

Mestrado em Engenharia Informática
Estágio
Relatório Final

Projecto EcoMobile: desenvolvimento de uma infraestrutura para processamento de dados em larga escala em ambientes *"cloud"*

David Baptista Miranda
davidmir@student.dei.uc.pt

Orientador:
Prof. Bruno Miguel Brás Cabral

Co-orientador:
Prof. António Jorge da Costa Granjal

01 de Julho de 2014



FCTUC DEPARTAMENTO
DE ENGENHARIA INFORMÁTICA
FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

Agradecimentos

Gostaria de agradecer aos meus orientadores, Professor Bruno Cabral e Professor Jorge Granjal, pela oportunidade e confiança depositada em mim para integrar este projecto. Agradeço ainda à minha equipa de trabalho, Pedro Costa e Rui Chicória, pelo bom ambiente de trabalho proporcionado, companheirismo e apoio oferecido. Por fim, mas nunca menos importante, agradeço a toda a minha família, namorada e aos meus amigos de sempre, por todos os valores transmitidos, pela compreensão e apoio nos bons e maus momentos.

Resumo

Devido à actual conjuntura económica e ao aumento do preço dos combustíveis, da sinistralidade rodoviária e dos custos inerentes à manutenção dos veículos, os condutores e organizações têm sentido cada vez mais a necessidade de verem reduzidos os custos com os seus veículos.

Assim, com o objectivo de ajudar os condutores a tornarem a sua condução mais económica e segura, surgiu o projecto EcoMobile que pretende, através da análise de informação sensorial digital proveniente dos *smartphones*, desenvolver soluções que avaliem a economia e segurança dos condutores e os auxiliem na tomada de decisão.

Pretende-se no âmbito deste estágio, desenvolver uma arquitectura de software altamente escalável, que possua a capacidade de processar dados sensoriais provenientes dos *smartphones*, com a finalidade de criar soluções de *business intelligence* para suporte à decisão dos condutores e organizações. Estas soluções permitirão aos condutores compararem a evolução da sua condução ao longo dos seus trajectos mais frequentes, obter a recomendação dos melhores trajectos para uma determinada rota e realizar a comparação de trajectos e diversos indicadores-chave que medem a economia e segurança da sua condução. Permitirá também aos gestores de frotas criar um conjunto de análises, *dashboards* e relatórios personalizados, que espelham o estado actual da organização e a evolução da condução dos seus condutores.

No final, as soluções desenvolvidas serão disponibilizadas como um serviço na *cloud* que, através de uma aplicação web, permitirá visualizar e manipular todo um conjunto de informação, que ajudará a desenvolver um estilo de condução mais económico e seguro e a tomar as melhores decisões operacionais e estratégicas.

Palavras chave: *business intelligence*, *cloud computing*, aplicações web, engenharia de software

Conteúdo

1	Introdução	1
1.1	Enquadramento	1
1.2	Objectivos	3
1.3	Estrutura do documento	4
2	Metodologia	5
2.1	Metodologia de desenvolvimento	5
2.1.1	Desenvolvimento ágil de software	5
2.2	Planeamento	7
2.3	Processos de engenharia de software	13
2.3.1	Controlo de versões de software	13
2.3.2	Ambientes	14
3	Estado da arte	17
3.1	Metodologias e tecnologias de referência	17
3.1.1	Sistemas de suporte à decisão	17
3.1.2	<i>Business intelligence</i>	19
3.1.3	<i>Data mining</i>	20
3.1.4	NoSQL	24
3.1.5	<i>Cloud computing</i>	28
3.2	Análise de soluções comerciais	32
4	Análise de requisitos	39
4.1	Actores do sistema	40
4.2	Prototipagem	41
4.2.1	Prototipagem de baixa fidelidade	41
4.2.2	Prototipagem de alta fidelidade	43
4.3	Requisitos funcionais	46
4.3.1	<i>Business-to-consumer</i>	47
4.3.2	<i>Business-to-business</i>	48
4.4	Requisitos não funcionais	50

5	Análise e desenho da arquitectura	53
5.1	Visão global da arquitectura	54
5.2	Decisões arquitecturais	56
5.3	Arquitectura desenvolvida	58
5.3.1	<i>Business-to-consumer</i>	58
5.3.2	<i>Business-to-business</i>	63
5.3.3	Perspectiva física e tecnológica	67
5.3.4	Resumo do desenho arquitectural	70
6	Trabalho desenvolvido	73
6.1	Classificação de trajectos	73
6.1.1	Definição do problema	73
6.1.2	Desenho da solução	76
6.1.3	Validação	80
6.2	Comparação de trajectos	82
6.2.1	Desenho da solução	82
6.2.2	Interface e funcionalidades	85
6.3	Recomendação de trajectos	87
6.3.1	Desenho da solução	87
6.3.2	Interface e funcionalidades	88
6.4	<i>Reporting</i>	89
6.4.1	Modelo de dados	89
6.4.2	Interface e funcionalidades	93
7	Testes	95
7.1	Testes de aceitação	95
7.2	Testes de carga	95
7.2.1	Cenário	96
7.2.2	Resultados	96
8	Conclusão	97
A	Artefactos da metodologia ágil	99
A.1	<i>Business-to-consumer</i>	99
A.2	<i>Business-to-business</i>	103
B	Análise de riscos	105
C	Análise de funcionalidades e custos	109
D	Análise das necessidades de negócio	111

E	Análise de requisitos	113
E.1	Requisitos funcionais	113
E.1.1	<i>Business-to-consumer</i>	113
E.1.2	<i>Business-to-business</i>	117
F	Modelo de dados	123
F.1	Classificação e recomendação	123
F.1.1	Colecção <i>trips_geospatial</i>	123
F.1.2	Colecção <i>trips_clusters</i>	124
F.2	<i>Reporting</i>	125
F.2.1	Colecção <i>user_daily_metrics</i>	125
F.2.2	Modelo <i>reporting</i> B2B	128
G	Trabalho desenvolvido	131
G.1	Algoritmo de recomendação	132
H	Testes e validação	133
H.1	Validação <i>clusters</i>	133
H.2	Testes de aceitação das funcionalidades <i>business-to-consumer</i> .	136
H.3	Testes de aceitação das funcionalidades <i>business-to-business</i> .	137

Lista de Acrónimos

API	<i>Application Programming Interface</i>
B2B	<i>Business-to-business</i>
B2C	<i>Business-to-consumer</i>
BI	<i>Business Intelligence</i>
CAP	<i>Consistency-Availability-Partition tolerance</i>
CSS	<i>Cascading Style Sheets</i>
DBMS	<i>Database Management Systems</i>
DSS	<i>Decision Support Systems</i>
EIS	<i>Enterprise Information Systems</i>
GPRS	<i>General Packet Radio Service</i>
GPS	<i>Global Positioning System</i>
HTML	<i>HyperText Markup Language</i>
HTTP	<i>HyperText Transfer Protocol</i>
IaaS	<i>Infrastructure as a Service</i>
JSON	<i>JavaScript Object Notation</i>
KDD	<i>Knowledge Discovery in Databases</i>
KPI	<i>Key Performance Indicators</i>
LCS	<i>Longest Common Subsequence</i>
MVC	<i>Model-View-Controller</i>
MVT	<i>Model-View-Template</i>
MVW	<i>Model View Whatever</i>
OLAP	<i>On-line Analytical Processing</i>
PaaS	<i>Platform as a Service</i>
REST	<i>Representational State Transfer</i>
SaaS	<i>Software as a Service</i>
SOAP	<i>Simple Object Access Protocol</i>
XML	<i>eXtensible Markup Language</i>

Lista de Tabelas

3.1	Características de dois géneros de BI	20
3.2	Propriedades garantidas pelas bases de dados (CAP)	27
4.1	Actores presentes na solução B2C	40
4.2	Actores presentes na solução B2B	40
4.3	Requisitos funcionais da vertente B2C	47
4.4	Requisitos funcionais da vertente B2B	48
6.1	Resumo da avaliação da coesão dos <i>clusters</i>	81
6.2	Resumo da avaliação da separação dos <i>clusters</i>	81
6.3	<i>Performance</i> do algoritmo de redução da dimensionalidade e cálculo de agregados	82
6.4	<i>Performance</i> do algoritmo de classificação	82
6.5	<i>Performance</i> do algoritmo de alinhamento local	84
6.6	<i>Performance</i> do algoritmo de alinhamento global	84
6.7	<i>Performance</i> do algoritmo de recomendação	88
6.8	Descrição das entidades do modelo da componente de <i>reporting</i>	92
7.1	Condições do teste de visualizar e analisar viagens	96
F.1	Modelo de dados <i>trips_geospatial</i>	123
F.2	Modelo de dados <i>trips_clusters</i>	124
F.3	Dimensões do modelo de dados <i>user_daily_metrics</i>	126
F.4	Descrição dos factos pré-calculados respeitantes às viagens de um condutor	126
F.5	Descrição das entidades do modelo da componente de <i>reporting</i>	128
H.1	Avaliação da coesão e separação dos <i>clusters</i>	133
H.2	Conjunto de testes de aceitação da vertente B2C	136
H.3	Conjunto de testes de aceitação da vertente B2B	137

Lista de Figuras

2.1	Representação da metodologia <i>Scrum</i> . Imagem adaptada [4]	6
2.2	Estrutura do repositório de código e controlo de versões	13
3.1	Arquitectura de um sistema de suporte à decisão utilizando uma <i>data warehouse</i> [12]	19
3.2	Fases do processo de <i>data mining</i> [20]	21
3.3	Procedimento para elaboração de <i>clusters</i> [24]	23
3.4	Representação visual do teorema de CAP	26
3.5	Exemplo de uma arquitectura para ” <i>Big Data Analytics</i> ” [32]	28
3.6	<i>Cloud computing</i> [37]	31
3.7	<i>Dashboard</i> da solução TomTom WEBFLEET	33
3.8	<i>Dashboard</i> da solução Fleetmatics	34
3.9	Solução Frotcom	35
3.10	Solução <i>mobile</i> Automatic	36
3.11	Solução MyCarTracks	36
4.1	Elaboração dos protótipos de baixa fidelidade do módulo de comparação de rotas	42
4.2	Elaboração dos protótipos de baixa fidelidade do módulo de <i>reporting</i>	43
4.3	Protótipo de alta fidelidade do módulo de comparação de trajectos	44
4.4	Protótipo de alta fidelidade do módulo de recomendação de trajectos	45
4.5	Protótipo de alta fidelidade do módulo de <i>reporting</i>	45
4.6	Protótipo de alta fidelidade do módulo de <i>reporting</i> no perfil do condutor	46
5.1	Visão global da arquitectura da solução B2C	54
5.2	Visão global da arquitectura da solução B2B	55
5.3	Perspectiva estática da arquitectura B2C	58

5.4	Perspectiva dinâmica da arquitectura - Classificação	61
5.5	Perspectiva dinâmica da arquitectura - Comparação	62
5.6	Perspectiva dinâmica da arquitectura - Recomendação	63
5.7	Perspectiva estática da arquitectura B2B	64
5.8	Perspectiva dinâmica da arquitectura - Agregados	66
5.9	Perspectiva dinâmica da arquitectura - Disponibilização de agregados	67
5.10	Perspectiva física e tecnológica	68
6.1	Representação de trajectos semelhantes e não semelhantes . . .	74
6.2	Redução da granularidade das coordenadas GPS	75
6.3	Modelo de dados da colecção <i>trips_geospatial</i>	79
6.4	Modelo de dados da colecção <i>trips_clusters</i>	80
6.5	Exemplo do alinhamento local entre trajectos do mesmo <i>cluster</i>	83
6.6	Exemplo do alinhamento global entre trajectos que possuem zonas iguais	83
6.7	Interface desenvolvido para visualizar a evolução e comparação de trajectos	85
6.8	Seleção dos trajectos para comparação	86
6.9	Interface desenvolvido para visualizar a recomendação de trajectos	88
6.10	Modelo de dados para agregação de métricas de condução . . .	90
6.11	Exemplo de um <i>pipeline</i> de agregação	91
6.12	Modelo de dados da componente de <i>reporting</i> da versão B2B .	92
6.13	Interface que permite a criação de análises, relatórios e <i>dashboards</i> personalizados	93
6.14	Funcionalidades de <i>reporting</i> incorporadas no perfil de cada condutor	93
F.1	Modelo de dados da colecção <i>trips_geospatial</i>	123
F.2	Modelo de dados da colecção <i>trips_clusters</i>	124
F.3	Modelo de dados da componente de <i>reporting</i> da versão B2B .	128

Capítulo 1

Introdução

Este documento tem como objectivo apresentar o trabalho desenvolvido na disciplina "Dissertação/Estágio" do Mestrado em Engenharia Informática da Universidade de Coimbra, no ano lectivo de 2013/2014, pelo aluno David Baptista Miranda, integrado no projecto EcoMobile.

Neste capítulo será realizada uma contextualização do projecto EcoMobile e do presente estágio, os objectivos a cumprir e a estrutura do documento.

1.1 Enquadramento

Contexto

Com a crise da energia no ano 2000 [1] em que o preço do barril disparou drasticamente, afectando principalmente os combustíveis líquidos, o gasto energético dos veículos passou a ser uma das maiores preocupações dos consumidores. Começou assim a sentir-se a necessidade de tornar mais eficiente a utilização dos veículos, de forma a garantir uma diminuição do consumo de combustível e dos custos de manutenção.

Este é um problema que afecta todos os condutores no seu quotidiano e as organizações que possuem uma frota de veículos, o que produz um grande impacto nos orçamentos familiares e custos operacionais e estratégicos das organizações.

Projecto EcoMobile

É neste sentido que surge o projecto EcoMobile. Um projecto com uma forte componente de investigação nas áreas de *business intelligence* (BI), redes e computação móvel e que pretende ajudar os condutores a adoptarem

comportamentos de eco-condução e de segurança rodoviária através da monitorização da condução dos veículos, com base no tratamento inteligente de informação sensorial digital, e da criação de soluções de suporte à decisão que ajudem os condutores a tornarem a sua condução mais eficiente.

Para a monitorização da condução e comunicação da informação é utilizado um *smartphone*. Com as capacidades dos *smartphones*, nos dias de hoje, é possível recolher informação de diversos sensores (e.g., GPS, acelerómetro, giroscópio) e criar indicadores, através de algoritmos de tratamento de dados, que avaliem a segurança e economia da condução dos condutores. Toda a informação monitorizada durante a condução permite, após a sua comunicação para o servidor, disponibilizar serviços de suporte à decisão que auxiliem os condutores no seu quotidiano.

Numa primeira instância do projecto este serviu como prova de conceito para diversos concursos, tendo vencido dois prémios de inovação tecnológica, em 2011 no concurso PT Galp Innovation Challenge e em 2012 no concurso Arrisca C, dando assim origem à empresa Sentilant¹. A empresa é actualmente constituída por uma equipa de gestão de projecto e uma equipa de desenvolvimento constituída por três elementos divididos por três áreas: desenvolvimento móvel, desenvolvimento da infra-estrutura web (*backend/frontend*) e desenvolvimento de soluções de BI.

Actualmente a Sentilant pretende expandir o projecto EcoMobile para o mercado e desenvolver produtos e serviços nas vertentes de *business-to-consumer*² (B2C) e *business-to-business*³ (B2B).

Na vertente de B2C pretende-se desenvolver uma solução com a capacidade de monitorizar e auxiliar a condução em tempo-real, integrar uma forte componente social que permita a partilha de viagens e conteúdos multimédia entre condutores e desenvolver soluções de BI que ajudem os condutores a perceberem a evolução da sua condução e que aspectos podem melhorar e como os podem melhorar. O principal objectivo de toda esta solução é ajudar os condutores a adoptarem uma condução mais eficiente e segura, o que se traduzirá numa poupança de cerca de 40% em combustível [2], aumento da segurança rodoviária e maior poupança na manutenção dos veículos.

A solução de B2B possui um foco mais ligado à gestão operacional e estratégica de organizações que possuem uma frota de veículos. Para além da monitorização e auxílio da condução em tempo-real, pretende-se desenvol-

¹Sentilant: <http://www.sentilant.com/>

²Business-to-consumer: comércio efectuado directamente entre a empresa produtora, vendedora ou prestadora de serviços e o consumidor final, através da Internet.

³Business-to-business: operações de compra e venda, de informações, de produtos e de serviços através da Internet ou através da utilização de redes privadas partilhadas entre duas empresas.

ver componentes de gestão de condutores, veículos e tarefas. Esta solução possuirá também uma forte componente de *reporting, analytics* e BI que ajudará as pessoas responsáveis pela tomada de decisão *just in time*. Numa organização, este género de solução, permitirá não só gerir como também maximizar uma frota de veículos e ajudar na redução dos custos operacionais, manutenção dos veículos e aumento da satisfação dos clientes.

1.2 Objectivos

Este estágio consistiu no desenvolvimento de soluções de BI do projecto Eco-Mobile, em particular para os produtos nas vertentes B2C e B2B.

Como objectivos globais do estágio foram propostos ao estagiário os seguintes:

- Projectar e desenvolver uma arquitectura de software escalável para processar e analisar dados recolhidos pelos sensores dos equipamentos móveis.
- Aprendizagem e aplicação de metodologias e processos de engenharia de software.

Como principais objectivos de trabalho, no âmbito do produto na vertente B2C, foram propostos ao estagiário os seguintes:

- Desenvolvimento de um algoritmo de classificação de trajectos semelhantes.
- Desenvolvimento de um algoritmo que permita a comparação de indicadores de condução, entre trajectos semelhantes.
- Desenvolvimento de um algoritmo que permita a comparação de indicadores de condução, entre trajectos não semelhantes mas que possuem áreas geográficas em comum.
- Implementar um *frontend* que permita a manipulação, visualização e comparação da informação produzida pelos algoritmos.

Para o produto na vertente B2B os principais objectivos de trabalho propostos ao estagiário foram:

- Projectar uma infra-estrutura que processe e guarde indicadores de condução a vários níveis de granularidade, de forma a garantir a análise exploratória da informação (e.g., *drill down, slice and dice*).

- Implementar um *front-end* que permita a criação de *dashboards*, relatórios e análises de indicadores personalizados.

1.3 Estrutura do documento

Neste primeiro capítulo pretende-se contextualizar o leitor sobre o problema, a motivação da realização deste estágio e quais os objectivos a concretizar durante o mesmo.

No segundo capítulo é demonstrado o planeamento, metodologia e processos de Engenharia de Software utilizados durante o estágio.

No terceiro capítulo é apresentado o estudo do estado da arte e análise de soluções comerciais concorrentes.

No quarto capítulo é descrito como foi elaborado o processo de levantamento de requisitos e validação, bem como a especificação de todos os requisitos funcionais e não funcionais da solução.

No quinto capítulo é realizada a descrição da arquitectura dos sistemas desenvolvidos neste estágio.

No sexto capítulo é demonstrado o trabalho desenvolvido durante o estágio.

No sétimo capítulo são demonstrados os testes funcionais e de qualidade realizados aos sistemas projectados neste estágio.

No oitavo e último capítulo referem-se as conclusões relativas ao estágio e perspectivas de trabalho futuro.

Capítulo 2

Metodologia

2.1 Metodologia de desenvolvimento

O processo de desenvolvimento utilizado neste estágio é baseado na metodologia ágil *Scrum*. No entanto a equipa de desenvolvimento adopta também algumas ferramentas do XP, nomeadamente a técnica de *Pair Programming* e do *Kanban*, o *Kanban Board*.

Durante a primeira metade do 1º semestre, foi seguida uma metodologia *waterfall* para a realização do estudo do estado da arte, análise de produtos e levantamento de requisitos. Já na segunda metade do 1º semestre, aquando a experimentação da arquitectura projectada, o estagiário foi inserido nos *sprints* de desenvolvimento [3], tendo seguido essa abordagem durante o resto do estágio. Os artefactos da metodologia ágil produzidos podem ser consultados no Anexo A.

2.1.1 Desenvolvimento ágil de software

A metodologia ágil *Scrum* é vantajosa em projectos deste género, onde os requisitos da solução são pouco estáveis. O *Scrum* foi desenvolvido com o intuito de que os projectos consigam entregar incrementos de alto valor para o cliente através de iterações, permitindo assim que o desenvolvimento da solução seja incremental, garantindo uma maior proximidade com o cliente.

Desta perspectiva torna-se possível uma gestão mais eficaz dos requisitos, uma melhor comunicação entre os elementos de toda a equipa e o cliente, e uma diminuição da centralização das responsabilidades, sendo estas distribuídas também por parte dos elementos da equipa de desenvolvimento. Ao contrário de uma metodologia *waterfall*, que pode tirar partido dos casos de uso como forma de garantir um "contracto" com o cliente, uma metodologia

ágil de desenvolvimento permite uma participação muito mais activa do cliente, o que se traduz numa melhor qualidade do produto final e satisfação do cliente.

A progressão do projecto é realizada através de um conjunto de iterações designadas de *sprints*. Para além do conjunto de práticas inerentes à metodologia, esta é também constituída por um conjunto de papéis principais pré-definidos.

Principais papéis

Product Owner É quem representa o cliente e possui uma visão sobre aquilo que ele deseja, tendo assim a responsabilidade de transmitir essa visão para toda a equipa inerente ao *Scrum*. Este prioriza também as tarefas existentes no *Product Backlog*. Neste projecto este papel é assumido pelos professores Bruno Cabral e Jorge Granjal.

Development Team A equipa de desenvolvimento é responsável pela entrega de incrementos potencialmente utilizáveis do produto no final de cada *sprint*. A equipa de desenvolvimento no projecto EcoMobile é composta por 3 elementos, como referido na secção 1.1.

Scrum Master É o responsável por garantir que a equipa de desenvolvimento cumpre as práticas inerentes ao *Scrum*, tendo também um papel importante na ajuda aos elementos da equipa de desenvolvimento por forma a que o trabalho seja feito da melhor forma possível. Este papel dentro da equipa é desempenhado pelo Rui Chicória.

Ciclo da metodologia ágil

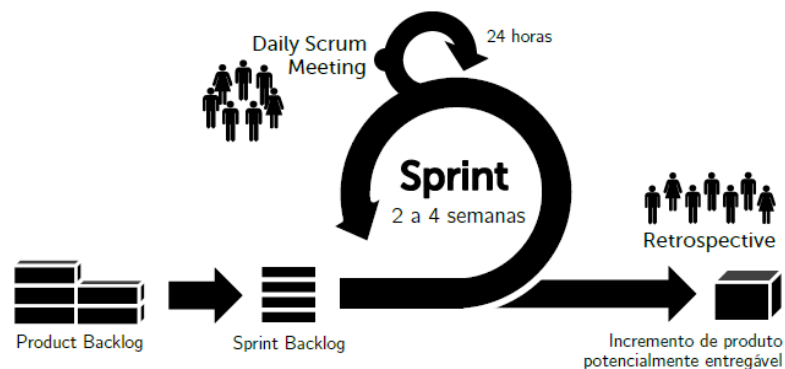


Figura 2.1: Representação da metodologia *Scrum*. Imagem adaptada [4]

Product Backlog É uma lista de tarefas definidas e priorizadas pelo *Product Owner*.

