GUI_2,Guimarota,,39.73405,-8.799682,,
44_ANSF_2,Avenida Nossa Senhora de Fátima,,39.737243,-8.800403,,
45_ASE_1,Avenida Nossa Senhora de Fátima (Srª da Encarnação),,
39.739528,-8.803303,,
46_AMP_2,Avenida Marquês de Pombal,,39.739831,-8.806704,,
47_BS_2,Bairro dos Sargentos,,39.738909,-8.808997,,
48_SAP_2,Centro de Saúde Gorjão Henriques,,39.737051,-8.810797,,
51_PM_2,Porto Moniz,,39.739312, -8.815671,,
52_CML_2,Câmara Municipal/ESECS,,39.740242, -8.811885,,
53_BC_2,Bairro dos Capuchos,,39.742548,-8.811191,,
54_ESDS_2,Escola Secundária Domingos Sequeira,,39.745806,-8.813911,,
55_A25A2,Avenida 25 de Abril,,39.746736,-8.815214,,
56_PIS_2,Piscinas,,39.749801,-8.809008,,
57_EMP_2,Estádio Magalhães Pessoa,,39.750916,-8.812524,,
59_AG_2,Almoíña Grande,,39.754175,-8.81188,,
60_NL_2,Nova Leiria,,39.752348,-8.808525,,
61_MM_2,Mercado Municipal/Rua Capitão Mouzinho de Albuquerque,,
39.748497,-8.80587,,
62_LCM,Largo Cónego Maia/Sé,,39.745717,-8.806009,,
63_RESTG_2,Rotunda ESTG,,39.735132, -8.823689,,

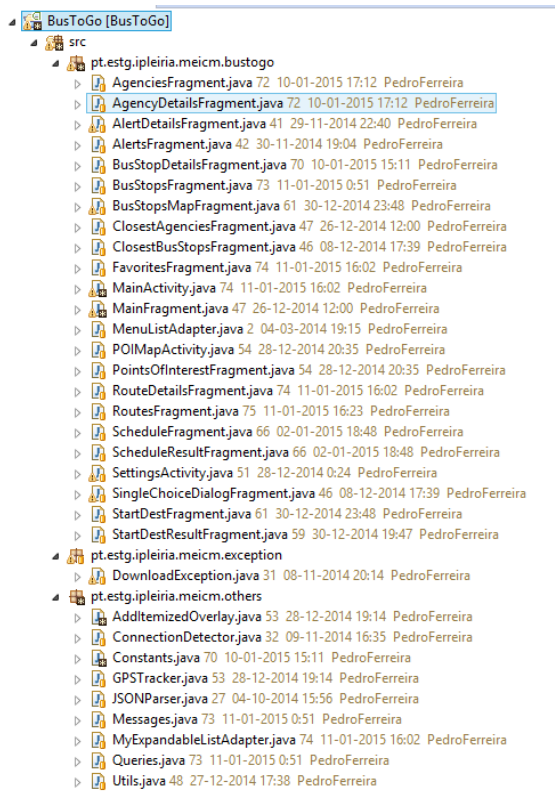
Apêndice F

Script de Alteração à Referência GTFS Online

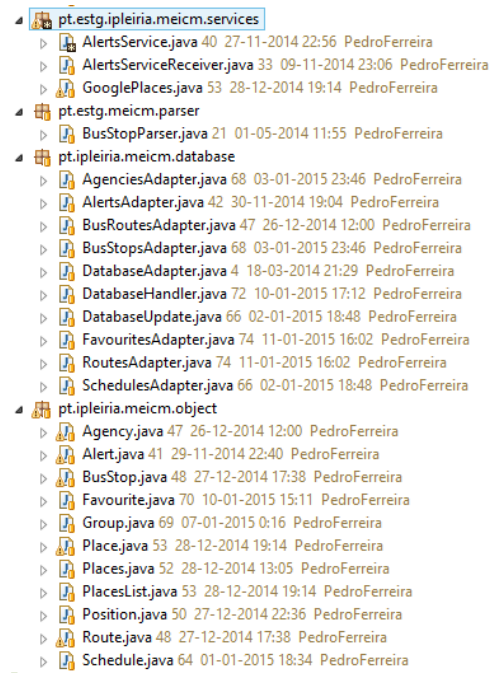
```
mongo kahana.mongohq.com:10082/bustogoDB -u <username> -p <password>
db.agencies.remove()
db.calendardates.remove()
db.calendars.remove()
db.fareattributes.remove()
db.feedinfos.remove()
db.frequencies.remove()
db.routes.remove()
db.stops.remove()
db.stoptimes.remove()
db.trips.remove()
heroku login (<user> - <password>)
heroku run node ./scripts/download --app bustogo-data
```