Daily Scrum Meeting É uma breve reunião realizada todos os dias à mesma hora, com duração máxima de 15 minutos, entre os elementos da equipa de desenvolvimento, com o objectivo de fazer um ponto de situação do trabalho realizado e a realizar.

Sprint Backlog Diz respeito à lista de tarefas identificadas pela equipa de *Scrum* para serem concluídas durante a *sprint*. Neste projecto estas tarefas são planeadas, sob a forma de *user stories*, e estimadas entre a equipa de desenvolvimento, consoante os itens presentes no *Product Backlog*, utilizando por vezes a técnica de *planning poker* [5] quando não existe um consenso entre todos os elementos da equipa. A duração dos *sprints* de desenvolvimento no actual projecto, têm a duração de 15 dias.

Pair Programming

O *Pair Programming* é uma técnica de desenvolvimento ágil de software, que consiste na produção de código num computador partilhado por dois membros da equipa de desenvolvimento. Aqui existem essencialmente dois papéis principais - o *driver*, que escreve o código, e o *observer*, que analisa cada linha de código produzida. Esta para além de permitir a transmissão de conhecimento, está também directamente relacionada com uma melhor qualidade de produção do código [6].

Esta foi uma prática mais recorrente no início do período de desenvolvimento, facilitando ao estagiário ambientar-se no projecto, com novas tecnologias e à equipa de desenvolvimento de uma forma mais eficiente.

Kanban Board

O *Kanban Board*, descrito na metodologia *Kanban* [7], consiste num quadro que é dividido em secções, de acordo com as várias etapas do processo de desenvolvimento, onde são colocados cartões para representar as tarefas a desenvolver. Este permite visualizar o estado actual do projecto e facilmente detectar *bottlenecks* ao longo do desenvolvimento deste.

2.2 Planeamento

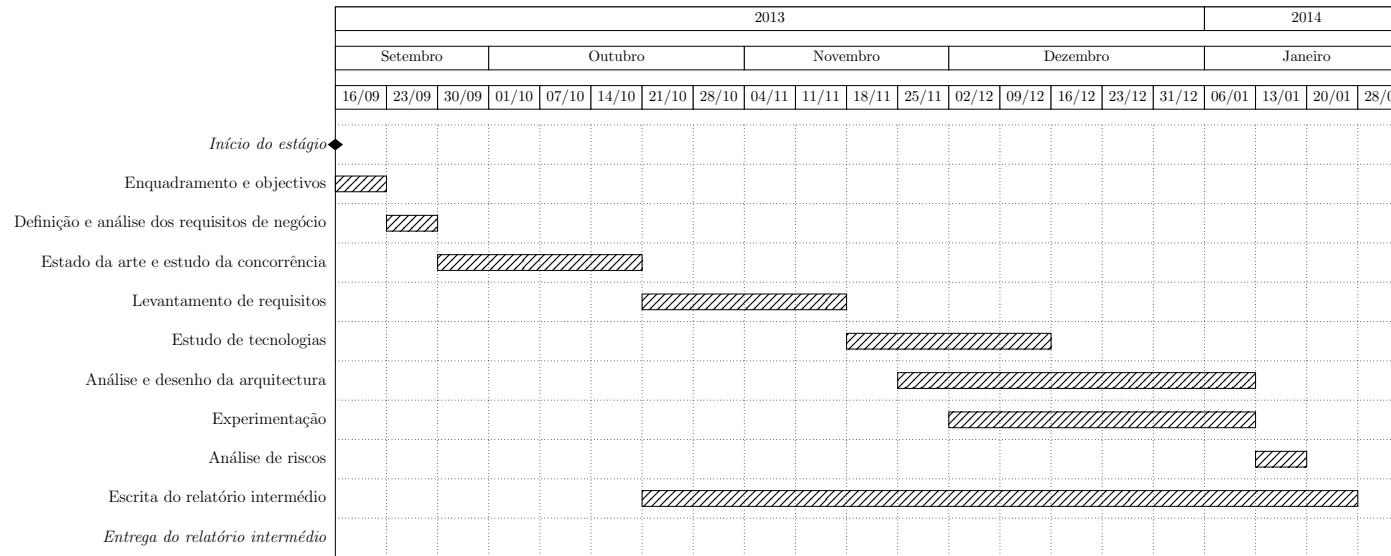
Nesta secção é apresentado o plano de trabalhos do estágio, representado através de dois diagramas *Gantt*, e os artefactos inerentes à metodologia ágil.

O plano de trabalhos durante o 1º semestre, como referido na secção 2.1, seguiu uma perspectiva mais clássica, isto é, foi mais orientado às tarefas.

Já o plano de trabalhos do 2º semestre foi elaborado com base na definição de *milestones*, uma vez que o desenvolvimento irá decorrer segundo a metodologia ágil definida na secção 2.1.1.

Foi também realizada uma análise de riscos que pretende apresentar os riscos, de maior impacto, identificados durante o estágio e a sua ocorrência ou não ocorrência. Essa análise de riscos encontra-se detalhada em anexo (Anexo B).

Planeamento do 1º Semestre



Descrição do planeamento do 1º semestre

De seguida será realizada uma descrição do trabalho realizado em cada uma das tarefas, representadas no diagrama anterior.

Enquadramento e objectivos Decorreu o processo de adaptação ao grupo de trabalho da empresa, enquadramento na visão do projecto EcoMobile e definição dos objectivos a cumprir neste estágio com os orientadores.

Definição e análise de requisitos de negócio Elaboração de um estudo sobre as necessidades de negócio existentes no mercado, por parte dos condutores e gestores de frotas.

Estado da arte e estudo da concorrência Estudo do estado da arte como forma de dar suporte ao desenvolvimento deste estágio. Realização do estudo de produtos concorrentes, análise de custos e de funcionalidades oferecidas pelas soluções existentes e respectivo estudo comparativo entre estas.

Levantamento de requisitos Início do processo de levantamento de requisitos funcionais e não-funcionais, em primeiro lugar com a visita a uma empresa ligada à gestão de frotas e por fim com a realização de um *brainstorming* entre os elementos do projecto, tendo em conta toda a informação recolhida nas fases anteriores. Elaboração dos protótipos de baixa e alta fidelidade das soluções.

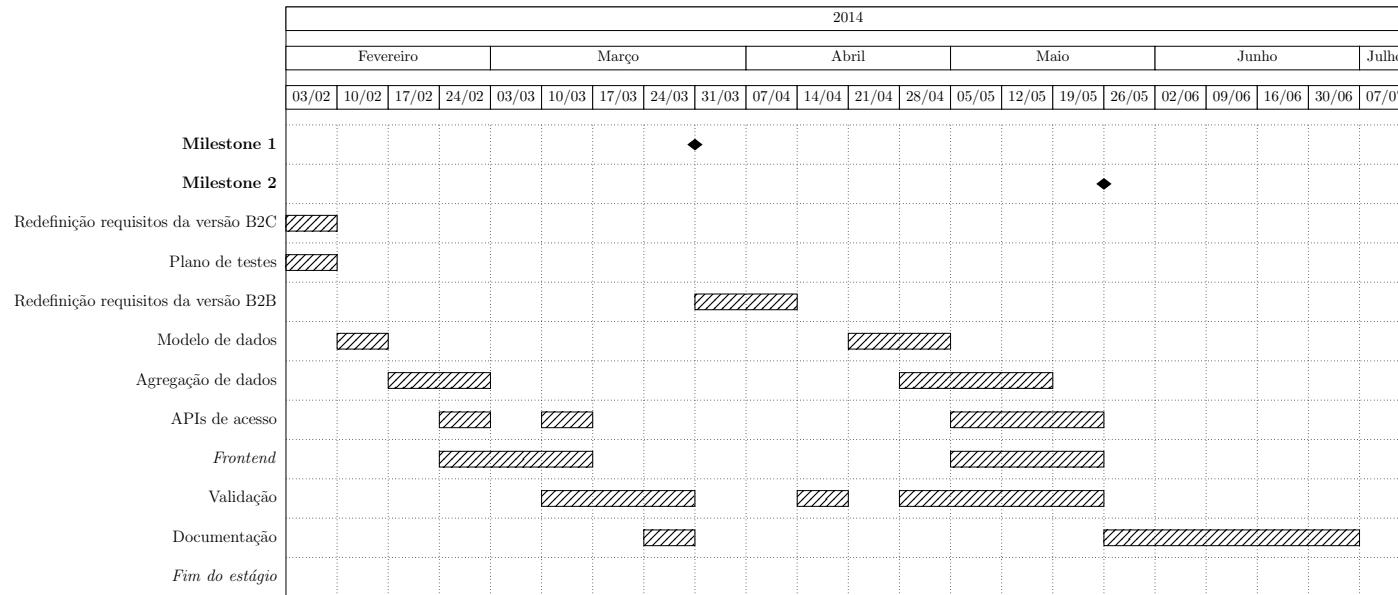
Estudo de tecnologias Foi realizado um estudo ao nível de ferramentas de *business intelligence*, de desenvolvimento web (*backend* e *frontend*) e de bases de dados relacionais e não-relacionais.

Análise e desenho da arquitectura Esta tarefa aconteceu em paralelo com o estudo de tecnologias e experimentação. Em primeiro lugar foi feita a análise da arquitectura actual do projecto EcoMobile e de seguida, com vista a realizar os requisitos funcionais e não-funcionais da vertente B2C, foi projectada a arquitectura a desenvolver no âmbito do estágio.

Experimentação À medida que a elaboração da arquitectura ia avançando, foram feitos testes para comprovar se o *design* da arquitectura se adequava ao problema. Foi também feito o estudo e implementação do algoritmo de classificação de trajectos semelhantes, construção do *backend* e respectivas APIs para comunicação da informação.

Análise de riscos Foram identificados alguns riscos que poderiam ocorrer durante a execução do estágio.

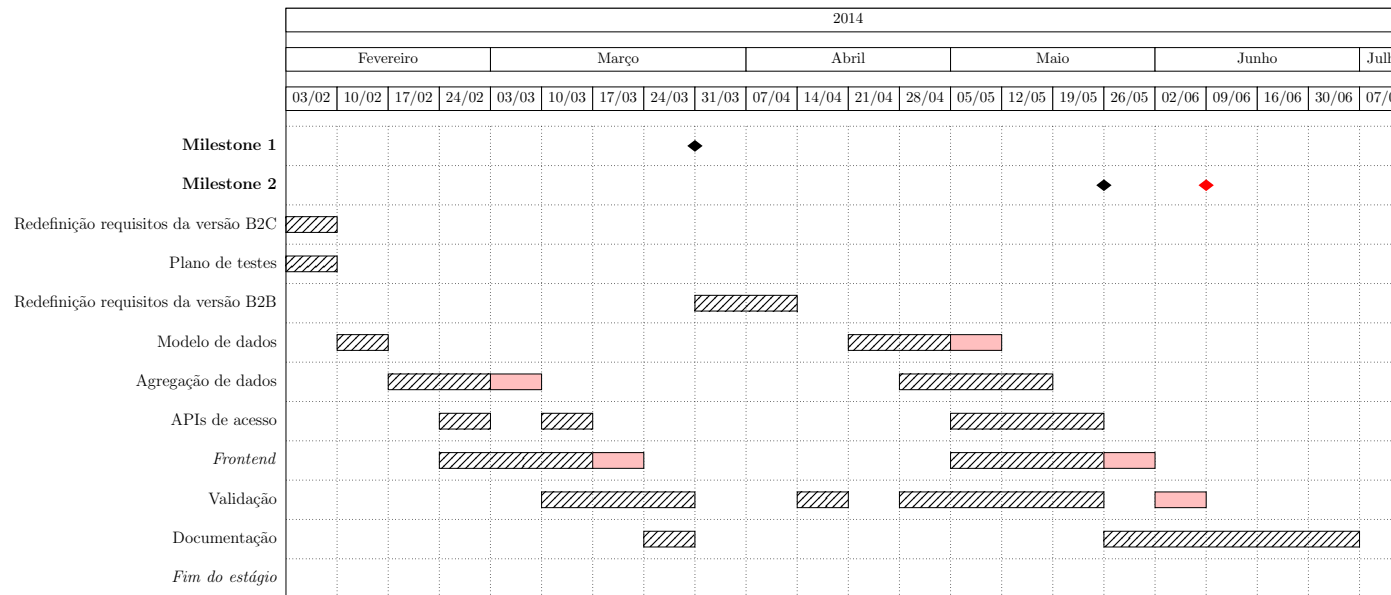
Planeamento do 2º Semestre



Milestone 1: módulo de comparação e recomendação de trajectos a ser incorporado na versão B2C.

Milestone 2: módulo de *reporting* a ser incorporado na versão de B2B

Execução do planeamento do 2º Semestre



O atraso relativamente ao planeamento da *milestone 2* ocorreu principalmente devido à redefinição da solução B2B e de não existir qualquer desenvolvimento sob esta solução. Assim, o estagiário teve de realizar toda a análise de requisitos, prototipagem e implementação da infra-estrutura base, para que fosse possível cumprir os objectivos do estágio. Os restantes desvios ocorreram principalmente devido à pouca experiência com as tecnologias presentes.

2.3 Processos de engenharia de software

2.3.1 Controlo de versões de software

Como forma de garantir uma maior qualidade no processo de desenvolvimento do software, e uma vez que o projecto EcoMobile é constituído por duas pessoas responsáveis pelo desenvolvimento web, tanto ao nível de *backend* como *frontend* (Secção 1.1), existe um processo de qualidade bem delineado para garantir a boa gestão do repositório de código e controlo de versões do software produzido.

Para este efeito é utilizado o sistema de controlo de versões Git, através da ferramenta GitLab ¹. Este é um sistema que possui uma natureza distribuída, mas que também permite a cada programador manter diversas versões do software localmente no seu ambiente de desenvolvimento (*commits*). A utilização do Git permite também agilizar o processo de desenvolvimento de diversos componentes de software de forma não linear (*branching/merging*), uma vez que permite a criação de diferentes ramos que representam diferentes funcionalidades do software [8].

No contexto deste projecto existem sempre duas *branches* remotas. Uma denominada de *staging*, que possui o código que se encontra testado e outra de *development* que possui as funcionalidades que ainda não se encontram totalmente implementadas e testadas. Quando um dos programadores começa a desenvolver novas funcionalidades no software, cria uma nova *branch* que irá conter a implementação do código necessário para o desenvolvimento dessas funcionalidades. Só será feito o *merging* da *branch* que possui as novas funcionalidades com a de *staging*, quando o código se encontrar totalmente implementado e testado.

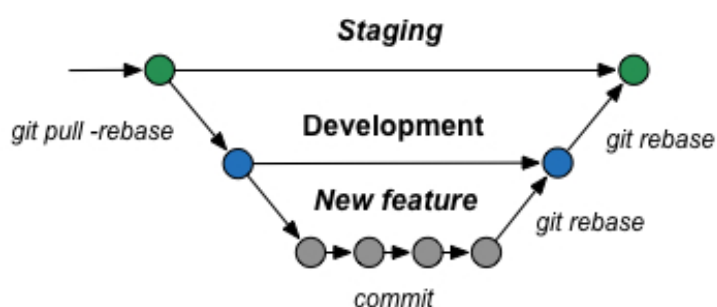


Figura 2.2: Estrutura do repositório de código e controlo de versões

¹GitLab: <http://gitlab.org/>

2.3.2 Ambientes

Existe a necessidade de no desenvolvimento do projecto existirem diferentes ambientes, para que seja possível ter cada máquina com configurações específicas para diferentes propósitos. A utilização deste processo permite aos diferentes programadores que tenham diferentes tecnologias configuradas no seu ambiente, bem como diferentes populações de dados para efeitos distintos.

Durante a realização deste estágio existiram três ambientes distintos.

Ambiente de desenvolvimento

Este é o ambiente que permite ao estagiário instalar e configurar as tecnologias necessárias para desenvolver os componentes de software necessários no âmbito do estágio. Este possui um carácter mais simples por forma a tornar o desenvolvimento mais ágil. É neste ambiente que também são realizados todos os testes unitários e os primeiros testes funcionais.

Aqui encontram-se também diferentes instâncias das bases de dados, que possuem dados gerados aleatoriamente, com a finalidade de aumentar a qualidade dos testes a realizar.

Ambiente de *staging*

É o ambiente que possui as mesmas configurações, mas não os mesmos recursos do ambiente de produção e que tem a finalidade de servir para efeitos de testes privados e demonstrações. Aqui é onde são realizados mais testes funcionais e também de usabilidade.

Ao possuir as mesmas configurações do ambiente de produção, este garante que todas as funcionalidades irão encontrar-se estáveis quando se encontrarem em produção. Este ambiente é actualizado no final de cada *sprint*, a partir da *branch* de *staging*, com o propósito de mostrar ao *Product Owner* e/ou ao cliente um incremento do produto com valor.

Ambiente de produção

Este ambiente consiste na versão que é disponibilizada para o público da aplicação. As componente aplicacionais e de bases de dados encontram-se alojadas numa *private cloud* (e.g., Linode), enquanto ficheiros de multimédia e de dados sensoriais encontram-se alojados num serviço de *storage* (e.g., Amazon S3).

O ambiente de produção é actualizado no início de cada *sprint*, com a versão da aplicação desenvolvida na *sprint* anterior, para que primeiramente

a versão de produção possa ser totalmente testada no ambiente privado (*staging*). A descrição deste ambiente será aprofundada aquando a explicação da arquitectura (Capítulo 5).

Capítulo 3

Estado da arte

Neste capítulo pretende-se apresentar o estudo do estado da arte realizado, por forma a dar a entender quais as actuais metodologias, arquitecturas e tecnologias existentes, que se enquadram no âmbito deste estágio. Será também realizada uma descrição e análise de soluções existentes, que possuem funcionalidades parecidas com a solução que se pretende desenvolver.

3.1 Metodologias e tecnologias de referência

Esta secção apresenta as metodologias, arquitecturas e tecnologias exploradas e conceptualizadas na comunidade científica, bem como alguns exemplos de aplicações práticas no mundo real. A necessidade do estudo de soluções de suporte à decisão tem como base as necessidades do projecto EcoMobile, que se pretendem ver resolvidas no âmbito deste estágio.

3.1.1 Sistemas de suporte à decisão

Cada vez mais as pessoas responsáveis pela tomada de decisão estão à procura de soluções que lhes façam chegar a informação o mais rápido possível e de forma clara e intuitiva. Desde o início da década de 1970 [9] que os sistemas de suporte à decisão (DSS) estão presentes nas organizações. Estes são uma classe específica dos sistemas de informação, que têm a finalidade de dar suporte ao negócio e ajudar a realizar uma melhor tomada de decisão. Este género de sistemas são adequadamente projectados para dar aos utilizadores a capacidade de compilar informações úteis a partir de dados em bruto, provenientes de diversas fontes de informação, por forma a ajudar a identificar e resolver problemas dentro da menor janela temporal possível.

Estes sistemas de informação tiveram uma forte componente de investigação na comunidade científica e empresarial, tendo resultado numa ampliação e evolução do leque de aplicações de suporte à decisão: *model-driven DSS*, *data-driven DSS*, *communication-driven DSS*, *document-driven DSS*, *knowledge-driven DSS*, *web-based DSS* [10].

Data warehousing

Uma vez que no contexto do estágio se pretendiam desenvolver soluções de processamento, armazenamento e análise de grandes volumes de informação, decidiu-se realizar um estudo numa das principais áreas de suporte ao processamento de informação e análise de dados consolidados, as *data warehouses*.

Na década de 1990 surgiu, por Ralph Kimball [11], uma das principais metodologias e arquitecturas para o desenvolvimento, evolução e ampliação dos sistemas de suporte à decisão, as *data warehouses*, que definiram assim uma categoria mais ampla para os modelos *data-driven*. Uma *data warehouse* é uma colecção de dados não volátil e variante ao longo do tempo, que permite integrar a informação extraída de diferentes fontes de dados (e.g., bases de dados operacionais, documentos, ficheiros de *logging*) e armazená-la num modelo multidimensional que permite realizar consultas de forma eficiente sobre a informação agregada.

O modelo multidimensional é um modelo de dados o mais desnormalizado possível, que permite representar as dimensões de um negócio (e.g., tempo, clientes, vendas) e relacioná-las com os factos correspondentes, possibilitando assim inúmeras combinações.

Um dos processos mais importantes em projectos de *data warehouse*, é o denominado processo de Extração-Tranformação-Carregamento (ETL), onde, segundo Ralph Kimball, é durante a elaboração deste processo que cerca de 70% dos projectos de BI falha.

Para analisar toda a informação histórica presente numa *data warehouse* é frequentemente utilizada uma ferramenta de *Online Analytical Processing* (OLAP). A análise OLAP, criada por Nigel Pends [10], permite aos responsáveis pela tomada de decisão ganhar um maior conhecimento através do acesso à informação sobre diversos pontos de vista.

Na figura 3.1 pode observar-se a arquitectura de um sistema de suporte à decisão que utiliza uma *data warehouse*. Este é constituído pelo *back room*, que é o componente responsável por todo o processo que é transparente para o utilizador final. É aqui que é realizado o processo de ETL, o carregamento dos dados para a *data warehouse* e a criação de diferentes cubos OLAP para disponibilizar a informação sobre diferentes perspectivas. No *front room* é onde são criados os *dashboards* e relatórios para visualização da informação,

bem como utilizadas ferramentas de produtividade para criação de análises personalizadas e aplicação de algoritmos de *data mining* sobre a informação.

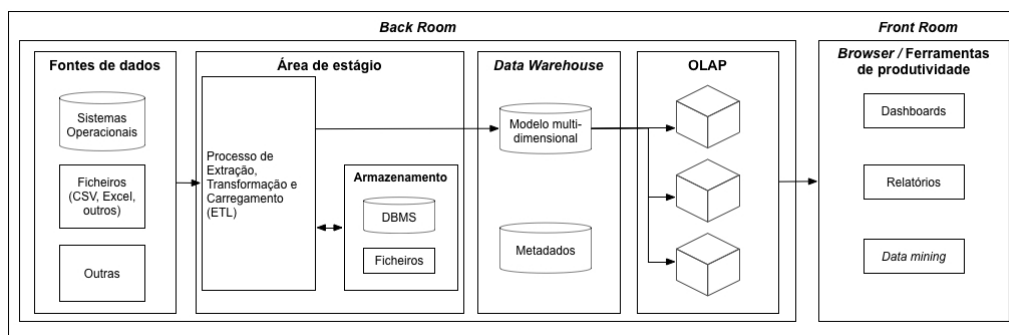


Figura 3.1: Arquitectura de um sistema de suporte à decisão utilizando uma *data warehouse* [12]

3.1.2 *Business intelligence*

Business Intelligence (BI) é um conjunto de teorias, metodologias, processos, arquiteturas e tecnologias que permitem a transformação de dados em bruto, em informações significativas e úteis para o negócio com o objectivo de proporcionar uma vantagem competitiva no mercado. As organizações têm vindo a utilizar BI ao longo de muitos anos para monitorização, criação de relatórios e realização de análises exploratórias sobre a informação com a finalidade de melhorar o desempenho dos seus processos de negócio. A este género de BI dá-se o nome de *strategic BI* [13].

A maior parte das aplicações de BI têm-se vindo a centrar na gestão de planos estratégicos, isto é, tem tido como principal finalidade extrair, transformar e carregar a informação ao final de um determinado período de tempo (e.g., dia, mês, trimestre, semestre, ano) e assim produzir indicadores que permitam a avaliação do negócio nesse determinado *snapshot*.

No entanto as necessidades têm-se vindo a alterar. A informação cresce e move-se a uma velocidade cada vez maior, onde cada vez mais tempo é dinheiro e obter uma vantagem competitiva no mercado o mais rapidamente possível é uma prioridade. Neste sentido estamos a falar na necessidade crescente de obter informação de mais fontes de dados (estruturadas, semi-estruturadas ou não estruturadas) em tempo-real ou quase em tempo-real. Este género de BI é denominado de *operational BI* [13] e tem como principal preocupação a gestão e optimização das operações diárias de um negócio, uma vez que o seu principal objectivo é fornecer a informação certa no tempo

certo para os utilizadores, para que estes consigam reagir mais rapidamente aos problemas e tornarem-se mais competitivos.

Tabela 3.1: Características de dois géneros de BI

	<i>Strategic BI</i>	<i>Operational BI</i>
Objectivo	Atingir objectivos a longo prazo	Gerir e otimizar operações diárias
Período	Dias, semanas ou meses	No decorrer do dia
Informação	Indicadores históricos	Indicadores em tempo-real ou quase em tempo-real

Tendo em conta os dois pontos de vista apresentados anteriormente, as suas aplicações dependem dos requisitos de cada caso em específico. Por exemplo, a exigência de tempo de acção num sistema de detecção de fraude deve rondar os segundos, enquanto que num sistema de gestão de stocks pode rondar os minutos ou horas.

No entanto, isto não significa que os sistemas de suporte à decisão que fazem uso de uma *data warehouse* estejam a ser descontinuados. Muito pelo contrário, as *data warehouses* são uma tecnologia perfeitamente comprovada [14] e com casos de uso de sucesso no mundo real [15]. Contudo tem surgido nos últimos tempos novas tecnologias e metodologias, capazes de responder às novas necessidades, e de conjugar o melhor dos dois géneros.

Com o volume de informação a crescer a um ritmo elevado, o termo *Big Data* tem ganho cada vez maior destaque e com este surgiram novas tecnologias e metodologias (e.g., NoSQL, MapReduce [16]), capazes de responder a estas novas necessidades e de conjugar o melhor dos dois géneros de BI [17].

3.1.3 *Data mining*

Um dos principais objectivos do projecto EcoMobile é permitir ao condutores avaliarem a evolução da sua condução ao longo dos trajectos mais comuns. Uma vez que no âmbito deste estágio se pretende desenvolver funcionalidades que permitam a classificação de trajectos semelhantes, como forma de avaliar a evolução da condução, realizou-se um estudo sobre uma das principais áreas de descobrimento de padrões na informação, a área de *data mining*.

Data mining refere-se ao processo computacional de descobrir padrões existentes em grandes conjuntos de dados através da utilização de métodos de inteligência artificial, estatísticos e *machine learning*. Inicialmente este

termo era utilizado principalmente por pessoas do campo da matemática. No entanto, ganhou bastante popularidade, e expandiu as suas potencialidades para o campo das bases de dados [18]. Esta é uma área bastante complexa e que possui bastantes vertentes [19]. No âmbito deste trabalho será dada uma ideia geral do processo para a aplicação de técnicas de *data mining*, mas o foco principal incidirá na utilização de algoritmos de agrupamento de dados (*clustering*).

O processo de descobrir conhecimento presente nas bases de dados (KDD), segundo o modelo CRISP-DM [20], é definido em seis fases. Neste processo existe o pressuposto de que em algumas fases pode existir a necessidade de reverter para uma das fases anteriores.

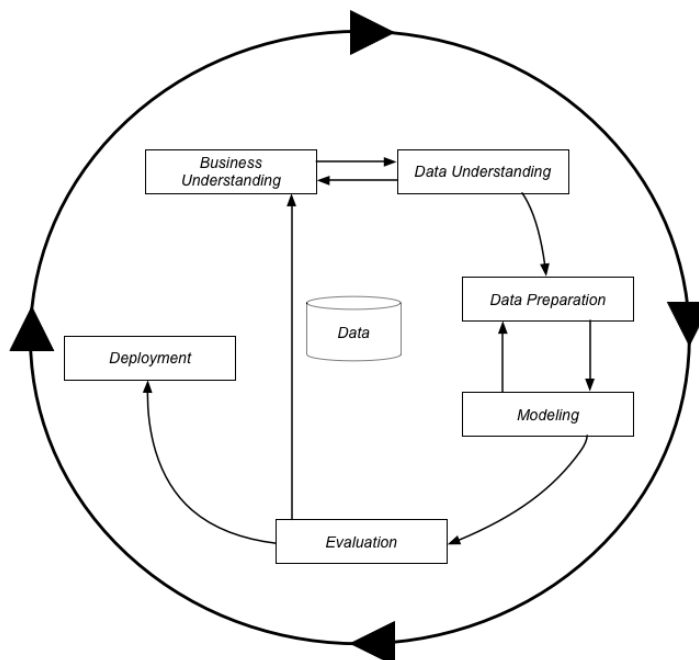


Figura 3.2: Fases do processo de *data mining* [20]

1. **Perceber o negócio (*Business understanding*):** A fase inicial foca-se essencialmente na percepção dos objectivos e requisitos do projecto sob o ponto de vista das pessoas do negócio, para que de seguida seja possível traduzi-los num problema de *data mining*. Nesta fase é elaborado o planeamento, tendo em conta as necessidades, para que os objectivos sejam cumpridos.
2. **Perceber os dados (*Data understanding*):** Nesta tarefa pretende-se que se faça uma selecção das fontes de dados, para que se proceda à

recolha de dados e numa primeira instância se realize a familiarização com os dados. Aqui deve também tentar-se detectar subconjuntos de informação que formem possíveis hipóteses de informação oculta.

3. **Preparação dos dados (*Data preparation*):** A fase de preparação dos dados abrange todas as actividades para a construção do *dataset* final, sob os dados obtidos a partir do conjunto de dados brutos iniciais.
4. **Modelação (*Modeling*):** Várias técnicas de modelação são seleccionadas e aplicadas, tendo em conta o tipo de problema, e os seus parâmetros são calibrados por forma a obter os melhores resultados. Existem diversas técnicas de *data mining* possíveis para o mesmo tipo de problema, porém cada técnica tem requisitos específicos sobre a forma dos dados o que pode levar a que nesta fase se tenha de recuar novamente para a fase de preparação dos dados (Figura 3.2).
5. **Validação (*Evaluation*):** Antes de prosseguir para o *deployment* do modelo criado, é importante realizar uma boa validação do modelo e se este cumpre todos os requisitos para o negócio. Consoante o modelo criado, deve-se escolher a técnica de validação que melhor se adapta ao modelo de *data mining* construído.
6. **Deployment:** Após a criação do modelo e a sua respectiva validação, o *deployment* deste terá de ter em conta os requisitos do negócio. Para além do modelo criado aumentar o conhecimento sobre a informação, este conhecimento deverá ser organizado e apresentado de uma forma intuitiva para o cliente, para que a sua utilização seja perceptível.

Clustering

A quantidade de informação encontrada e armazenada nos dias que correm é cada vez maior [21]. Uma das formas de lidar com a organização de grandes conjuntos de dados, e que sempre desempenhou um papel importante e indispensável ao longo do desenvolvimento da sociedade [22], é classificar e agrupar esses dados em grupos (*clusters*) que possuem características em comum (similaridade).

A similaridade entre os objectos é definida utilizando uma medida de distância, que é escolhida consoante as características do problema. A aplicação típica geralmente envolve espaços Euclidianos de baixa dimensionalidade, onde se utilizam as chamadas distâncias Euclidianas (e.g., distância Euclidiana, distância de Manhattan). No entanto, existem novos problemas que requerem outras técnicas devido à sua complexidade. Estes podem envolver espaços Euclidianos de dimensão muito elevada ou espaços que não

são totalmente denominados de Euclidianos (e.g., criar *clusters* de documentos através dos seus tópicos mais comuns) e requerem medidas de distância diferentes, as chamadas medidas não Euclidianas (e.g., distância de Jaccard, distância de Hamming, distância de edição) [23].

Os algoritmos de *clustering* podem ser definidos em dois grupos que seguem duas estratégias diferentes para a elaboração de conjuntos [23].

Hierárquico ou aglomerativo Inicialmente cada ponto começa no seu próprio *cluster*. De seguida os *clusters* são combinados tendo em conta a sua proximidade, definida pela medida de distância seleccionada. A combinação pára quando se atingir um determinado número de *clusters* ou a coesão entre *clusters* for considerada aceitável.

Point assignment Consideram-se os pontos partindo de uma ordem arbitrária onde cada um desses pontos é atribuído ao *cluster* que melhor se adapta. Neste processo existe uma curta fase inicial onde são estimados conjuntos iniciais, o que ocasiona a existência tanto da atribuição como da divisão (no caso da existência de *outliers*) durante classificação dos restantes pontos. Exemplo: *K-means* [23]

O processo de elaboração e análise de *clusters* é composto por quatro fases, sendo que este tem como base o processo representado na figura 3.3 [24].

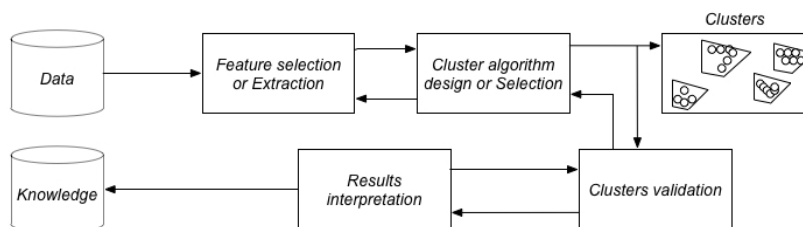


Figura 3.3: Procedimento para elaboração de *clusters* [24]

1. **Seleção ou extracção de características (*Features selection or Extraction*):** A selecção de características diz respeito à escolha de características distintivas de um conjunto de candidatos, enquanto que a extracção de características utiliza transformações sobre a informação para gerar novas características a partir das originais.
2. **Seleção ou implementação do algoritmo (*Cluster algorithm design or Selection*):** Inicialmente é feita a selecção da medida de proximidade, que melhor se adapta às características do problema, e de

seguida aplicado um ou vários algoritmos de *clustering* para realizar o agrupamento segundo o critério de proximidade escolhido. Neste passo poderá existir a necessidade de retornar à fase anterior, caso exista a necessidade de seleccionar ou extrair mais características.

3. **Validação (*Clusters validation*):** Após a criação dos agrupamentos, é necessário realizar uma validação dos resultados, por forma a perceber se existe a necessidade de fazer ajustamentos nos parâmetros relacionados com o algoritmo de classificação. Em geral, existem três categorias de critérios de teste: índices externos, índices internos e índices relativos [25].

- **Índices externos:** baseiam-se numa estrutura pré-definida, que é o reflexo do conhecimento prévio sobre os dados, que é utilizada como um padrão para validar os *clusters* criados (e.g., Entropia).
- **Índices internos:** estes não têm por base um conhecimento prévio sobre determinados padrões sobre a informação, mas sim a estrutura presente sobre os dados originais (e.g., *Sum of Squared Error* (SSE)).
- **Índices relativos:** é uma abordagem híbrida, uma vez que permite comparar estruturas pré-definidas com as estruturas dos dados originais, por forma a decidir qual poderá revelar as melhores características dos objectos (e.g., Entropia ou SSE).

Caso a validação do algoritmo de classificação não obtenha bons resultados, deverá retornar-se a fase de *design* e rever os critérios e parâmetros do algoritmo.

4. **Interpretação dos resultados (*Results interpretation*):** O objectivo final é proporcionar um conhecimento significativo, obtido a partir dos dados originais, de modo a que se possa efectivamente tirar partido da solução criada.

3.1.4 NoSQL

Os sistemas de bases de dados relacionais (DBMS) encontram-se presentes no mundo *enterprise* desde os anos 70 [26]. Temos visto diversas mudanças ao nível das linguagens de programação, arquitecturas de software, plataformas e processos. No entanto houve algo que sempre se manteve constante - bases de dados relacionais armazenam os dados. Estas possuem uma enorme estabilidade no armazenamento da informação, o que sempre deu um grande valor e segurança para o desenvolvimento de qualquer aplicação.

No entanto, como já referido na secção 3.1.2 e também com o aparecimento da Web 2.0 [27], existe o desafio em lidar com o crescente volume de informação. As bases de dados NoSQL vêm dar uma resposta a este problema, uma vez que estas possuem características como:

- Capacidade de se replicar e distribuir horizontalmente ao longo de muitos servidores.
- Capacidade de replicar e distribuir operações simples (e.g., escritas, leituras, agrupamentos) ao longo de muitos servidores, que numa base de dados relacional seria um problema complexo de lidar.
- Utilização eficiente de índices distribuídos e RAM para o armazenamento de dados.
- Facilidade em adicionar dinamicamente novos atributos e registos.
- Um modelo de concorrência mais fraco, comparado ao das bases de dados relacionais (ACID).

Principais modelos de dados

Este género de bases de dados possui quatro categorias principais que permitem a implementação de diferentes modelos de dados sem relações físicas.

Key-Value Store É utilizada principalmente quando todo o acesso à base de dados é feito através de uma chave primária. Tipicamente, estas disponibilizam apenas operações para obter e eliminar uma chave e podem guardar desde simples valores inteiros, a complexas estruturas de dados. Estas limitam-se apenas a guardar aquilo que a aplicação lhes fornece, sendo que é da responsabilidade da aplicação perceber o que foi guardado. Exemplos: Riak, Redis, MemcacheDB, Berkley DB, HamsterDB, Amazon DynamoDB e Voldemort.

Document-Oriented O conceito principal é guardar e disponibilizar documentos no formato XML, JSON, BSON, entre outros. Estes são auto-descritivos e podem ser organizados em mapas, colecções ou valores escalares. Os documentos guardados numa mesma colecção podem possuir *schemas* diferentes. Exemplos: MongoDB, CouchDB e SimpleDB.

Column-Family Store Existem bastantes tipos de bases de dados colunares. Mas, em geral, estas permitem armazenar dados mapeados por chave e agrupados por famílias de colunas. Têm como principal característica a sua eficiência na agregação dos dados sobre pequenos subconjuntos de toda a informação. Exemplos: Cassandra, HBase, Hypertable, BigTable e Vertica.

Graph Database Têm por base as conhecidas estruturas em grafo. Permitem armazenar entidades e relações entre essas entidades, sendo que as entidades são conhecidas por nós que possuem propriedades e as relações podem ser vistas como vértices que possuem uma direcção de significância. Exemplos: Neo4J, Infinite Graph, OrientDB e FlockDB.

Teorema de CAP

A utilização de uma base de dados com estas características deve ser ponderada com as propriedades do problema, os requisitos da solução a desenvolver e tendo sempre em conta um *trade-off* conhecido como teorema de CAP [28]. Este teorema diz que dadas três propriedades (*Consistency*, *Availability* e *Partition Tolerance*), apenas se podem garantir duas propriedades.

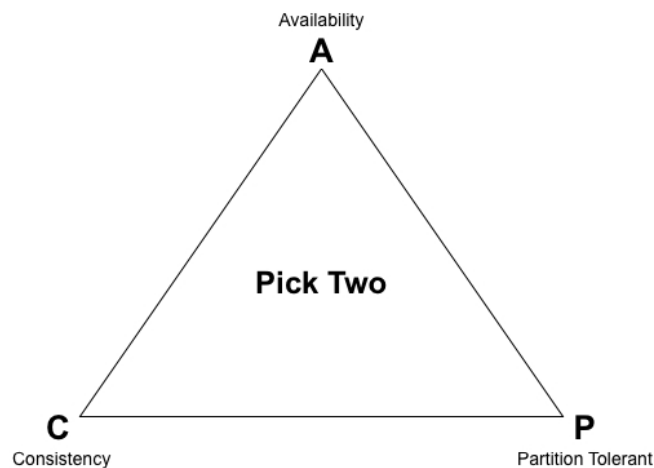


Figura 3.4: Representação visual do teorema de CAP

Consistency Significa que cada servidor retorna a resposta certa para cada pedido. No entanto o significado de consistência depende do serviço que é disponibilizado, ou seja, poderão existir eventualmente várias respostas correctas possíveis.

Availability Esta propriedade simplesmente significa que cada pedido, eventualmente, deve receber uma resposta. Obviamente, obter uma resposta rápida é melhor do que uma resposta lenta, mas segundo o propósito do teorema de CAP, o facto de exigir uma resposta é o suficiente para criar um problema, pois em grande parte dos sistemas reais uma resposta demorada é equivalente a não existir uma resposta.

Partition Tolerant O *cluster* continua a funcionar mesmo que existam quebras na comunicação física entre as várias partições existentes.

Como observado no guia visual (Figura. 3.4), confirma-se que os sistemas de bases de dados apenas podem garantir duas das três propriedades: CA (*Consistency, Availability*), CP (*Consistency, Partition Tolerant*) e AP (*Availability, Partition Tolerant*). De seguida será demonstrado, segundo o teorema de CAP, onde se enquadra cada uma das bases de dados NoSQL em cima exemplificadas, bem como as bases de dados relacionais (e.g., MySQL, PostgreSQL).

Tabela 3.2: Propriedades garantidas pelas bases de dados (CAP)

Propriedade	Base de dados	Observação
CA	PostgreSQL, MySQL, Vertica	Demonstram problemas com o particionamento. Lidam com o problema através de replicação.
CP	MongoDB, HBase, Hypertable, BigTable, Redis, MemcacheDB, Berkeley DB	Apresentam problemas com a disponibilidade, enquanto mantêm os dados consistentes entre os nós particionados.
AP	DynamoDB, Voldemort, Cassandra, CouchDB, Riak, SimpleDB	Conseguem alcançar uma "eventual consistência", através da replicação e verificação.

Aplicação

Este novo paradigma de bases de dados trouxe novas potencialidades para complementar os sistemas de *analytics* e BI. Organizações como o Facebook, Twitter e Netflix incorporam nas suas arquiteturas de BI algumas destas novas tecnologias para que assim consigam tirar um maior partido e valor da informação no momento certo [29] [30] [31].

Na figura 3.5 pode observar-se o exemplo de um sistema cujo a sua arquitetura utiliza as potencialidades das *data warehouses*, para elaboração de análise OLAP sobre a informação, como também utiliza novos mecanismos para extrair grandes volumes de informação (MapReduce) proveniente de fontes de dados não estruturadas (e.g., redes sociais) para enriquecer todos os dados presentes na *data warehouse*. Uma vez que o ETL é um processo

batch que corre geralmente ao fim do dia, é utilizada uma base de dados NoSQL (HBase) que permite armazenar a informação assim que esta se encontra disponível, permitindo que a janela temporal da sua disponibilização seja reduzida.

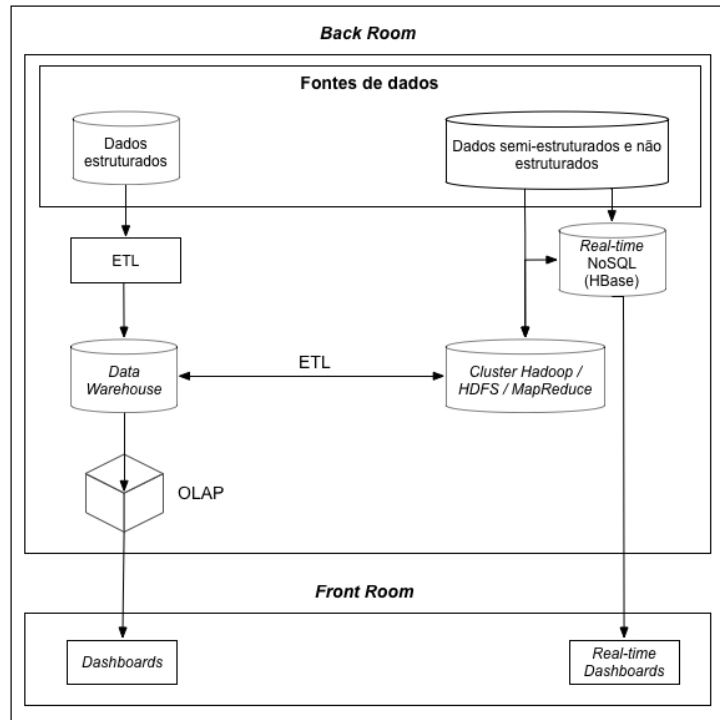


Figura 3.5: Exemplo de uma arquitectura para "Big Data Analytics" [32]

3.1.5 Cloud computing

Pretende-se que as duas vertentes dos produtos a desenvolver, no contexto do projecto EcoMobile, possuam a capacidade de se adaptarem consoante as necessidades de negócio e que também sejam disponibilizadas para qualquer utilizador como um serviço na Internet. Tendo em conta estas necessidades foi realizado um estudo na área de *cloud computing*.

Cloud computing é um termo genérico que envolve a disponibilização de serviços que se encontram alojados na Internet. A Gartner define *cloud computing* como um estilo de computação com capacidades massivamente escaláveis para entrega de serviços aos clientes através da Internet [33]. Este tem como objectivo mais amplo disponibilizar grande poder computacional para as massas e permitir às organizações crescerem consoante as suas necessidades (*pay-as-you-go*).

Benefícios

Nos últimos anos temos assistido a uma crescente "migração" das organizações para ambientes *cloud*. As principais razões que levam cada vez mais e mais organizações a optarem por este estilo de computação incluem [34]:

- Redução dos custos de aquisição e manutenção de equipamentos.
- Redução dos custos associados com a entrega de serviços de TI, obtendo assim a possibilidade de desviar recursos para outras atividades, tal como a integração de serviços.
- Redução da responsabilidade da gestão de TI, permitindo ao pessoal-chave da organização concentrar-se mais na produção e inovação.
- Aumento da agilidade dos negócios, permitindo assim que as organizações cumpram satisfatoriamente as necessidades em ambientes que mudam rapidamente.