Apêndice G

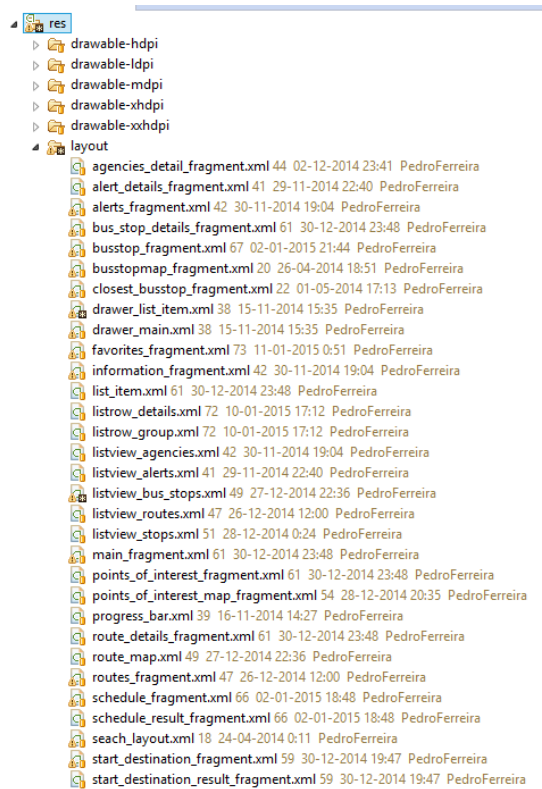
Estrutura Aplicação Cliente



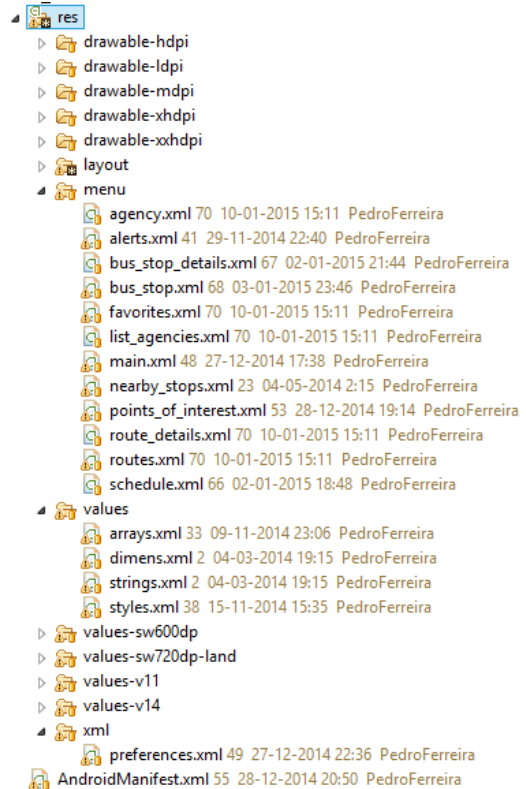
(i) Estrutura I



(j) Estrutura II



(k) Estrutura III

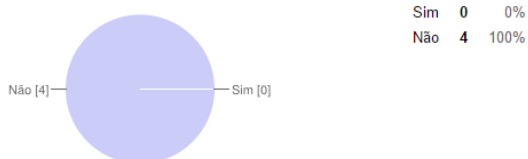


(l) Estrutura IV

Apêndice H

Inquérito Realizado às Empresas de Transporte

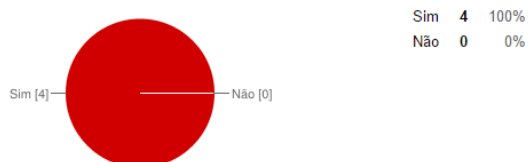
1 - De uma forma geral, considera que tem existido um aumento na utilização dos transportes públicos nos últimos anos?



2 - De que forma é disponibilizada, pela sua empresa, informação sobre horários, paragens, entre outros, aos passageiros?



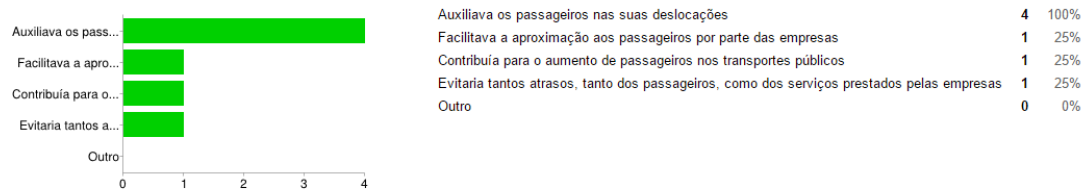
3 - Acha que seria uma mais valia a existência de uma aplicação móvel para devolver informação aos passageiros?



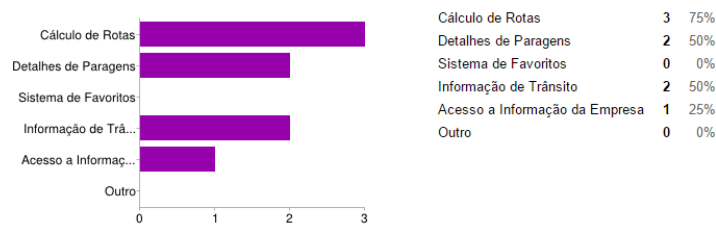
3a - Tendo em conta a resposta anterior como classifica a existência de uma aplicação deste género?



3b - Tendo em consideração a resposta à pergunta 3, considera que esta existência era uma mais valia porque ...



3c - Do ponto de vista de empresa, quais as funcionalidades que considera importantes num aplicação deste género?



4 - Se não tem aplicação móvel, avalie, em questão de necessidade, a existência de uma para a sua empresa.



(n) Resultados Inquérito II