Características

Organizações como a IBM, Dell, Microsoft, Google e Amazon já começaram a tomar fortes posições no que diz respeito à disponibilização e prestação de serviços em ambientes *cloud* [35]. As principais características que os ambientes *cloud* garantem incluem [34]:

- Serviços de computação *on-demand* que permitem aos utilizadores consumir recursos (e.g., aplicações, armazenamento) sempre que necessário.
- Garantias de recursos *multi-tenancy*¹ e *pooling* que permitem combinar recursos computacionais de forma heterogênea (e.g., hardware, software, processamento, largura de banda) e responder a diversos clientes, de forma dinâmica.
- Elasticidade e escalabilidade de forma rápida e tendo em conta as necessidades.
- Simplicidade na manutenção e monitorização das infra-estruturas.

¹Multi-tenancy: é um modo de operação de software que permite, a várias instâncias independentes de uma ou várias aplicações, operarem num ambiente compartilhado. As instâncias (*tenants*) encontram-se logicamente isoladas, mas fisicamente partilhadas [36].

Abordagens de *deployment* - Modelos

As abordagens de *deployment* em ambientes *cloud* representam tipos específicos de modelos, isto é, a forma como um ambiente *cloud* disponibiliza os serviços pode variar. As organizações escolhem o modelo que melhor se adapta ao seu negócio e requisitos operacionais. Tipicamente estes modelos podem ser classificados como *Public Clouds*, *Private Clouds* e *Hybrid Clouds* [34].

Public Clouds É uma infra-estrutura na *cloud* que disponibiliza, através de terceiros, serviços para o público em geral ou para um determinado conjunto de organizações. Toda a instalação e manutenção destas infra-estruturas fica a cargo do provedor de serviços. Tipicamente os consumidores, neste modelo, podem usufruir da vantagem *pay-as-you-go* uma vez que apenas são cobrados os recursos e serviços que cada um especificamente utiliza. Exemplos: Amazon Web Services, Microsoft Azure, Google Apps

Private Clouds Este género de modelo, diz respeito a uma infra-estrutura que é operada somente por uma organização ou público específico. A instalação e manutenção destas infra-estruturas fica a cargo de quem opera os serviços (e.g., departamento de TI). Uma das vantagens deste género de modelo é permitir às organizações terem um melhor controlo no que toca a questões de segurança e questões de conformidade regulamentar.

Hybrid Clouds Estas são uma combinação entre uma *private cloud* e uma *public cloud*. Combinam a capacidade externa com os recursos locais existentes. Neste caso, as responsabilidades de gestão são divididas entre os dois provedores de *cloud*. Este modelo tem como principal objectivo permitir que as organizações possam migrar para um ambiente *cloud*, mas que consigam manter os seus dados confidenciais e aplicações de missão crítica na sua *cloud* privada e migrar outras aplicações para uma *cloud* pública.

Abordagens de *deployment* - Serviços

Os ambientes *cloud* têm como núcleo, três géneros de serviços: *Software as a Service* (SaaS), *Platform as a Service* (PaaS), *Infrastructure as a Service* (IaaS).

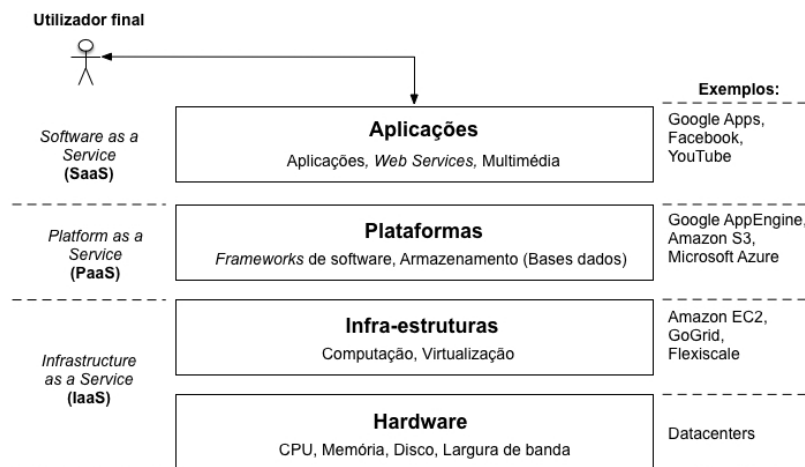


Figura 3.6: *Cloud computing* [37]

Software as a Service (SaaS) Refere-se a funcionalidades pré-construídas e independentes que se encontram universalmente disponíveis e são disponibilizadas aos clientes como um serviço na Internet. Quando os clientes procuram estes serviços, pretendem pagar pelas funcionalidades consoante as suas necessidades (quando estas são comercializadas). As vantagens desta abordagem incluem a rápida disponibilização de funcionalidades bem testadas, uma implementação mais ágil, escalabilidade, confiabilidade e facilidade de gestão.

Platform as a Service (PaaS) Refere-se a software e ferramentas de desenvolvimento de um produto que os clientes alugam, para que possam implementar as suas próprias aplicações para uso específico. Os clientes que procuram estes serviços pretendem reduzir o tempo de desenvolvimento e *deployment* das suas aplicações, porém a adesão é por vezes condicionada pelos elevados custos. No entanto estas plataformas incluem suporte para o desenvolvimento, redução dos custos de desenvolvimento e manutenção e escalabilidade.

Infrastructure as a Service (IaaS) É essencialmente a disponibilização de hardware, geralmente na forma de um ambiente virtualizado com armazenamento, processamento e largura de banda associados, que permite a comunicação e operação entre aplicações. Neste caso, o cliente possui um controle total sobre toda a infra-estrutura e têm de garantir toda a configuração e manutenção. No entanto permite ao cliente possuir a sua aplicação dividida por diversos provedores de serviços *cloud* de forma transparente para quem as utiliza.

3.2 Análise de soluções comerciais

Neste sub-capítulo é feita uma análise de algumas soluções comerciais que efectuam a monitorização da condução e fornecem estratégias para ajudar tanto os condutores a tornarem a sua condução mais eficiente, como os gestores de frotas a tomarem as melhores decisões estratégicas e operacionais para a sua organização.

Durante a elaboração desta análise foram identificadas duas vertentes de soluções. Soluções que necessitam de hardware dedicado, que tem de ser instalado individualmente em cada veículo, para monitorizar e comunicar os dados relativos à condução de cada veículo e soluções que apenas utilizam um *smartphone* para realizar a análise e comunicação dos dados da condução de cada veículo.

TomTom WEBFLEET

A TomTom [38] é uma empresa que possui bastante experiência no desenvolvimento de produtos de navegação GPS, sendo que os seus produtos são comercializados por todo o mundo. No contexto da TomTom foi criada a TomTom Business [39], cuja finalidade é criar serviços que respondam às necessidades de gestão de frotas, através da monitorização de veículos utilizando hardware dedicado. Esta possui uma solução denominada de TomTom WEBFLEET [40], uma aplicação que permite gerir uma frota de veículos a partir de qualquer computador com ligação à Internet e que disponibiliza as funcionalidades básicas de gestão de veículos/frotas tais como localização de veículos em tempo-real, gestão de quilometragem, velocidade e tempos de viagem, recomendação de rotas, emissão de relatórios e comunicação *full-duplex*.

Para além das funcionalidades anteriores, o TomTom WEBFLEET possui um componente de exploração de informação, que permite a criação de um *dashboard* personalizado (Figura 3.7) onde é permitido analisar e explorar um conjunto de indicadores, como tempo em excesso de velocidade, consumo médio de combustível, pegada carbónica, eventos de condução por dia e por 100 quilómetros, tempo total em excesso de rotações e tempo total parado durante as viagens.



Figura 3.7: *Dashboard* da solução TomTom WEBFLEET

Fleetmatics

A Fleetmatics [41] é uma empresa que desenvolve soluções de gestão de frota, oferecidas como *Software as a service* (SaaS). A solução da Fleetmatics permite monitorizar, após instalação de hardware nos veículos, a localização em tempo-real dos veículos, velocidades, distâncias percorridas, consumo de combustível e eventos de condução (e.g., excesso de velocidade, travagens e curvas bruscas). Existe também a possibilidade de incorporar esta solução com dispositivos de navegação GPS o que permite ao condutor comunicar com a central e obter os melhores trajectos para determinadas rotas.

De toda a informação que é monitorizada nos veículos, é criado um conjunto de indicadores (e.g., eventos de condução, consumos de combustível, emissões de CO₂, violações do limite de velocidade, quilometragem, produtividade dos condutores), que se encontram disponíveis através de um *dashboard* (Figura 3.8) para permitir a criação de análises e relatórios personalizados.

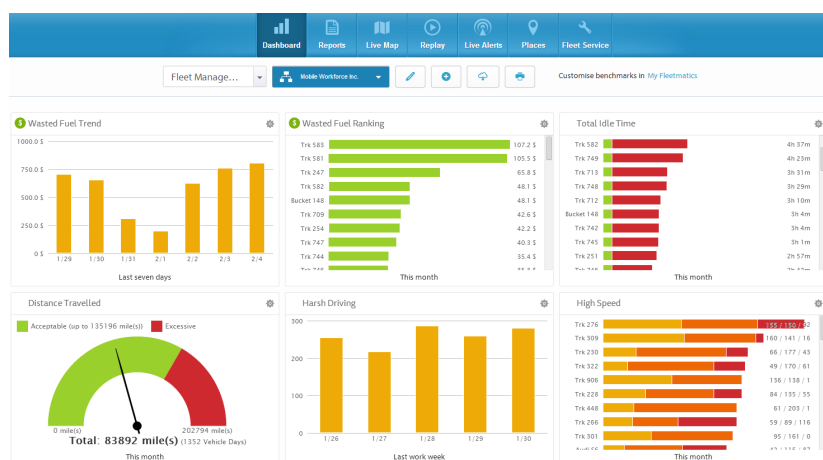


Figura 3.8: *Dashboard* da solução Fleetmatics

Frotcom

A Frotcom é uma empresa com actual presença em 30 países, que em Portugal se faz representar pela distribuidora Frotcom Lusitana [42], cujo o seu principal objectivo é ajudar as organizações a obterem os melhores resultados da sua frota, através de soluções inteligentes de localização, monitorização e gestão de frotas. A solução desenvolvida pela Frotcom faz uso de hardware que tem de ser instalado nos veículos, para que assim seja feita a monitorização e comunicação dos dados de condução. Os dados monitorizados nos veículos podem ser consultados a partir de um serviço na web (Figura 3.9) que permite localizar em tempo-real os veículos, saber o estado da ignição, visualizar graficamente dados relativos a velocidades, quilometragem e outros sensores dos veículo. Esta solução permite realizar uma análise de custos da frota, através de um conjunto de indicadores criados a partir do consumo de combustível, quilometragem e tempo em elevada rotação do veículo, bem como permite também a integração com navegadores GPS para permitir a comunicação com os condutores e optimização de rotas.



Figura 3.9: Solução Frotcom

Automatic

A Automatic Labs, Inc. [43] é uma empresa com foco no mercado Norte Americano, que desenvolveu uma aplicação móvel para sistemas operativos móveis Android e iOS, com o principal objectivo de se tornar num assistente pessoal de condução. Esta aplicação funciona em conjunto com um pequeno dispositivo, denominado de Automatic Link, que faz a ligação ao computador de bordo do veículo e comunica a informação ao *smartphone*.

Durante a viagem, a aplicação (Figura 3.10) tem a capacidade de emitir alertas sonoros que chamam a atenção do condutor, caso este efectue travagens ou acelerações bruscas, dirija em excesso de velocidade, exista algum problema mecânico com o veículo ou em caso de acidente comunique às autoridades locais e familiares. Para além da emissão de alertas, o Automatic para cada viagem regista o consumo médio, o custo de combustível, a duração da viagem e o número de alertas de condução. Toda esta informação pode ser consultada a partir da aplicação móvel de forma bastante perceptível, pois encontra-se agrupada semanalmente e com uma classificação atribuída à condução do condutor, permitindo assim ter uma visão sobre a eficiência da sua condução actual, bem como uma perspectiva da sua evolução.

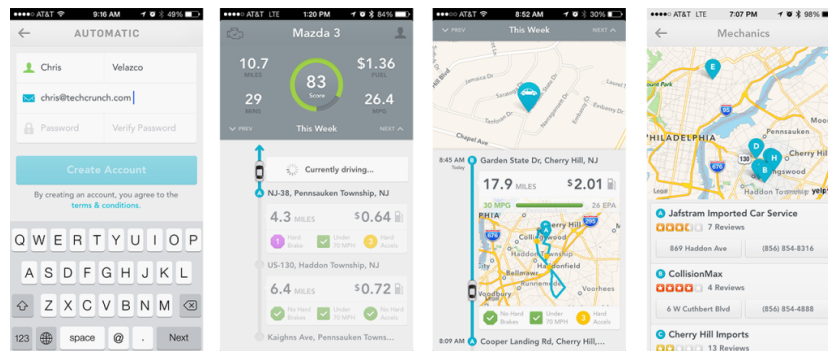


Figura 3.10: Solução *mobile* Automatic

MyCarTracks

O MyCarTracks [44] é uma aplicação desenvolvida para sistemas operativos móveis Android, pela Slash Idea. Esta aplicação utiliza os sensores existentes nos *smartphones* para realizar a monitorização dos veículos, fazendo uma análise sensorial que permite localizar a posição dos veículos, velocidade, quilómetros percorridos e duração total das viagens.

A informação relativa a cada veículo pode ser consultada no dispositivo móvel e também num *frontend* web (Figura 3.11), que dispõe de informação mais detalhada do veículo ou no caso de uma frota, do conjunto dos veículos constituintes da frota. Na aplicação web pode-se localizar a posição dos veículos em tempo-real, saber a distância percorrida, o tempo que demorou a realizar cada viagem e obter médias relativas à velocidade e tempos de paragem. Toda esta informação pode ser visualizada graficamente ou em forma de relatórios que podem ser exportados.

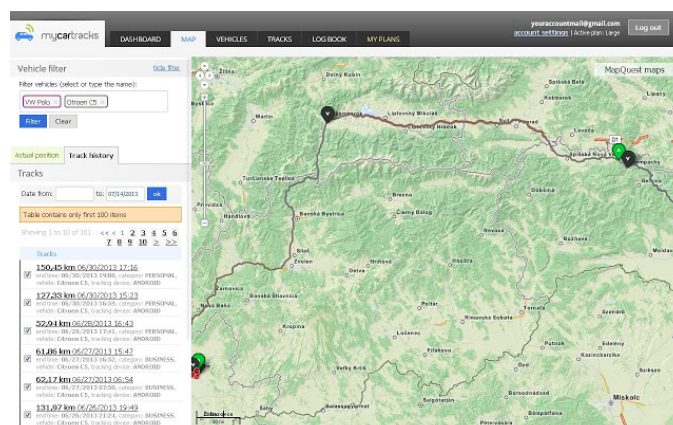


Figura 3.11: Solução MyCarTracks

Análise de funcionalidades e custos

Do conjunto de produtos identificados, foi realizada uma análise aprofundada das características, funcionalidades e custos oferecidos por cada um destes. Pretende-se com esta análise, descobrir quais os factores críticos de sucesso em que o produto da Sentilant deverá ter um desempenho superior ao dos seus possíveis concorrentes, ou seja, tentar identificar concretamente quais são as variáveis que condutores e gestores de frotas dão mais importância, para que seja possível criar mais valor para os clientes através dos produtos a desenvolver.

Para a realização do estudo comparativo de cada um dos produtos, foram definidos os seguintes critérios:

- **Características do sistema:** pretende-se identificar quais os requisitos necessários para adquirir o produto, se este possui um ambiente web ou *mobile* para interacção com o utilizador e se é disponibilizado como um serviço na *cloud*.
- **Plano de custos:** analisar os custos inerentes à aquisição do produto, bem como os custos mensais e anuais associados à utilização do produto.
- **Funcionalidades:** realizar o levantamento das funcionalidades oferecidas por cada solução, tais como monitorização em tempo-real, alertas proactivos, interacção social, análise da informação histórica e configurações de utilização e partilha da informação.

Esta análise encontra-se formalizada num documento interno que pode ser consultado em anexo (Anexo C), onde se pode observar a comparação entre cada solução em termos de custos e de funcionalidades.

Conclusões

Após a realização do estudo de soluções existentes no mercado, concluí-se que grande parte destas tem como principal objectivo simplesmente fornecer os valores monitorizados durante a condução. Estas não fornecem a informação devidamente tratada, através de algoritmos inteligentes de tratamento de dados que possam ajudar os condutores a tornarem a sua condução mais eficiente e segura. Visto que este é um mercado emergente, onde a necessidade deste género de soluções começa a aumentar de dia para dia, justifica-se assim plenamente o desenvolvimento dos produtos da Sentilant.

Estes conseguirão dar uma resposta às necessidades actuais do mercado e dos seus consumidores, através da produção de uma solução com custos de

aquisição mais baixos e com funcionalidades capazes de tornarem a condução de cada condutor mais eficiente.

Com a introdução de soluções de *business intelligence* que permitam detectar padrões comportamentais nas viagens realizadas pelos condutores, torna-se possível a compreensão dos factores que os condutores devem melhorar para conseguirem melhorar a sua condução.

Capítulo 4

Análise de requisitos

Este capítulo descreve a abordagem considerada para realizar o levantamento de requisitos funcionais e não funcionais para desenvolver a solução nas vertentes de B2C e B2B. Aqui será apresentada uma visão sobre o problema, os actores intervenientes no sistema, os protótipos de baixa e alta fidelidade desenvolvidos e a especificação dos requisitos funcionais e não funcionais.

Para elaborar todo o conjunto de requisitos, foi feito inicialmente o levantamento de um conjunto de necessidades de negócio, com a finalidade de recolher vários indicadores-chave relevantes para a área dos transportes. Em primeiro lugar através de uma análise de soluções comerciais (Secção 3.2), como forma de conhecer as funcionalidades que essas soluções oferecem, seguindo-se de uma reunião com o director geral da empresa ELACONTA¹, uma empresa que possui bastante conhecimento na área dos transportes rodoviários de mercadorias, com o intuito de perceber quais as necessidades reais que os condutores e gestores de frotas pretendem ver respondidas.

Após este processo, as necessidades de negócio obtidas foram discutidas e validadas junto de toda a equipa de desenvolvimento e gestão do projecto, uma vez que não existe concretamente um cliente final, dando assim origem ao conjunto de requisitos funcionais e não funcionais a serem desenvolvidos na solução para o problema.

Os indicadores-chave obtidos encontram-se formalizados num documento interno, em anexo (Anexo D), que especifica como estes podem auxiliar os condutores e gestores de frotas de empresas de transporte, distribuição e vendas, no processo de tomada de decisão, em determinados cenários possíveis.

¹ELACONTA: <http://www.elaconta.pt/>

4.1 Actores do sistema

Nesta secção serão identificados, descritos e qual o papel dos actores que irão interagir directamente com a solução a desenvolver, nas suas duas vertentes (B2C e B2B).

Tabela 4.1: Actores presentes na solução B2C

Actor	Representa	Papel
Condutor	Pessoa que utiliza o <i>smartphone</i> .	Registar grande parte das viagens realizadas, com a finalidade de recolher os indicadores de condução.
Utilizador web	Pessoa que utiliza o <i>frontend</i> web.	Utilizar as soluções incorporadas no <i>frontend</i> web, por forma a visualizar a evolução da eficiência da sua condução e perceber como a pode melhorar.

Tabela 4.2: Actores presentes na solução B2B

Actor	Representa	Papel
Condutor	Pessoa que utiliza o <i>smartphone</i> .	Registar todas as suas viagens e tarefas realizadas, com a finalidade de recolher os indicadores de condução e produtividade.
Gestor operacional	Pessoa que utiliza o <i>frontend</i> web.	Utilizar as soluções de <i>reporting</i> incorporadas no <i>frontend</i> web, por forma a gerar <i>dashboards</i> e relatórios com a informação de toda a organização, condutores, veículos e grupos.
<i>Business Analyst</i>	Pessoa que utiliza o <i>frontend</i> web.	Utilizar os <i>dashboards</i> e relatórios previamente gerados, para que consiga ter uma visão operacional e estratégica sobre a sua organização, por forma a tomar as melhores decisões.

4.2 Prototipagem

A utilização de prototipagem é uma técnica bastante útil no processo de engenharia de requisitos, pois permite aos *stakeholders*² identificarem concretamente aquilo que desejam de uma forma visual.

O desenvolvimento deste trabalho permitiu numa primeira iteração, através de prototipagem de baixa fidelidade, elaborar uma primeira versão dos sistemas baseados em requisitos ainda pouco definidos, servindo assim como forma de especificar concretamente qual o conjunto de requisitos desejados. Já numa segunda iteração, tornou-se possível criar protótipos que espelham o conjunto de requisitos definidos na primeira iteração, permitindo assim efectuar a validação e negociação, com os *stakeholders*, do conjunto de requisitos fundamentais para o sistema.

Numa fase inicial do desenvolvimento dos protótipos, foram primeiramente tomadas decisões ligadas ao *design* e usabilidade da aplicação com toda a equipa de desenvolvimento, uma vez que a aplicação deveria possuir um aspecto coerente em todas as suas versões.

Os protótipos elaborados e de seguida descritos visam abordar as duas vertentes do produto a desenvolver.

4.2.1 Prototipagem de baixa fidelidade

Numa primeira fase foi realizada uma análise sobre os dados disponíveis na fonte de dados existente, com a finalidade de desenvolver um conhecimento aprofundado sobre a estrutura, qualidade e granularidade dos dados. Visto que o sucesso de um projecto de *business intelligence* depende bastante da qualidade dos dados disponíveis, esta tarefa revela-se fundamental durante esta fase do projecto, uma vez que permite avaliar se é possível elaborar um conjunto de requisitos que respondam às necessidades dos actores do sistema. Através desta tarefa foi possível conhecer as características dos dados fornecidos e determinar um conjunto de agregados possíveis capazes de suportar os objectivos da solução a desenvolver.

Após a análise e compreensão dos dados partiu-se, numa segunda fase, para a elaboração dos primeiros protótipos dos sistemas a desenvolver, tendo como referência, as necessidades de negócio obtidas durante a fase de análise de soluções comerciais e em sessões de *brainstorming*³ individuais e com a

²*Stakeholders*: qualquer pessoa ou organização que tenha interesse, ou seja afectada pelo projecto (*e.g.*, cliente, gestor de projecto, analista, programador).

³*Brainstorming*: técnica de dinâmica de grupo ou individual utilizada para explorar a potencialidade criativa de um indivíduo ou de um grupo, com a finalidade de obter ideias e definir objectivos.

equipa de desenvolvimento e gestão do projecto.

Nesta fase foram tomadas decisões tanto ao nível de funcionalidades, que as soluções nas vertentes B2C e B2B deveriam garantir, como ao nível do *design* da solução. De seguida serão demonstrados apenas alguns exemplos dos muitos protótipos desenhados.

Business-to-consumer

Na figura 4.1 é apresentada a primeira versão da funcionalidade de comparação de trajectos, que deverá estar integrado na versão B2C. Este pretende demonstrar, para um conjunto de trajectos semelhantes, qual a evolução dos seus índices de condução e permitir a comparação entre trajectos.



Figura 4.1: Elaboração dos protótipos de baixa fidelidade do módulo de comparação de rotas

Business-to-business

Para a vertente B2B do projecto EcoMobile foram inicialmente realizados protótipos de baixa fidelidade que espelham as funcionalidades de criação de análises, *dashboards* e relatórios personalizados (Figura 4.2).

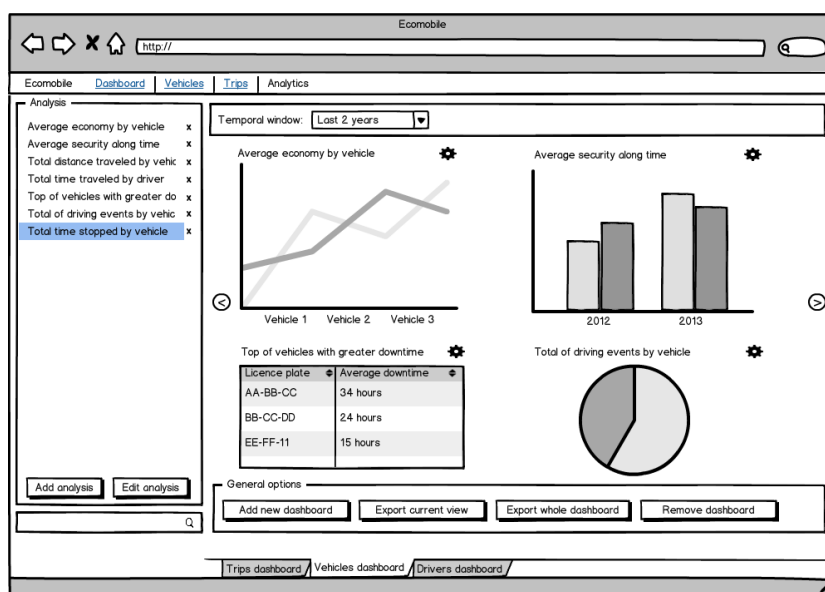


Figura 4.2: Elaboração dos protótipos de baixa fidelidade do módulo de *reporting*

4.2.2 Prototipagem de alta fidelidade

Após a elaboração dos protótipos de baixa fidelidade e depois de estes serem apresentados, discutidos e validados com a equipa de gestão do projecto, foi possível elaborar um conjunto de requisitos bem definidos para a solução a desenvolver. A partir dos requisitos definidos nesta fase, foi possível elaborar uma versão do sistema mais próxima da final, através de protótipos de alta fidelidade. Nesta secção serão demonstrados apenas alguns exemplos dos muitos protótipos desenhados.

Business-to-consumer

Os protótipos apresentados nas figuras 4.3 e 4.4 dizem respeito à comparação entre trajectos semelhantes e recomendação dos melhores trajectos, em termos de segurança, economia, distância e tempo de viagem, para os actores do sistema. Estes foram apresentados e validados junto da equipa de gestão do projecto e serão especificados nas secções seguintes como requisitos funcionais e não funcionais para a solução na vertente B2C.

Na figura 4.3 é possível observar as funcionalidades e *design* da componente de comparação de trajectos semelhantes. Neste encontra-se presente a evolução da segurança e economia do actor Condutor, ao longo de um determinado trajecto, bem como uma lista com os trajectos semelhantes a uma

determinada viagem, onde é possível pesquisar, filtrar pelas próprias viagens e viagens dos amigos, ordenar e seleccionar uma viagem para comparar.

A comparação tem como principal objectivo permitir ao Condutor perceber quais são as zonas com pior eficiência na sua condução. Ao seleccionar uma viagem para comparar é desenhado um gráfico que compara os indicadores de economia, segurança e velocidade entre os dois trajectos. É também representado no mapa os dois trajectos que estão a ser comparados, as zonas em comum e os eventos de condução ao longo dos trajectos.

Já na figura 4.4 encontram-se as funcionalidades e o *design* da componente de recomendação dos melhores trajectos. Aqui é disponibilizado ao actor Condutor os melhores trajectos em termos de economia, segurança, distância e tempo de viagem, independentemente de quem o tenha realizado, uma vez que aqui o importante é alertar um condutor para a existência de melhores alternativas para a realização de um determinado trajecto.

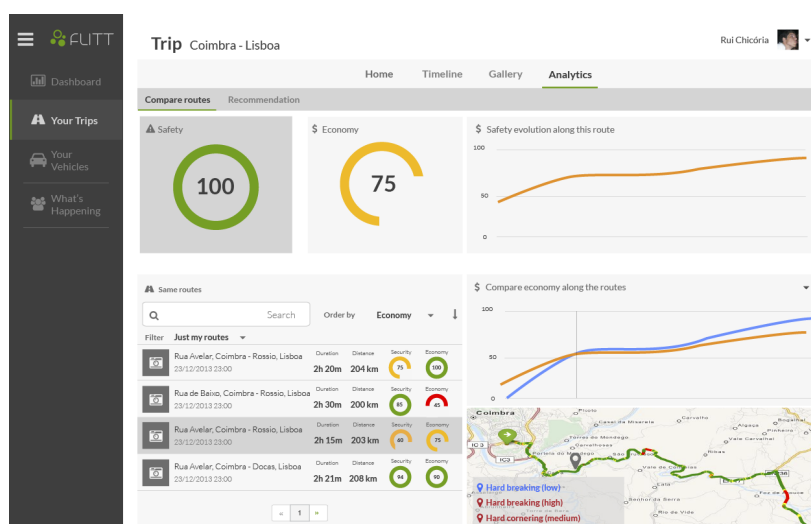


Figura 4.3: Protótipo de alta fidelidade do módulo de comparação de trajectos

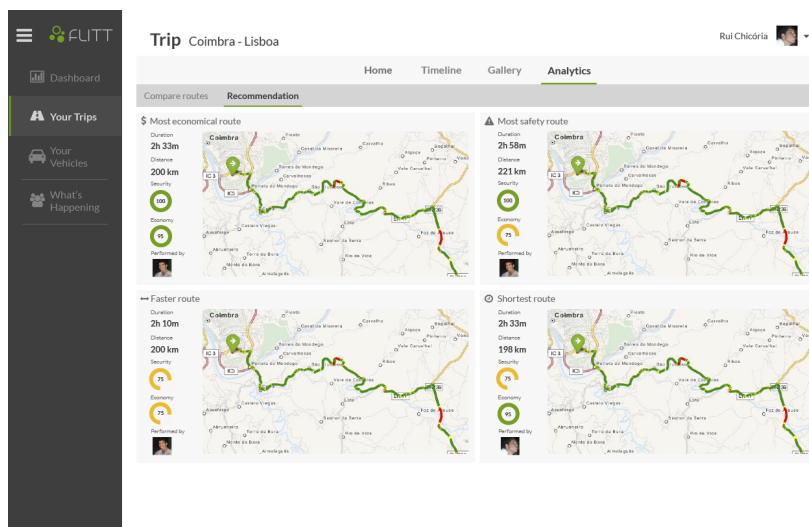


Figura 4.4: Protótipo de alta fidelidade do módulo de recomendação de trajectos

Business-to-business

Os protótipos apresentados nas figuras 4.5 e 4.6 demonstram as funcionalidades e *design* do componente de *reporting*. Este tem como principal objectivo permitir ao actor Gestor de Frota e *Business Analyst* criar relatórios e *dashboards* personalizados, bem como análises personalizadas tendo como base um conjunto de indicadores disponibilizados.



Figura 4.5: Protótipo de alta fidelidade do módulo de *reporting*

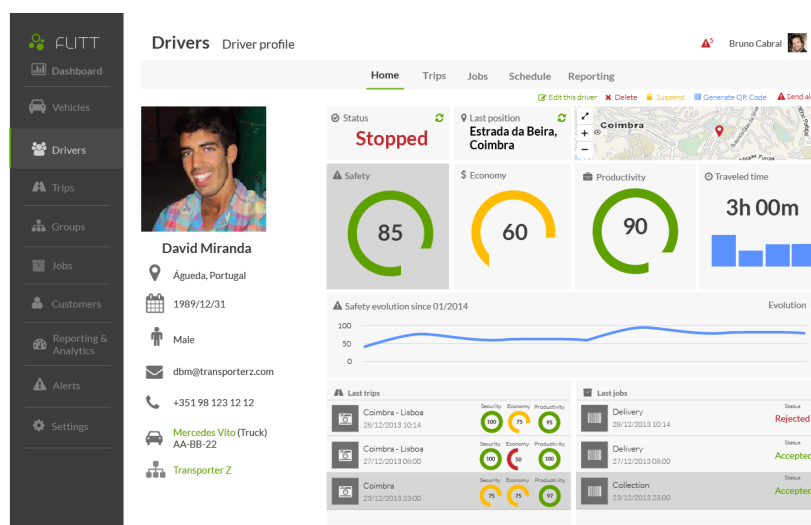


Figura 4.6: Protótipo de alta fidelidade do módulo de *reporting* no perfil do condutor

4.3 Requisitos funcionais

A especificação de requisitos funcionais a desenvolver foi realizada através da utilização de *user stories*, tipicamente utilizadas em cenários de desenvolvimento ágil. Uma *user story* consiste numa descrição curta e simples de uma interação que um actor pode realizar com o sistema [45]. A grande vantagem da utilização de *user stories* para especificação de requisitos funcionais prende-se com o facto de estas nos fazerem pensar naquilo que se deve elaborar do ponto de vista de um utilizador final, para além de que são bastante rápidas de construir, de fácil compreensão para qualquer cliente não técnico e facilmente mapeadas em testes de aceitação para validação da implementação de cada funcionalidade.

Para a definição das *user stories* um dos formatos mais utilizados baseia-se em três expressões, utilizadas para responderem a três questões relacionadas com uma funcionalidade [46]:

- **Enquanto** <actor do sistema> (Quem?)
- **Quero** <acção> (O quê?)
- **De forma a** <benefício> (Porquê?)

A definição da prioridade dos requisitos é um aspecto bastante importante a definir, tendo para isso sido utilizadas as palavras-chave, definidas no

RFC 2119, "Must", "Should", "Could" e "Optional" como forma de criar uma priorização sobre as funcionalidades dos sistemas a desenvolver.

De seguida será apresentado um resumo sucinto de cada um dos requisitos. A definição completa de cada requisito pode ser consultada no Anexo E.

4.3.1 *Business-to-consumer*

Tabela 4.3: Requisitos funcionais da vertente B2C

ID	Resumo	Prioridade
B2C-01	Visualizar a média da segurança para uma determinada rota	<i>MUST</i>
B2C-02	Visualizar a média da economia para uma determinada rota	<i>MUST</i>
B2C-03	Visualizar graficamente a evolução da segurança ao longo do tempo para uma determinada rota	<i>MUST</i>
B2C-04	Visualizar graficamente a evolução da economia ao longo do tempo para uma determinada rota	<i>MUST</i>
B2C-05	Seleccionar um trajecto semelhante, realizado por mim, para comparação	<i>MUST</i>
B2C-06	Seleccionar um trajecto semelhante, realizado por um amigo, para comparação	<i>MUST</i>
B2C-07	Seleccionar um trajecto com o mesmo início e fim, realizado por mim, para comparação	<i>MUST</i>
B2C-08	Seleccionar um trajecto com o mesmo início e fim, realizado por amigo, para comparação	<i>MUST</i>
B2C-09	Seleccionar o indicador-chave pretendido para comparar globalmente entre dois trajectos	<i>MUST</i>

ID	Resumo	Prioridade
B2C-10	Seleccionar o indicador-chave pretendido para comparar ao longo de dois trajectos	<i>MUST</i>
B2C-11	Visualizar no mapa os dois trajectos que estão a ser comparados	<i>MUST</i>
B2C-12	Visualizar no mapa os eventos de condução de cada trajecto	<i>MUST</i>
B2C-13	Recomendar o trajecto mais económico para uma determinada rota	<i>SHOULD</i>
B2C-14	Recomendar o trajecto mais seguro para uma determinada rota	<i>SHOULD</i>
B2C-15	Recomendar o trajecto mais rápido para uma determinada rota	<i>SHOULD</i>
B2C-16	Recomendar o trajecto mais curto para uma determinada rota	<i>SHOULD</i>

4.3.2 *Business-to-business*

Tabela 4.4: Requisitos funcionais da vertente B2B

<i>User story</i>	Resumo	Prioridade
B2B-01	Criar análises personalizadas, tendo como base um conjunto de indicadores-chave pré-definidos	<i>COULD</i>
B2B-02	Pesquisar as análises criadas e pré-definidas	<i>MUST</i>
B2B-03	Criar uma vista com várias análises, através do <i>drag-and-drop</i> das análises criadas ou pré-definidas	<i>MUST</i>

<i>User story</i>	Resumo	Prioridade
B2B-04	Aplicar um filtro temporal global sobre a vista actual que contém as análises seleccionadas	<i>MUST</i>
B2B-05	Fazer o <i>drill-down</i> e <i>drill-up</i> em cada uma das análises seleccionadas	<i>MUST</i>
B2B-06	Ao realizar o <i>drill-down</i> em cada uma das análises seleccionadas, escolher por qual tipo de granularidade se deseja agrupar	<i>MUST</i>
B2B-07	Remover uma análise, previamente arrastada da vista actual	<i>MUST</i>
B2B-08	Alterar o tipo de vista, sobre uma das análises, entre gráfico e tabela	<i>COULD</i>
B2B-09	Exportar uma análise específica para PDF ou CSV	<i>COULD</i>
B2B-10	Filtrar a agregação dos dados da actual vista por determinados condutores	<i>SHOULD</i>
B2B-11	Filtrar a agregação dos dados da actual vista por determinados veículos	<i>SHOULD</i>
B2B-12	Filtrar a agregação dos dados da actual vista por determinados grupos da organização	<i>SHOULD</i>
B2B-13	Exportar a actual vista para PDF	<i>COULD</i>
B2B-14	Guardar a actual vista como um <i>dashboard</i>	<i>SHOULD</i>
B2B-15	Guardar a actual vista como um relatório	<i>SHOULD</i>
B2B-16	Limpar todas as análises da vista actual	<i>MUST</i>
B2B-17	Aceder à lista de <i>dashboards</i> previamente criados	<i>SHOULD</i>

<i>User story</i>	Resumo	Prioridade
B2B-18	Aceder à lista de relatórios previamente criados	<i>SHOULD</i>

4.4 Requisitos não funcionais

O objectivo da elaboração de requisitos não funcionais é descrever as qualidades necessárias que o sistema deve compreender, bem como as suas características de usabilidade e desempenho. Uma vez que as *user stories* não se adequam à elaboração de requisitos desta natureza, não existiu uma restrição específica para efectuar a sua descrição, tendo estes sido descritos de forma livre.

Os atributos de qualidade definidos aplicam-se tanto à aplicação na vertente B2C como na vertente B2B.

Escalabilidade

O sistema deve possuir a capacidade de aumentar o seu desempenho, sob um aumento da carga de utilização ao nível aplicacional, sem exigir mudanças da arquitectura ou perda do serviço devido ao tempo necessário para redimensionar os recursos. Assim a plataforma deverá estar projectada para ser totalmente elástica e, no contexto destas soluções, suportar um mínimo de 30 mil utilizadores provenientes do mercado norte-americano e europeu.

Robustez

O sistema deve garantir toda a integridade da informação, quer seja em situações de *stress* ou situações de falha aplicacional. Não é admitido qualquer perda de informações relacionadas com as viagens que foram realizadas pelos utilizadores. Assim o sistema deve possuir a capacidade de recuperar a informação que não foi devidamente processada no limite máximo de 1 dia.

Segurança

Os acessos não autorizados à aplicação e aos dados não devem ser permitidos pelo sistema. Este deverá também garantir a anonimização de toda a informação que é disponibilizada aos utilizadores para efeitos de comparação e recomendação. Todas as comunicações externas entre a aplicação e os clientes devem ser encriptadas.

Custos

Este requisito está directamente relacionado com o plano de negócios da empresa. A infra-estrutura deverá possuir a capacidade de se adaptar às crescentes ou decrescentes necessidades do negócio (*pay-as-you-go*).

Capítulo 5

Análise e desenho da arquitectura

Neste capítulo será descrita a arquitectura do projecto EcoMobile, da vertente B2C e B2B, e quais as contribuições que foram realizadas na arquitectura do projecto no âmbito do estágio para a resolução do problema. Serão também demonstradas as decisões arquitecturais tomadas, que visam suportar os requisitos funcionais e não funcionais da solução desenvolvida.

Uma vez que a arquitectura do projecto EcoMobile possui já alguma complexidade e é modularizada, por forma a permitir o desenvolvimento paralelo de funcionalidades com outros programadores, optou-se por abordar apenas os aspectos arquitecturais relacionados com a resolução do problema proposto neste estágio.

Para a descrição detalhada da arquitectura, e para uma melhor percepção desta, serão utilizadas três perspectivas diferentes.

- **Perspectiva estática:** é uma perspectiva que pretende demonstrar como se encontram estruturadas as responsabilidades do sistema.
- **Perspectiva dinâmica:** esta perspectiva pretende demonstrar os diversos fluxos existentes no sistema.
- **Perspectiva física e tecnológica:** pretende demonstrar as perspectivas anteriores em termos de distribuição das responsabilidades por diversas máquinas físicas, as tecnologias utilizadas e os protocolos de comunicação existentes.

5.1 Visão global da arquitectura

Nesta secção pretendem-se dar ao leitor a percepção de quais os componentes arquitecturais trabalhados no âmbito deste estágio, através da demonstração da arquitectura global do projecto EcoMobile. Como referido anteriormente (Secção 1.1) este projecto é composto por três grandes áreas (desenvolvimento *mobile*, desenvolvimento web e BI), onde cada uma delas tem a participação de diferentes programadores.

Visão global da arquitectura B2C

Na figura 5.1 é possível observar a arquitectura global, da vertente B2C, do projecto EcoMobile. Os módulos de cor verde, presentes nos componentes do EcoMobile, foram os módulos nos quais o estagiário trabalhou para a resolução do problema apresentado neste estágio.

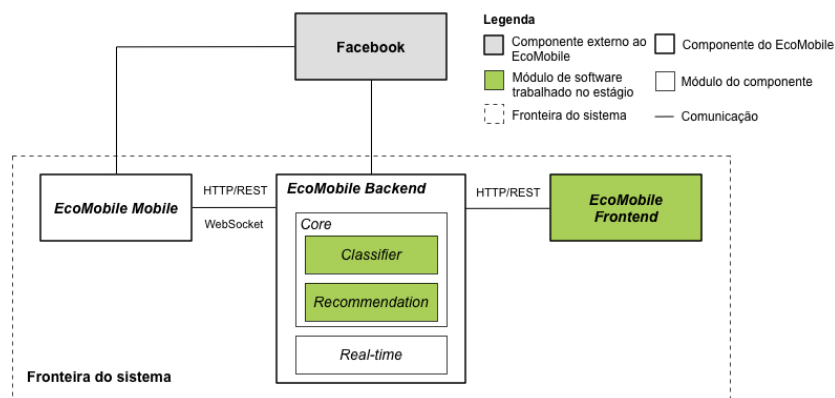


Figura 5.1: Visão global da arquitectura da solução B2C

EcoMobile Mobile Este componente é da responsabilidade do programador da área *mobile* e tem como principal função a recolha e tratamento inteligente da informação sensorial proveniente do *smartphone*, para posteriormente ser comunicada ao serviço *EcoMobile Backend*.

EcoMobile Backend Este componente é, actualmente, da responsabilidade de dois programadores e tem como principal função a parte operacional e de BI de todo o sistema. Os módulos operacionais do sistema (*Core* e *Real-time*) encontram-se ao cargo de um dos programadores de *backend/frontend* do projecto e é independente dos módulos implementados no estágio. Este implementa as funcionalidades de gestão de utilizadores, veículos, viagens e rede social.

Os módulos de BI (*Classifier* e *Recommendation*), presentes no *Core*, são os módulos da responsabilidade do estagiário e implementam as funcionalidades que permitem a classificação, comparação e recomendação de trajectos. Estes serão detalhados posteriormente.

EcoMobile Frontend Este componente é, actualmente, da responsabilidade de dois programadores e tem como principal função implementar a lógica aplicacional do lado do cliente (*client-side*). Neste componente, encontram-se ao cargo do estagiário a implementação do interface e lógica aplicacional que permite ao Utilizador web visualizar a evolução da sua condução ao longo de trajectos semelhantes e realizar a comparação dos indicadores-chave de condução entre trajectos, bem como obter recomendações sobre os melhores trajectos para uma determinada rota (Tabela 4.3).

Facebook É um componente que se encontra no exterior da fronteira do sistema EcoMobile. O EcoMobile utiliza o Facebook para realizar a integração e partilha de conteúdos na rede social. Neste estágio não existe qualquer interacção com este componente.

Visão global da arquitectura B2B

Na figura 5.2 é possível observar a arquitectura global, da vertente B2B, do projecto EcoMobile. Esta arquitectura segue os princípios arquitecturais adoptados pela arquitectura B2C, apenas os módulos presentes nos seus componentes possuem lógicas aplicacionais diferentes.

Os módulos de cor verde, presentes nos componentes do EcoMobile, foram os módulos nos quais o estagiário trabalhou para a resolução do problema apresentado neste estágio.

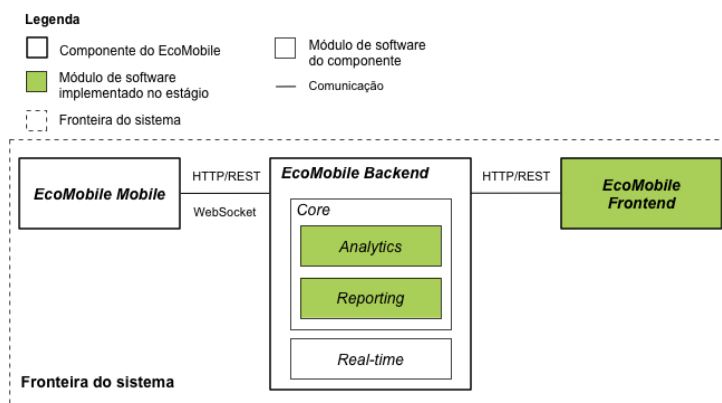


Figura 5.2: Visão global da arquitectura da solução B2B

EcoMobile Mobile Este componente, já descrito anteriormente, aquando a realização deste estágio o início do seu desenvolvimento sofreu uma alteração de planeamento. No entanto sabe-se, que para além de outras funcionalidades relacionadas com o negócio necessárias para esta vertente, este irá estender o módulo de recolha e tratamento inteligente da informação sensorial proveniente do *smartphone* já existente na vertente B2C.

EcoMobile Backend Este componente tem como principal função a parte operacional e de BI de todo o sistema. Aquando a realização deste estágio o início do seu desenvolvimento sofreu uma alteração de planeamento, tendo o estagiário assumido a implementação da infra-estrutura base do sistema (*Core*) para a realização dos objectivos a cumprir. Os módulos operacionais do sistema (*Core* e *Real-time*), irão estender algumas funcionalidades da versão B2C, no entanto grande parte da lógica aplicacional foi alterada.

Foram adicionados ao *Core* da vertente B2B, dois novos módulos de software (***Analytics***) e (***Reporting***). Estes módulos são da responsabilidade do estagiário e permitem o processamento de agregados sobre a informação sensorial recebida sob diferentes níveis de granularidade e visualização multidimensional sobre essa informação. Este será detalhado posteriormente.

EcoMobile Frontend Este componente tem como principal função implementar a lógica aplicacional do lado do cliente (*client-side*). Neste componente, encontram-se ao cargo do estagiário a implementação do interface e lógica aplicacional que permite ao Gestor de frota criar análises personalizadas, tendo por base um conjunto de indicadores-chave, com a finalidade de criar os *dashboards* e relatórios necessários para gestão da informação operacional e estratégica da organização (Tabela 4.4).

5.2 Decisões arquitecturais

Nesta secção serão demonstradas algumas das decisões que influenciaram o desenho da arquitectura para resolver o problema proposto neste estágio. As decisões tomadas tiveram como base o estudo de metodologias e tecnologias realizado na secção 3.1, bem como alguns princípios arquitecturais que já se encontravam presentes no projecto EcoMobile no início deste estágio.

Desacoplamento

O princípio do desacoplamento permite reduzir o risco de que alterações realizadas num determinado módulo, não afectem o funcionamento de outros módulos do sistema. Este princípio promove também a escalabilidade do

sistema, uma vez que permite escalar partes da infra-estrutura que sejam o *bottleneck* no sistema. Visto que este projecto possuía algumas funcionalidades já incorporadas e também possui um plano de desenvolvimento a longo prazo, decidiu-se seguir esta abordagem durante o desenvolvimento das funcionalidades pretendidas neste estágio (e.g., criação do módulo de comparação de trajectos).

REST

Representational state transfer (REST) é um mecanismo que permite criar, ler, actualizar ou excluir informações de um servidor através de simples chamadas HTTP. Ao contrário de mecanismos mais complexos como o SOAP, o mecanismo REST permite que a formatação dos dados fique a cargo do sistema e estes sejam disponibilizados no formato XML, JSON ou HTML, facilitando de certa forma a integração e comunicação da informação.

No projecto EcoMobile já se encontravam implementados serviços REST, assim para o desenvolvimento do trabalho realizado neste estágio, não se encontrou nenhuma desvantagem na utilização de serviços REST para a disponibilização da informação.

Sistema de mensagens

A utilização de um sistema de mensagens assíncrono é fundamentalmente uma reacção pragmática para problemas relacionados com sistemas distribuídos. As infra-estruturas que utilizam comunicação assíncrona têm a tendência a escalar mais facilmente e a possuir uma maior disponibilidade, uma vez que os serviços continuam a possuir a capacidade de realizar as suas funções e a responder aos clientes. Uma vez que a aplicação irá lidar com uma grande quantidade de dados e tendo em conta algumas tarefas computacionalmente pesadas, torna-se necessário a utilização de comunicações assíncronas para realizar o tratamento e armazenamento da informação.

Caching

A utilização de mecanismos de cache permite armazenar informação de forma transparente, de modo a que futuros pedidos sejam servidos de forma mais rápida. Os dados que são armazenados em cache são dados que já foram previamente processados, e no caso de serem solicitados (*cache hit*) estes podem ser servidos através de uma simples leitura da cache, que é bastante mais rápido e retira carga computacional desnecessária.

No contexto do trabalho realizado, torna-se importante a utilização de mecanismos de cache (*in-memory*) tanto a nível de armazenamento de dados

processados ou agregados, como também para a sincronização distribuída de processamento paralelo da informação.

5.3 Arquitectura desenvolvida

Nesta secção pretende-se abordar a solução arquitectural desenvolvida para a implementação das funcionalidades propostas durante a realização do estágio. Será apresentada a arquitectura, da vertente B2C e B2B, de um ponto de vista estático, dinâmico e físico.

5.3.1 Business-to-consumer

Pretende-se nesta subsecção demonstrar a solução arquitectural para a implementação das funcionalidades que permitem realizar os requisitos relacionados com a comparação e recomendação de trajectos (Tabela 4.3).

Perspectiva estática

Na figura 5.3 é demonstrada a perspectiva estática da arquitectura da vertente B2C.

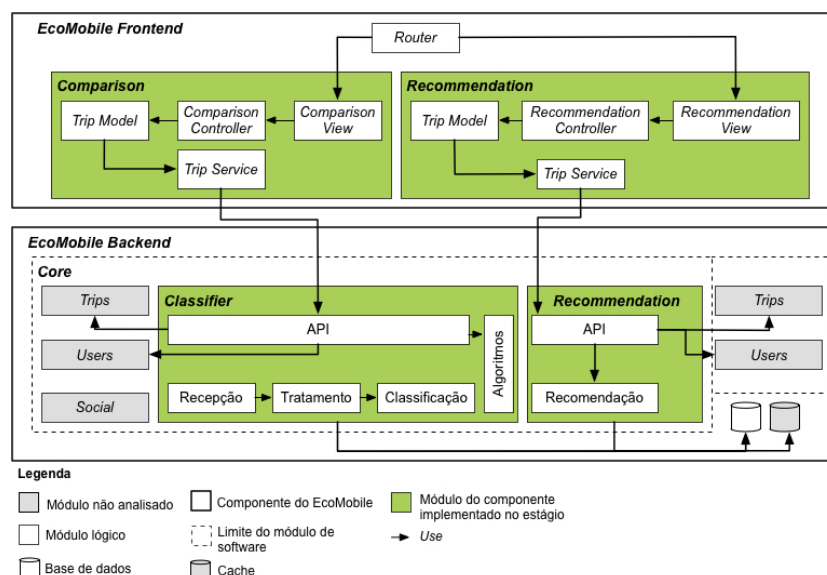


Figura 5.3: Perspectiva estática da arquitectura B2C

Estrutura do *EcoMobile Frontend*

Para o desenvolvimento do ambiente de *frontend* optou-se pela utilização da *design pattern Model-View-Controller* (MVC), por forma a garantir o desacoplamento, facilidade de manutenção e extensibilidade do código.

Router Garante a gestão do *browser* em termos de navegação, permitindo organizar as partes existentes na aplicação numa máquina de estados, ao invés da utilização de rotas pelo *url*.

Trip Service Possui os métodos que permitem a chamada aos serviços REST, existentes no *backend*, para obter a informação necessária para o lado do cliente relacionada com a comparação e recomendação de trajectos.

Trip Model Representa o modelo de dados da informação para a comparação e recomendação de trajectos, no lado do cliente, que é obtida através dos serviços REST provenientes do *backend*.

Comparison Controller Realiza toda a gestão da interacção do utilizador com o interface de comparação de trajectos e o acesso aos dados a serem disponibilizados para esse interface.

Comparison View Possui a vista que implementa o interface disponibilizado ao utilizador para realizar a comparação de trajectos.

Recommendation Controller Realiza toda a gestão da interacção do utilizador com o interface de recomendação de trajectos e o acesso aos dados a serem disponibilizados para esse interface.

Recommendation View Possui a vista que implementa o interface disponibilizado ao utilizador para a recomendação de trajectos.

Estrutura do *EcoMobile Backend*

O *backend* implementado assenta também sobre uma *design pattern* (MVC), para que assim também sejam adoptados princípios de desacoplamento e modularidade, por forma a garantir uma boa extensibilidade e manutenção do código produzido.

Trips Implementa toda a gestão de viagens do sistema. Permite adicionar, editar, eliminar e aceder ao conjunto de viagens que o utilizador já tenha realizado. O desenvolvimento deste módulo não foi da responsabilidade do estagiário.

Users Implementa a gestão de utilizadores do sistema. Permite realizar o registo de utilizadores na plataforma, realizar a autenticação e fazer toda a gestão da informação pessoal. O desenvolvimento deste módulo não foi da responsabilidade do estagiário.

Social Implementa a rede social para partilha de conteúdos relacionados com as viagens entre utilizadores do sistema. O desenvolvimento deste módulo não foi da responsabilidade do estagiário.

Classifier Módulo que implementa o algoritmo de classificação de trajectos semelhantes para disponibilizar as funcionalidades de comparação de trajectos.

- **API:** Implementa o serviço REST para disponibilização dos dados para a comparação de trajectos, bem como faz a gestão toda a lógica aplicacional necessária para a autorização de acesso à informação.
- **Recepção:** Realiza a recepção dos dados sensoriais provenientes do *smartphone*.
- **Tratamento:** É responsável pelo pré-processamento dos dados sensoriais provenientes do *smartphone*, por forma a limpar e a agregar para posterior utilização.
- **Classificação:** Tem a principal função de realizar a classificação e respectivo agrupamento de viagens que são consideradas semelhantes pelo classificador.
- **Algoritmos:** Implementa todo um conjunto de algoritmos que fornecem suporte às funcionalidades de classificação e comparação.

Recommendation Módulo que implementa o algoritmo de recomendação de trajectos, por forma a disponibilizar as funcionalidades de recomendação dos trajectos mais seguros, mais económicos, mais rápidos e mais curtos.

- **API:** Implementa o serviço REST para disponibilização dos dados para a recomendação de trajectos e faz a gestão toda a lógica aplicacional necessária para a autorização de acesso à informação.
- **Recomendação:** Implementa o algoritmo de recomendação dos melhores trajectos para uma determinada rota.

Perspectiva dinâmica

Aqui pretende-se demonstrar a arquitectura do seu ponto de vista dinâmico, por forma a dar a perceber os diversos fluxos existentes no sistema. Esta será descrita em três fases.

1. **Perspectiva dinâmica da classificação:** demonstra o fluxo de dados existentes para realizar a classificação de trajectos semelhantes.

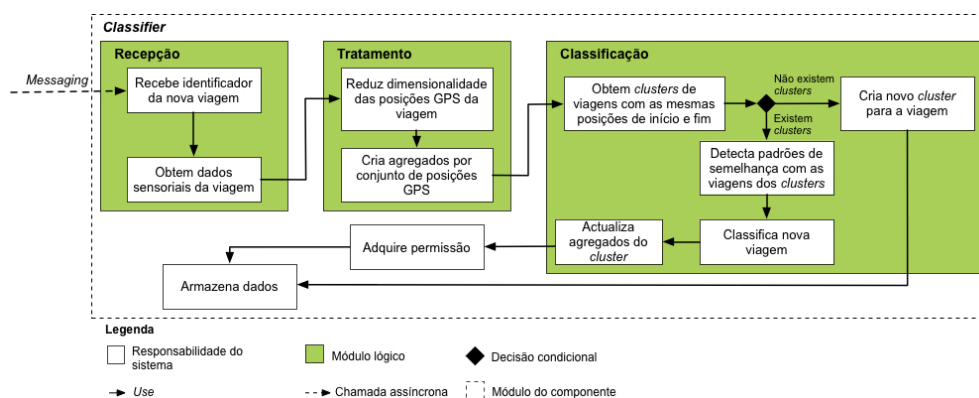


Figura 5.4: Perspectiva dinâmica da arquitectura - Classificação

Na figura 5.4 observar-se a realização da recepção de uma nova viagem para processar, através de uma chamada assíncrona. Após essa notificação é feita a leitura dos dados sensoriais, da correspondente viagem, para serem reencaminhados para o módulo lógico de tratamento.

Este tem como principal função reduzir a dimensionalidade dos dados sensoriais brutos, através da criação do agrupamento de coordenadas consideradas semelhantes. Para cada grupo de coordenadas são calculados os indicadores-chave de condução (e.g., segurança, economia).

Após realizado o pré-processamento dos dados sensoriais da viagem, estes são reencaminhados para o módulo lógico de classificação que irá classificar a viagem e colocá-la no *cluster* correspondente, caso exista. Após a classificação da viagem essa informação é armazenada e é feita a actualização dos indicadores de segurança, economia, distância e tempo de viagem correspondente ao *cluster*, para posteriormente utilizar para efeitos de recomendação. Na secção 6.1 será detalhada a implementação do algoritmo.

2. **Perspectiva dinâmica da comparação:** demonstra o fluxo de dados existentes para a disponibilização de dados que permitam a comparação entre trajectos.

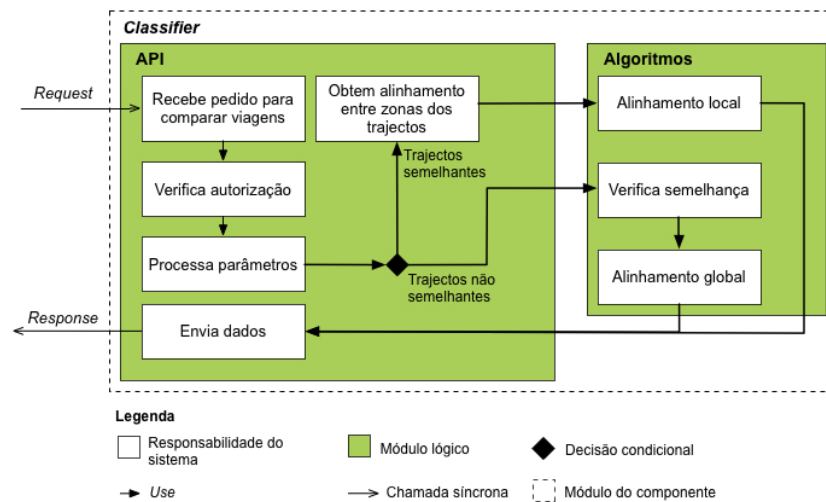


Figura 5.5: Perspectiva dinâmica da arquitectura - Comparação

É realizada uma chamada à API, do módulo de software *Classifier*, para a realização da comparação entre trajectos de duas viagens (Figura 5.5). Posteriormente são processados os parâmetros recebidos que podem ser, por exemplo, a indicar se é para comparar zonas semelhantes, ou o trajecto como um todo. No entanto este processo só é efectuado caso o cliente que realizou o pedido de comparação, possua autorização para aceder às viagens em comparação, sendo que os dados só serão enviados para o cliente caso a confirmação seja positiva.

Caso os trajectos se encontrem no mesmo *cluster*, significa que estes são semelhantes, e é então efectuado o alinhamento das zonas geográficas comuns para realizar a comparação. Caso contrário, é feita uma validação de semelhança por forma a verificar se existem zona comuns nos trajectos, e nesse caso é aplicado o algoritmo de alinhamento global.

3. **Perspectiva dinâmica da recomendação:** demonstra o fluxo de dados existente para a disponibilização dos melhores trajectos para uma determinada rota.

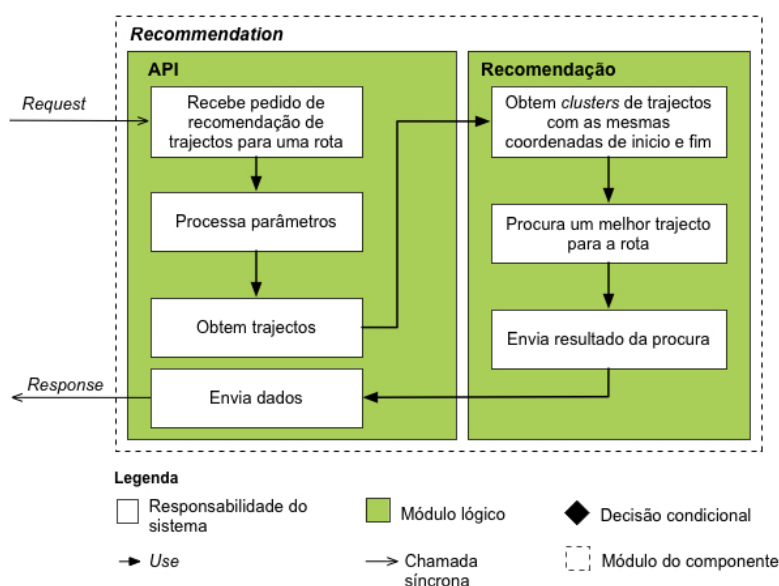


Figura 5.6: Perspectiva dinâmica da arquitectura - Recomendação

Na figura 5.6 observa-se que é realizada uma chamada à API, do módulo de software *Recommendation*, para efectuar um pedido para obter melhores trajectos para uma determinada rota. Após a recepção do pedido é efectuado o processamento dos parâmetros. Parâmetros esses que podem ser, por exemplo, a indicar se é pretendido o trajecto mais económico ou o trajecto mais curto.

No passo seguinte é invocado o algoritmo de recomendação, que procura os *clusters* de viagens que possuem as mesmas coordenadas de início e de fim, da rota pretendida, sendo que de seguida é realizada uma procura sobre o resultado por forma a perceber se existe um trajecto mais eficiente. No fim são enviados os dados para o cliente com a informação dos trajectos para a rota pretendida.

5.3.2 Business-to-business

Para o desenvolvimento da arquitectura da vertente B2B, devido ao desacoplamento e modularidade presente na arquitectura da vertente B2C, foi possível fazer a extensão dos componentes de software *Users*, *Trips*, *Classifier* e *Recommendation*. Sendo que foram apenas implementados os novos módulos de software que permitem realizar os requisitos pretendidos para esta versão.

Perspectiva estática

Na figura 5.7 é demonstrada a perspectiva estática da arquitectura da vertente B2B.

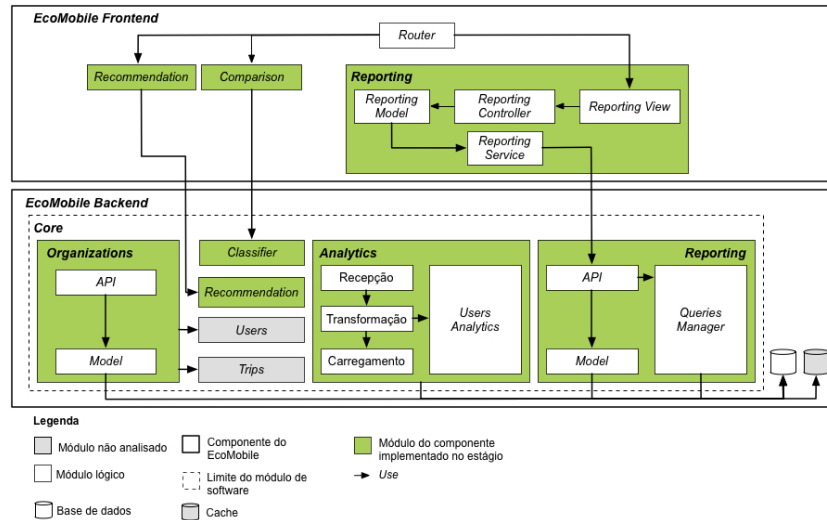


Figura 5.7: Perspectiva estática da arquitectura B2B

EcoMobile Frontend

A arquitectura do *frontend* da versão B2B segue os mesmos princípios e estrutura estática, descrita na subsecção 5.3.1, da arquitectura da versão B2C. Com o princípio da modularidade, foi possível estender as funcionalidades já existentes e acrescentar novas funcionalidades, sem que estas afectassem o funcionamento das restantes. De seguida será realizada a descrição dos novos módulos introduzidos.

Reporting Service Possui os métodos que permitem a chamada aos serviços REST, existentes no *backend*, para obter a informação necessária para o lado do cliente relacionada com a criação de análises, *dashboards* e relatórios personalizados.

Reporting Model Representa o modelo de dados da informação para a elaboração das análises, *dashboards* e relatórios, no lado do cliente, que é obtida através dos serviços REST provenientes do *backend*.

Reporting Controller Realiza toda a gestão da interacção do utilizador com o interface de criação de análises, *dashboards* e relatórios e o acesso aos dados a serem disponibilizados para esse interface.

Reporting View Possui a vista que implementa o interface disponibilizado ao utilizador para a criação e gestão de *dashboards* e relatórios.

EcoMobile Backend

A arquitectura do *backend* da versão B2B segue os mesmo princípios e estrutura estática, descrita na subsecção 5.3.1, da arquitectura da versão B2C. Com o princípio da modularidade, foi possível extender as funcionalidades já existentes e acrescentar novas funcionalidades, sem que estas afectassem o funcionamento das restantes. De seguida será realizada a descrição dos novos módulos introduzidos.

Organizations Implementa toda a gestão de uma organização. Este permite realizar toda a gestão de condutores, administradores e veículos. O desenvolvimento deste módulo foi da responsabilidade do estagiário.

Analytics Módulo responsável pela recepção, transformação e carregamento dos agregados das várias dimensões da organização. O desenvolvimento deste módulo foi da responsabilidade do estagiário.

- **Recepção:** Tem a função de receber os dados sensoriais provenientes do *smartphone*.
- **Transformação:** É responsável pela criação dos agregados e indicadores-chave para as diversas granularidades e dimensões da organização.
- **Carregamento:** É responsável pela gestão do armazenamento dos agregados calculados, para a estrutura de dados que os irá guardar para posterior disponibilização.
- **Users Analytics:** Possui as funcionalidades para a criação dos agregados multidimensionais para os utilizadores da organização.

Reporting Módulo responsável pela interpretação das *queries* sobre a informação, requisitadas pelo cliente, e respectiva disponibilização. O desenvolvimento deste módulo foi da responsabilidade do estagiário.

- **API:** Implementa o serviço REST para disponibilização dos dados para a criação de análises, *dashboards* e relatórios, bem como faz a gestão toda a lógica aplicacional necessária para a autorização de acesso à informação.
- **Queries Manager:** Possui toda a lógica necessária para interpretar as *queries* e criar os agregados segundo a sua interpretação.

- **Model:** É a camada que faz a gestão dos dados, correspondentes à organização, para realizar as análises, *dashboards* e relatórios personalizados.

Perspectiva dinâmica

De seguida será realizada uma descrição da perspectiva dinâmica da arquitectura em duas fases:

1. **Perspectiva dinâmica do processamento de agregados:** demonstra o fluxo de dados existente para a realização do processamento de agregados multidimensionais.

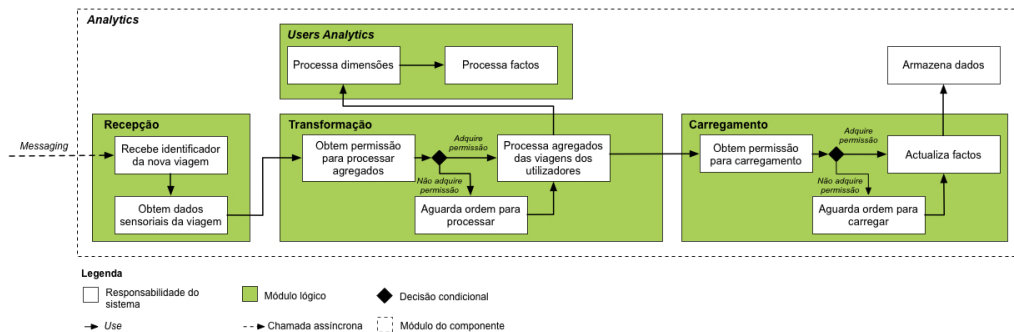


Figura 5.8: Perspectiva dinâmica da arquitectura - Agregados

O módulo recebe a notificação para processar uma nova viagem, através de uma mensagem assíncrona, que envia o identificador da nova viagem (Figura 5.8). De seguida são obtidos os dados sensoriais relativos à viagem para serem enviados para o módulo lógico de transformação.

Este tenta obter a possibilidade de processar os agregados correspondentes ao utilizador que realizou a viagem. Caso se verifique, é efectuado então o processamento das dimensões e factos relativos às viagens do utilizadores que são enviados para o módulo lógico de carregamento. Após a recepção dos agregados processados, este tenta obter a possibilidade de inserir ou actualizar os indicadores-chave relativos ao utilizador, que são guardados à granularidade mais fina possível.

2. **Perspectiva dinâmica da disponibilização de agregados:** demonstra o fluxo de dados existente para a disponibilização de agregados.

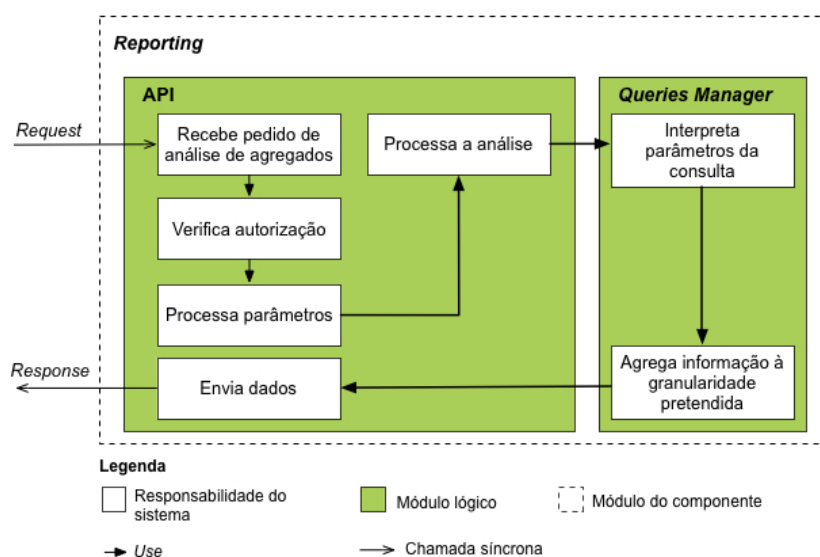


Figura 5.9: Perspectiva dinâmica da arquitectura - Disponibilização de agregados

O módulo lógico API recebe o pedido para efectuar a agregação de informação (Figura 5.9) e posteriormente processa os parâmetros recebidos, que só é realizado caso o cliente possua autorização para aceder aos dados. Estes parâmetros podem indicar, por exemplo, que se deseja obter um indicador-chave agregado por semestre, filtrado por utilizador. O pedido é enviado para o módulo lógico *Queries Manager* para que seja efectuada a interpretação dos parâmetros do pedido e a posterior agregação da informação, para que de seguida seja enviada para o cliente.

5.3.3 Perspectiva física e tecnológica

Neste subsecção será analisada a distribuição das responsabilidades analisadas anteriormente (Secções 5.3.1 e 5.3.2) por diferentes máquinas físicas e quais as tecnologias que dão suporte à implementação das funcionalidades do sistema.

Desta perspectiva não existe uma distinção arquitectural entre a vertente B2C e B2B, uma vez que estas assentam na mesma infra-estrutura, existindo apenas uma distinção entre a lógica aplicacional.

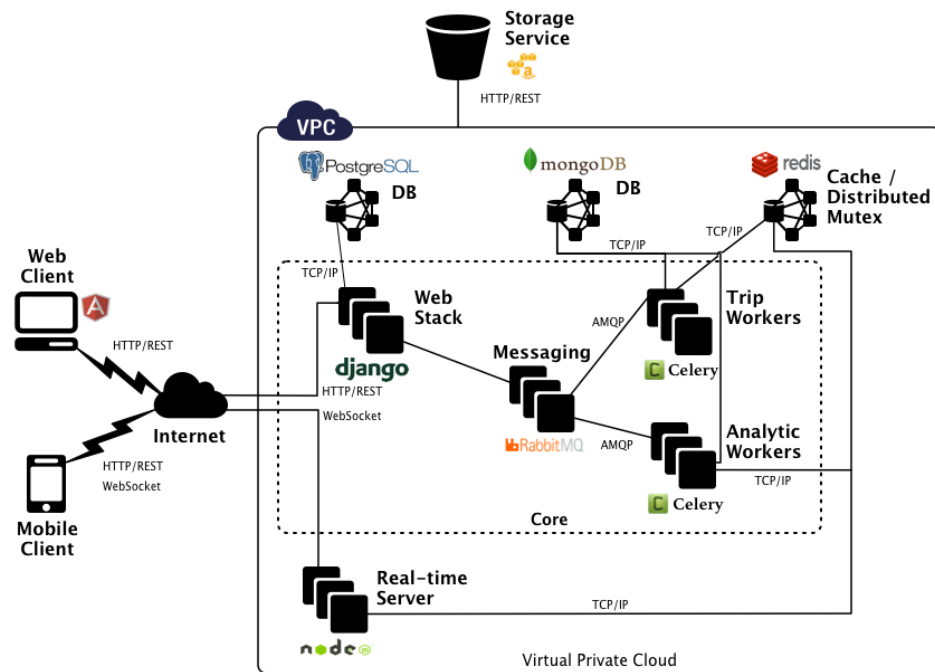


Figura 5.10: Perspectiva física e tecnológica

Mobile Client Utiliza a aplicação móvel desenvolvida para iOS, que regista os dados sensoriais relativos às viagens realizadas pelos utilizadores e comunica essa informação ao sistema através dos serviços REST. Através do *smartphone* existe também a possibilidade de registar viagens em tempo-real, utilizando para isso a tecnologia de *WebSockets*¹ para as comunicar com o *Real-time Server*.

Web Client Acede através do *browser* à interface web fornecida pelo EcoMobile, para realizar a manipulação e visualização da informação. É utilizada a framework de JavaScript, *AngularJS*², para o desenvolvimento do *frontend* web. Esta permite uma estruturação do código do lado do cliente através da implementação da *design pattern* MVW, bem como diminui a carga de processamento do lado do servidor, uma vez que permite passar parte da lógica aplicacional para o lado do cliente.

Web Stack É responsável por toda a comunicação *request-response* entre o EcoMobile e os clientes web e móveis. É aqui que se encontra grande parte da lógica de negócio do sistema e também a aplicação web disponibilizada

¹ *WebSockets*: é uma tecnologia que permite a comunicação bidirecional por canais *full-duplex* sobre um único *socket* TCP (*Transmission Control Protocol*).

² AngularJS: <https://angularjs.org/>

aos utilizadores, bem como os respectivos serviços REST para manipulação da informação. Este componente é desenvolvido utilizando uma *framework open-source*, escrita na linguagem Python, denominada *Django* ³. Esta *framework* faz uso da *design pattern* MVT (*Model-View-Template*), baseada no modelo *MVC* (*Model-View-Controller*), que permite dividir os módulos de código em grupos lógicos, por forma a torná-los mais flexíveis às mudanças. Este componente pode ser replicado horizontalmente, em caso de *bottleneck*, através da implementação de um *load balancer* para realizar o balanceamento da carga da aplicação.

É aqui que se encontram os serviços que disponibilizam as funcionalidades de comparação de trajectos, recomendação de trajectos e *reporting*.

Messaging Esta máquina incorpora o serviço de mensagens, através de uma tecnologia *open-source* denominada *RabbitMQ* ⁴, que permite o envio de mensagens assíncronas para processamento em *background*. Uma das grandes vantagens do sistema de mensagens *RabbitMQ* deve-se ao facto de este possuir a capacidade de se distribuir por diversas máquinas (*clusters*), garantindo assim maior escalabilidade e desempenho.

Analytic Workers Uma vez que o desacoplamento dos componentes deve ser garantido, tomou-se a decisão arquitectural de separar os *workers* que realizam o processamento de mensagens assíncronas por funções, visto que estes podem ter necessidades de escalabilidade diferentes. Assim, as funcionalidades implementadas neste estágio foram introduzidas num novo *worker* (***Analytic Workers***). Este componente terá a responsabilidade de realizar a agregação dos dados sensoriais das viagens e executar algoritmos de detecção de padrões. Uma máquina permite por defeito instanciar o número de *workers* igual ao número de *cores* da máquina, no entanto este número pode ser adaptado consoante as necessidades dos *workers*, bem como também poderão ser replicados horizontalmente.

Aqui encontram-se os módulos de software *Classification* e *Analytics*.

PostgreSQL Armazena os dados operacionais do sistema. Para a implementação deste componente é utilizado o sistema de gestão de dados *PostgreSQL* ⁵, um sistema *open-source* que possui diversos mecanismos de replicação, como é o caso do *Postgres-XC* (*eXtensible Cluster*) ⁶ e de outras

³*Django*: <https://www.djangoproject.com/>

⁴*RabbitMQ*: <http://www.rabbitmq.com/>

⁵*PostgreSQL*: <http://www.postgresql.org/>

⁶*Postgres-XC*: <http://wiki.postgresql.org/wiki/Postgres-XC>

extensões que permitem adicionar funcionalidades consoante as necessidades de utilização. No caso de existir um *bottleneck* na base de dados, esta encontra-se replicada por forma a permitir escalar as leituras.

No âmbito deste trabalho este é utilizado para representar o modelo de dados para suporte à implementação do módulo de software *Reporting*.

MongoDB O sistema de gestão de dados *MongoDB* ⁷ é um sistema *open-source*, orientado a guardar dados como documentos. Uma das principais características das bases de dados NoSQL é o facto de estas conseguirem ser distribuídas horizontalmente e no caso do *MongoDB* de forma automática (*auto-sharding*), como explicado na secção 3.1.4 . Esta base de dados possui nesta arquitectura a responsabilidade de dar suporte ao armazenamento do trabalho que é realizado pelos *Analytic Workers*. O *MongoDB* encontra-se replicado horizontalmente, na sua configuração mínima, por forma a escalar as leituras e escritas. Este pode ser facilmente replicado consoante as necessidades do sistema, através da adição de novos nós ao *cluster*.

É aqui que se encontram os modelos de dados dos módulos de software *Classification* e *Analytics*.

Cache/Distributed Mutex Para a implementação de mecanismos de *caching* e controlo de concorrência de processos distribuídos é utilizado o *Redis* ⁸. Este é uma tecnologia designada como uma *key-value in-memory data store*, ou seja, um mecanismo de armazenamento em memória de procura por chave. O *Redis* possui a característica de permitir escritas e leituras de dados de forma muito rápida. A utilização do *Redis* como mecanismo de *caching* melhora a escalabilidade do sistema, uma vez que retira a necessidade de estar constantemente a processar os mesmos pedidos.

5.3.4 Resumo do desenho arquitectural

Durante o processo de desenho da solução arquitectural demonstrada neste capítulo, o aspecto da escalabilidade foi um factor sempre presente, tendo sido seguida a estratégia *Design-Implement-Deploy* (D-I-D) [47] - projectar a escalabilidade da solução de 20x a infinito da capacidade, implementar para 3x da capacidade e colocar em produção para 1.5x da capacidade desejada.

Toda solução desenhada encontra-se perfeitamente desacoplada e replicada, tanto ao nível aplicacional como ao nível físico, garantindo assim as diferentes necessidades de escalabilidade de cada componente e a robustez de todo o sistema. De salientar também que a arquitectura projectada possui a

⁷ *MongoDB*: <http://www.mongodb.org/>

⁸ *Redis*: <http://redis.io/>

capacidade de escalar horizontalmente todos os seus componentes, sem que seja necessário realizar alterações ao nível aplicacional, físico ou tecnológico, através da utilização da técnica de balanceamento de carga, replicação dos componentes aplicacionais e das bases de dados, utilização de um sistema de mensagens horizontalmente distribuído e de um mecanismo de *caching* em memória.

Capítulo 6

Trabalho desenvolvido

Neste capítulo é demonstrado o trabalho desenvolvido no decorrer deste estágio. Para além do desenvolvimento das actividades que visam cumprir os objectivos planeados para este estágio, o estagiário esteve também inserido em diversas reuniões para tomada de decisões sobre o produto a desenvolver, bem como na realização de actividades, com a equipa de desenvolvimento, para análise de requisitos da aplicação móvel, aplicação web e respectiva prototipagem do produto final.

6.1 Classificação de trajectos

No sentido de realizar os requisitos funcionais relativos à comparação de trajectos semelhantes (Tabela 4.3), foi desenvolvido um algoritmo para detecção de padrões geoespaciais entre trajectos de viagens.

O principal objectivo deste algoritmo é conseguir classificar trajectos semelhantes e agrupá-los, por forma a permitir aos condutores visualizarem a evolução da sua economia e segurança ao longo da mesma rota. A título de exemplo, observar qual a evolução da segurança e economia de um condutor no seu trajecto diário casa-trabalho.

Aqui será demonstrada a abordagem utilizada para a implementação deste algoritmo, que teve por base o estudo realizado na secção 3.1.3, e a sua respectiva validação.

6.1.1 Definição do problema

A definição da similaridade entre objectos em movimento pode ser uma tarefa bastante complexa, uma vez que existe um conjunto de características que podem fazer variar as amostras ao longo do tempo, tais como a velocidade de

deslocação, o local de início e fim dos trajectos, a frequência de amostragem e o erro de GPS. Antes da elaboração da solução para o problema, foi definido formalmente o problema e quais os requisitos que o algoritmo deveria cumprir tendo em conta as características dos dados geoespaciais.

A abordagem utilizada para a resolução deste problema teve como referência o modelo CRISP-DM, estudado na secção 3.1.3.

Objectivo

Conseguir realizar a detecção de trajectos semelhantes das viagens realizadas pelos utilizadores e criar conjuntos de trajectos semelhantes que permitam aos utilizadores visualizarem a sua evolução e compararem as métricas de condução ao longo dessas rotas.

Trajectos semelhantes

São considerados trajectos semelhantes aqueles que têm em comum a mesma zona de início e fim e que possuem 80%¹ do padrão geoespacial igual.

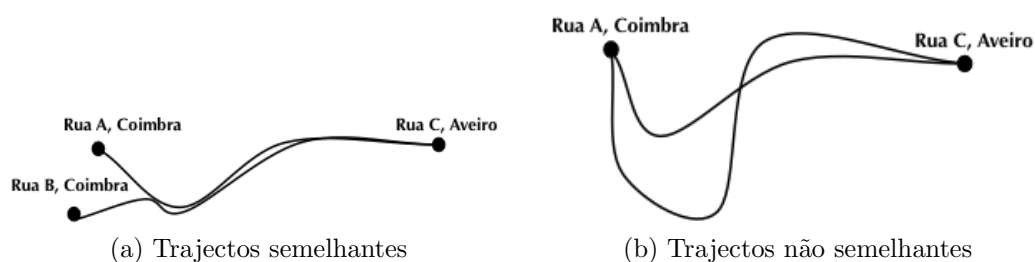


Figura 6.1: Representação de trajectos semelhantes e não semelhantes

Características dos dados

Uma vez que estamos a lidar com dados sensoriais móveis, as amostragens relativas a cada viagem podem conter erros associados (e.g., erro de GPS, frequência de amostragem). Dependendo da qualidade do dispositivo que recolhe a informação sensorial poderão existir erros mais significativos ao nível das posições GPS obtidas, bem como perdas de sinal em determinadas zonas dos trajectos, o que diminui a precisão das zonas geográficas dos trajectos.

Para contornar este problema foi criado o conceito de grupo de coordenadas semelhantes. Este permite reduzir a granularidade das coordenadas GPS,

¹Valor considerado empiricamente por forma a reduzir a dispersão dos conjuntos de viagens criados.

de forma a permitir agrupar as coordenadas, que a uma granularidade de 100 metros possuem o mesmo valor de latitude e longitude.

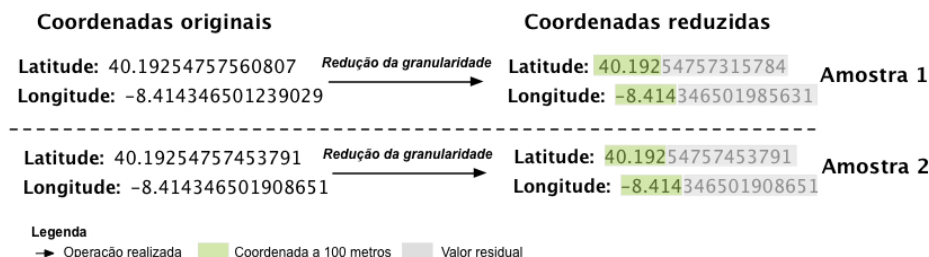


Figura 6.2: Redução da granularidade das coordenadas GPS

Como forma de exemplificar o conceito de grupo de coordenadas semelhantes, a figura 6.2 representa o resultado da operação que é realizada sobre as coordenadas. Pode observar-se que as coordenadas originais da Amostra 1 e da Amostra 2 ao nível de granularidade obtido pelo GPS são totalmente distintas, no entanto em termos de posição geográfica estas situam-se praticamente na mesma zona geográfica. Isso é verificado aquando a realização da operação de redução da granularidade, onde a coordenada original passa para uma coordenada reduzida (a 100 metros de granularidade), passando as duas amostra a serem consideradas coordenadas iguais.

Requisitos do algoritmo

O algoritmo a desenvolver deve possuir a capacidade de lidar com os seguintes aspectos:

- **Diferentes frequências de amostragem:** a frequência de amostragem durante a realização dos trajectos pode variar consoante a qualidade do sinal de GPS obtido, a qualidade do dispositivo móvel e as configurações definidas na aplicação móvel.
- **Diferentes tamanhos entre amostras:** as amostras obtidas entre a realização dos trajectos podem possuir diferentes tamanhos. Isto deve-se em primeiro lugar à frequência de amostragem e em segundo lugar às características inerentes a uma viagem. Por exemplo, supondo uma viagem casa-trabalho, um dia o condutor pode estacionar o carro exactamente no local de trabalho e noutra dia estacionar a 200 metros de distância do local de trabalho, o que não invalida o resto do trajecto efectuado.

- **Outliers**²: podem existir *outliers* ao longo da viagem que introduzem posições geográficas não relevantes para a análise. Por exemplo, um condutor pode ter de efectuar uma volta completa a uma rotunda devido ao trânsito, ou mesmo realizar um pequeno desvio no trajecto devido a obras.
- **Direcção dos trajectos**: é necessário identificar e ponderar a direcção com que o trajecto é realizado.

6.1.2 Desenho da solução

Após a definição e análise concreta do problema, foi realizada a prototipagem de possíveis soluções para a resolução do problema apresentado anteriormente.

As características presentes neste problema são bastante semelhantes a um problema de *clustering*, estudado na secção 3.1.3, uma vez que o objectivo a alcançar é detectar padrões de semelhança entre trajectos, classificá-los e por fim criar grupos de trajectos semelhantes (*clusters*).

Distância de semelhança

Qualquer problema de *clustering* é baseado num tipo de distância de semelhança para classificar grupos de pontos ou objectos. Assim tornou-se necessário definir a medida de semelhança que melhor se adapta a este problema. Existem duas classes principais de medidas de distância (Secção 3.1.3), distâncias Euclidianas e não Euclidianas.

Para a solução deste problema enquadra-se uma medida de distância não Euclidiana, uma vez que um trajecto pode ser visto como um objecto e não um ponto num determinado espaço. Uma medida de similaridade que tenha por base uma medida de distância como a *edit distance* (e.g., *Longest Common Subsequence* (LCS)), possui propriedades interessantes para o problema de detecção de dois trajectos semelhantes [48].

A utilização do LCS, para classificar a semelhança entre trajectos, consegue cumprir os requisitos definidos anteriormente para o algoritmo de classificação. Esta permite encontrar a maior subsequência, de grupos de coordenadas GPS, que são comuns entre duas sequências de coordenadas GPS.

Definição da medida de distância A medida de distância terá por base o algoritmo LCS. A implementação deste algoritmo, com utilização de programação dinâmica, possui um tempo de execução $O(n \times m)$, onde n e m

² *Outlier*: é um ponto de observação que está distante de outras observações realizadas.

são o número total de coordenadas dos trajectos. Neste caso é ainda possível realizar uma optimização ao nível da redução do espaço computacional necessário, uma vez que apenas é necessário saber o tamanho (factor) da maior sequência encontrada, para calcular a distância.

Assim a medida de distância é definida como:

$$Dist = 1 - LCS(trajecto_ref, trajecto_comp) / factor \quad (6.1)$$

Dist : Valor da distância de semelhança dois trajectos comparados entre 0 e 1
LCS : Implementação do LCS utilizando programação dinâmica
factor : Tamanho máximo entre os dois trajectos
trajecto_ref : Coordenadas GPS do trajecto de referência
trajecto_comp : Coordenadas GPS do trajecto a comparar

O valor da medida de distância é um valor real que pode variar entre 0 e 1, sendo que quanto mais este valor se aproximar de 0, menor é a distância entre os trajectos logo maior é a semelhança entre os trajectos.

O algoritmo de cálculo da medida de distância entre os trajectos pode ser observado de seguida (Algoritmo 1).

Algorithm 1 Algoritmo de cálculo da medida de distância

Pré-condições: *trajecto_ref* - Coordenadas do trajecto de referência

Pré-condições: *trajecto_comp* - Coordenadas do trajecto a classificar

Pós-condições: Retorna *Dist*

```

function MEDIDA_DISTANCIA(trajecto_ref, trajecto_comp)
  len_ref ← tamanho(trajecto_ref)
  len_comp ← tamanho(trajecto_comp)
  actual ← aloca(len_comp)
  for i = 0; i < len_ref; i = i + 1 do
    anterior ← actual
    for j = 0; j < len_comp; j = j + 1 do
      if trajecto_ref[i] = trajecto_comp[j] then
        actual[i+1] ← anterior[i] + 1
      else
        actual[i+1] ← max(actual[i], anterior[i+1])
      end if
    end for
  end for
  factor ← max(len_ref, len_comp)
  return 1 - actual[len_comp] / factor
end function

```

Classificação

A classificação da semelhança dos trajectos é realizada por um algoritmo de *clustering*, que utiliza a medida de distância definida anteriormente. A implementação deste algoritmo de classificação é inspirada no algoritmo de *clustering* aglomerativo. No entanto este possui alterações, relativamente ao original, com o intuito de otimizar o processo de classificação tendo em conta o problema presente (problema de carácter *online*).

A principal alteração relaciona-se com o trajecto que representa o *cluster* para realizar a classificação com novos trajectos. Aquando a classificação de um novo trajecto, ao invés de classificar o novo trajecto com todos os trajectos presentes nos *clusters* candidatos, é feita a classificação apenas com o trajecto que deu origem ao *cluster*. É assim colocado um *trade-off* entre velocidade de processamento computacional e precisão total na classificação.

O algoritmo de classificação de um novo trajecto no respectivo *cluster* pode ser observado de seguida (Algoritmo 2).

Algorithm 2 Algoritmo de classificação de trajectos

Pré-condições: *trajecto* - Coordenadas do novo trajecto a classificar

Pós-condições: Classifica *trajecto*

```

function CLASSIFICA(trajecto)
  raio_procura ← 800 metros
  len_trajecto ← tamanho(trajecto)
  coordenadas_iniciais ← trajecto[0]
  coordenadas_finais ← trajecto[len_trajecto]
  trajecto ← reduz_dimensionalidade(trajecto)
  agregados ← processa_agregados(trajecto)
  candidatos ← obtem_clusters_candidatos(coordenadas_iniciais,
  coordenadas_finais, raio_procura)
  cluster_actual ← ∅
  for candidato in candidatos do
    representante ← obtem_representante(candidato)
    distancia ← MEDIDA_DISTANCIA(trajecto, representante)
    if distancia ≤ 0.2 then
      cluster_actual ← candidato
    end if
  end for
  if cluster_actual = ∅ then
    cria_novo_cluster(trajecto, agregados)
  else
    insere_trajecto(cluster_actual, trajecto, agregados)
  end if
end function

```

No processo de classificação de um novo trajecto é importante salientar o facto de que após a inserção da nova viagem no seu respectivo *cluster* é

efectuada uma actualização, de forma consistente, dos indicadores-chave que avaliam aquela rota.

Ao efectuar esta acção permite manter sempre pré-calculados os indicadores-chave de segurança, economia, distância e tempo de viagem daquela rota. Estes são os indicadores que irão servir para avaliar e recomendar as melhores rotas.

Modelo de dados

Nesta secção serão apresentados os modelos de dados desenvolvidos, que visam dar suporte à funcionalidade de classificação de trajectos. O modelo de dados que dá suporte a esta funcionalidade, foi implementado no *MongoDB*, pelo que este se enquadra num modelo não-relacional representado no formato JSON, incorporados em colecções que representam o conjunto de documentos JSON.

A descrição completa dos modelos de dados pode ser consultada no Anexo F.

Colecção *trips_geospatial* Representa as coordenadas GPS do trajecto de uma viagem, após realizada a redução da dimensionalidade, agrupadas por semelhança e com os correspondentes indicadores-chaves para cada uma das posições geográfica.

trips_geospatial
trip_id: BIGINT [PK]
cluster_id: BIGINT
start_coordinates: STRUCT
end_coordinates: STRUCT
global_metrics: STRUCT
positions: ARRAY

Figura 6.3: Modelo de dados da colecção *trips_geospatial*

Colecção *trips_clusters* Representa os *clusters* de viagens semelhantes criados após a sua classificação. Para além de representarem o conjunto de viagens semelhantes, permitem também armazenar os indicadores-chave que avaliam a rota a que pertencem os trajectos.

trips_clusters
cluster_id: BIGINT [PK]
start_coordinates: STRUCT
end_coordinates: STRUCT
trips: ARRAY
cluster_measures: STRUCT
best_routes: STRUCT

Figura 6.4: Modelo de dados da colecção *trips_clusters*

Implementação

Como referido na arquitectura (Secção 5.3.1) a funcionalidade de classificação de trajectos encontra-se implementada, ao nível do *backend*, num *worker* que é implementado com suporte à tecnologia *Celery* para o *Django*. Este serve como *broker* para processar as tarefas que se encontram na fila de mensagens do *RabbitMQ*.

6.1.3 Validação

Pretende-se nesta secção validar o algoritmo de classificação implementado. Para a elaboração dos testes de validação foi utilizado um *dataset* constituído por 594 viagens realizadas por condutores reais ao longo de 3 meses.

Como forma de garantir uma validação correcta, foi utilizado um dos três critérios de testes estudados na secção 3.1.3, a validação utilizando índices internos. Estes permitem validar os *clusters* através de dois indicadores [49]:

1. **Coesão (*cluster cohesion*):** mede o quão próximos se encontram os objectos dentro dos *clusters*.
2. **Separação (*cluster separation*):** mede o quão distinto ou bem separados se encontram os *clusters* uns dos outros.

Resultados

Para realizar a validação descrita anteriormente, foi criado um *script* que avalia automaticamente a coesão e separação dos *clusters* existentes. A validação completa de cada *cluster* pode ser consultada no Anexo H.

- **Avaliação da coesão:** para cada *cluster* existente é calculada a distância, utilizando a medida de distância, entre todos os trajectos pertencentes ao *cluster*.

- **Avaliação da separação:** para cada *cluster* existente é calculada a distância, utilizando a medida de distância, entre todos os trajectos de todos os *clusters*.

Tabela 6.1: Resumo da avaliação da coesão dos *clusters*

Mínimo	Máximo	Média	Desvio padrão
0	0,2	0,10	0,02

Tabela 6.2: Resumo da avaliação da separação dos *clusters*

Mínimo	Máximo	Média	Desvio padrão
0,93	0,98	0,96	0,08

Uma vez que durante a classificação, se considerou um factor de aceitação da medida de distância igual a 0.2 (equivalente a possuir 80% de zonas semelhantes entre trajectos), ao efectuar a avaliação da coesão o valor nunca poderia ser superior a 0.2.

Como observado na tabela 6.1 o valor médio da coesão é igual a 0.1 com um desvio padrão baixo e um valor máximo igual a 0.2, o que significa que não existem *outliers* e que todos os trajectos presentes nos *clusters* se encontram bem classificados

Já ao avaliar a separação, o valor desta nunca poderia ser inferior a 0.8, devido ao factor de aceitação da medida de distância. Na tabela 6.2 observa-se que o valor médio da separação é de 0.96 com um desvio padrão e um valor mínimo de 0.93, o que significa que não existem *outliers* e que todos os trajectos presentes nos *clusters* são distintos em comparação aos restantes *clusters*.

Performance

Foram também realizados testes de *performance* aos algoritmos implementados. Nomeadamente ao algoritmo responsável pela redução da dimensionalidade e cálculo de agregados, bem como ao algoritmo de classificação. Os testes foram realizados com viagens que possuíam, em média, a duração de 30 minutos.

O ambiente de testes utilizado foi constituído por uma máquina com um processador Intel Core i5 2,5 GHz com 16 GB de RAM.

Tabela 6.3: *Performance* do algoritmo de redução da dimensionalidade e cálculo de agregados

<i>Runs</i>	Média (tempo em segundos)	Desvio padrão
30	0,012	8.81×10^{-6}

Tabela 6.4: *Performance* do algoritmo de classificação

<i>Runs</i>	Média (tempo em segundos)	Desvio padrão
30	0,14	0,002

Os números observados nas tabelas 6.3 e 6.4 indicam que os algoritmos implementados atendem ao problema e não serão um problema de escalabilidade para a arquitectura desenvolvida. Os tempos de execução sequencial destes são reduzidos, sendo que no ambiente de produção, estes poderão efectuar processamento paralelo e distribuído.

6.2 Comparação de trajectos

6.2.1 Desenho da solução

A comparação de trajectos tem como principal finalidade permitir ao utilizador comparar indicadores-chave de condução em zonas geográficas iguais, para que seja possível identificar e avaliar a condução realizada nos trajectos. A partir da implementação da classificação de trajectos, torna-se possível saber quais são os trajectos semelhantes, no entanto é ainda necessário alinhar as zonas geográficas iguais entre os trajectos que vão ser comparados.

Para resolver o problema de alinhamento de trajectos, foi utilizada uma abordagem inspirada nos algoritmos de alinhamento de sequências. Esses algoritmos são designados de alinhamento local (*Smith-Waterman*) e alinhamento global (*Needleman-Wunsch*) [50] [51] [52].

Alinhamento local

É aplicado o algoritmo de alinhamento local quando se pretende comparar trajectos que se encontram no mesmo *cluster*, ou seja, são trajectos que são considerados semelhantes. A aplicação deste algoritmo permite que se encontrem as regiões locais com alto nível de similaridade. Neste caso em

específico são as zonas dos trajectos que possuem as mesmas coordenadas GPS. A implementação que faz uso de programação dinâmica permite construir vários alinhamentos, neste caso é sempre feito o *traceback* a partir da posição que possui o maior *score*.



Figura 6.5: Exemplo do alinhamento local entre trajectos do mesmo *cluster*

Alinhamento global

Este algoritmo é aplicado quando se pretende comparar trajectos que não se encontram no mesmo *cluster*, mas que possuem algumas zonas geográficas iguais. Com a aplicação deste algoritmo é permitido alinhar os trajectos desde o seu início até ao fim, permitindo alinhar várias zonas iguais ao longo dos dois trajectos.

No entanto, na comparação de trajecto de diferentes *clusters*, é introduzido um *trade-off* entre custo de processamento do alinhamento e o factor de semelhança entre os trajectos a comparar. Para isso é sempre calculada a medida de distância entre os dois trajectos e o alinhamento global é apenas efectuado se entre os dois trajectos existir um factor de semelhança de 30%. A implementação que faz uso de programação dinâmica permite construir vários alinhamentos, neste caso é sempre feito o *traceback* a partir da posição que possui o maior *score*.

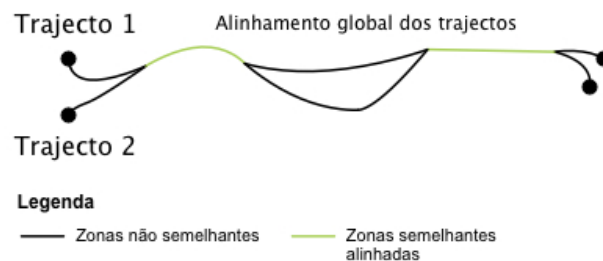


Figura 6.6: Exemplo do alinhamento global entre trajectos que possuem zonas iguais

Performance

Foram realizados testes de *performance* dos algoritmos de alinhamento local e global. A partir do *dataset* de 594 viagens foram seleccionadas para testes 10 viagens que possuíam, em média, a duração de 30 minutos, tendo sido aplicado o alinhamento entre todas as viagens.

O ambiente de testes utilizado foi constituído por uma máquina com um processador Intel Core i5 2,5 GHz com 16 GB de RAM.

Tabela 6.5: *Performance* do algoritmo de alinhamento local

<i>Runs</i>	Média (tempo em segundos)	Desvio padrão
30	0,45	0,07

Tabela 6.6: *Performance* do algoritmo de alinhamento global

<i>Runs</i>	Média (tempo em segundos)	Desvio padrão
30	0,45	0,06

Os números observados nas tabelas 6.5 e 6.6 indicam que os algoritmos implementados atendem ao problema e não serão um problema de escalabilidade para a arquitectura desenvolvida. Os tempos de execução sequencial destes são reduzidos, sendo que no ambiente de produção, estes poderão efectuar processamento em paralelo e distribuído.

Tuning

Uma vez que a funcionalidade de comparação de trajectos requer algoritmos com algum peso computacional, esta abordagem requereu especial cuidado no trabalho de optimização. Para reduzir o processamento constante, na verificação da semelhança entre trajectos de diferentes *clusters* e alinhamento dos trajectos, implementou-se um mecanismo de *caching*.

Caching da semelhança Sempre que se pretende comparar dois trajectos, que são de diferentes *clusters*, é sempre guardada a medida de distância entre os dois trajectos, para que no futuro se evite novamente a realização do processamento da medida de distância.

Caching do alinhamento Sempre que são alinhadas as zonas geográficas dos dois trajectos a comparar, são sempre guardados os índices das zonas

geográficas alinhadas, para evitar o constante alinhamento entre os trajectos.

6.2.2 Interface e funcionalidades

Nesta secção será demonstrado o trabalho que foi realizado para a implementação das funcionalidades que permitem a visualização da evolução da economia e segurança do condutor ao longo de uma determinada rota e a comparação entre trajectos semelhantes do condutor e de amigos da aplicação.

Na figura 6.7 é possível visualizar o interface desenvolvido e implementado na vertente B2C do projecto EcoMobile. Este faz uso da classificação de trajectos semelhantes (Secção 6.1) para permitir ao utilizador visualizar a evolução da sua condução ao longo das suas viagens mais frequentes. Para além da visualização da evolução da sua condução, é possível o utilizador realizar comparações de métricas de condução (e.g., segurança, economia, velocidade) ao longo de dois trajectos. Por exemplo, um utilizador pode comparar o trajecto menos económico de casa para trabalho com o trajecto mais económico e visualizar as zonas geográficas onde a sua condução teve pior economia.

O utilizador pode também aplicar um conjunto de filtros (e.g., viagens semelhantes dos amigos, viagens com o mesmo início e fim) sobre todas as viagens (Figura 6.8), por forma a aumentar o horizonte da comparação entre trajectos.

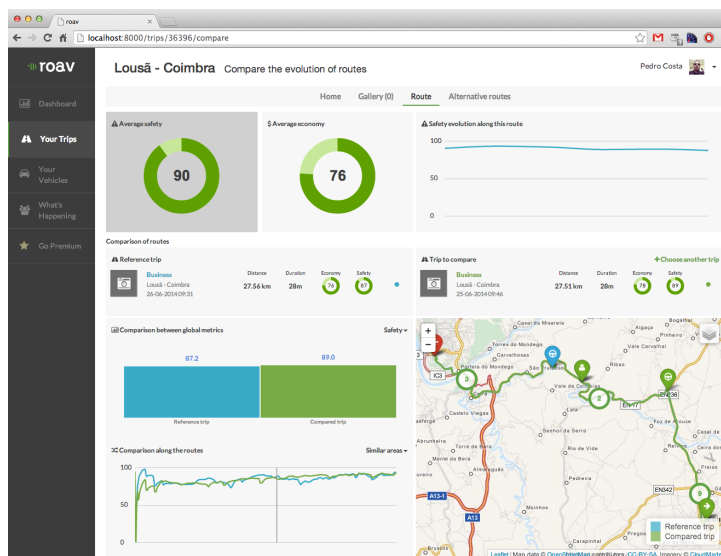


Figura 6.7: Interface desenvolvido para visualizar a evolução e comparação de trajectos

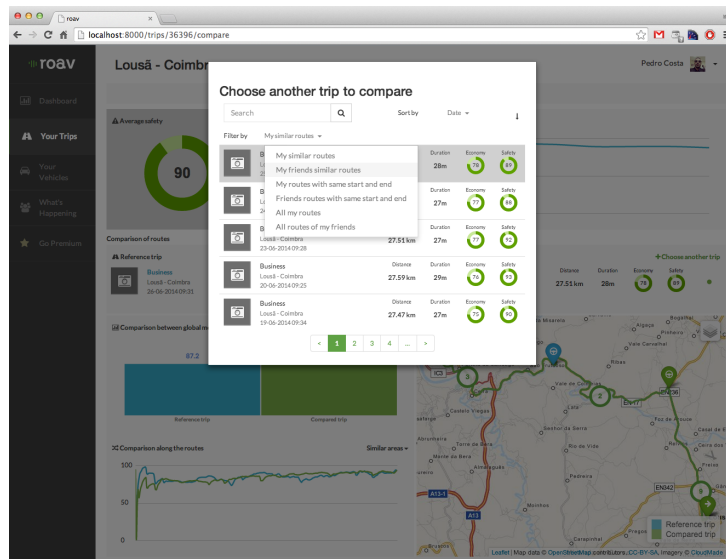


Figura 6.8: Selecção dos trajectos para comparação

Funcionalidades desenvolvidas

Como referido na secção da arquitectura (Secção 5.3.1) as funcionalidades de comparação de trajectos encontram-se, ao nível do *backend*, implementadas no módulo *Classifier* que se encontra no servidor *Django* e que efectua o processamento dos algoritmos em *runtime*. Já para a implementação de toda a componente de *frontend*, implementada no módulo *Comparison* (Secção 5.3.1), foi utilizada a *framework AngularJS*, que permite implementar lógica aplicacional do lado do cliente (JavaScript) e desenvolver o interface e estilo dos *templates* (HTML, CSS) de forma dinâmica. As funcionalidades desenvolvidas que permitem realizar os requisitos funcionais relacionados com a comparação de trajectos foram:

- Visualizar a média global da economia e segurança para os mesmos trajectos.
- Visualizar a evolução temporal da economia e segurança ao longo dos mesmos trajectos.
- Seleccionar trajectos para comparação:
 - Trajectos semelhantes do utilizador.
 - Trajectos semelhantes dos amigos do utilizador.
 - Trajectos do utilizador que possuem o mesmo início e fim.

- Trajectos dos amigos do utilizador que possuem o mesmo início e fim.
- Qualquer trajecto do utilizador.
- Escolher métricas de condução, como economia, segurança e velocidade, para comparar globalmente e ao longo dos trajectos.
- Filtrar a comparação ao longo dos trajectos por zonas geográficas iguais ou zonas globais.
- Visualizar no mapa os eventos de condução, como travagens, acelerações e curvas bruscas, entre os trajectos a comparar.

6.3 Recomendação de trajectos

A funcionalidade de recomendação de trajectos tem como principal objectivo, recomendar aos utilizadores os melhores trajectos para uma determinada rota, por forma a informar e auxiliar os utilizadores da existência de trajectos mais económicos, seguros, curtos e rápidos para as suas viagens mais comuns (e.g., casa-trabalho).

Neste sentido foi desenvolvido um algoritmo de recomendação, que tem por base informações colaborativas que são recolhidas pelos utilizadores, através da aplicação móvel para o sistema.

6.3.1 Desenho da solução

Como demonstrado na arquitectura (Figura 5.4) e também referido durante a explicação do algoritmo de classificação (Secção 6.1.2), os indicadores-chave que avaliam cada rota já se encontram pré-calculados. Assim, o algoritmo de recomendação apenas interpreta a informação que é pré-calculada pelo classificador, para efectuar a recomendação de trajectos.

A implementação do algoritmo de recomendação de trajectos pode ser consultada no Anexo G secção G.1.

Performance

Foram realizados testes de *performance* ao algoritmo de recomendação. Os testes foram realizados para 5 rotas que possuíam, em média, a duração de 30 minutos.

O ambiente de testes utilizado foi constituído por uma máquina com um processador Intel Core i5 2,5 GHz com 16 GB de RAM.

Tabela 6.7: *Performance* do algoritmo de recomendação

<i>Runs</i>	Média (tempo em segundos)	Desvio padrão
30	0,18	0,007

Os números observados na tabela 6.7 indicam que os algoritmos implementados atendem ao problema e não serão um problema de escalabilidade para a arquitectura desenvolvida. Os tempos de execução sequencial destes são reduzidos, sendo que no ambiente de produção, estes poderão efectuar processamento paralelo e distribuído.

6.3.2 Interface e funcionalidades

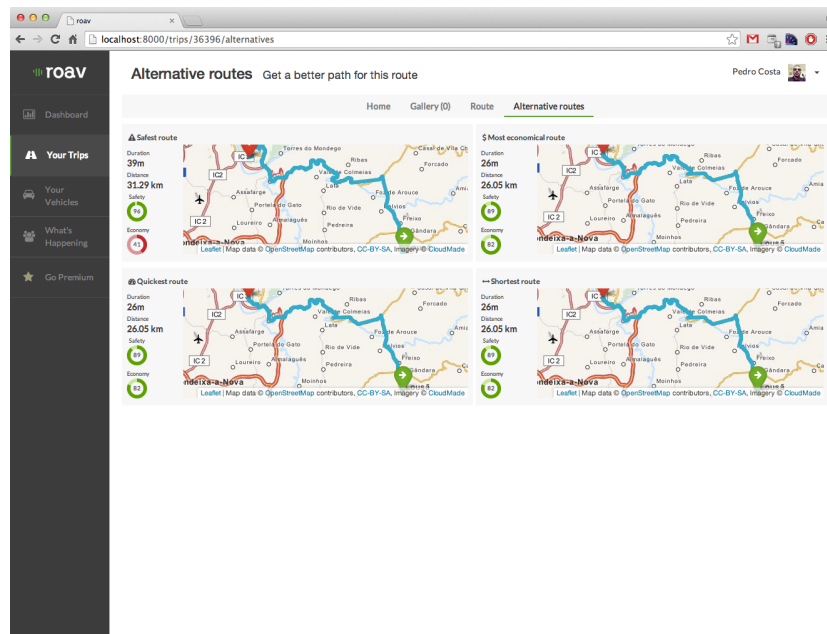


Figura 6.9: Interface desenvolvido para visualizar a recomendação de trajectos

Na figura 6.9 apresenta-se o interface desenvolvido e implementado na vertente B2C do projecto EcoMobile. Este utiliza o algoritmo explicado anteriormente (Secção G.1) para disponibilizar ao utilizador um interface que permite a visualização de melhores trajectos para uma determinada rota.

Funcionalidades desenvolvidas

Na secção da arquitectura (Secção 5.3.1) referiu-se que as funcionalidades de recomendação de trajectos encontram-se, ao nível do *backend*, implementadas no módulo *Recommendation* que se encontra no servidor *Django* e que efectua o processamento do algoritmo de recomendação em *runtime*. Para desenvolvimento de toda a componente de *frontend*, implementada no módulo *Recommendation* (Secção 5.3.1), foi utilizada a *framework* *AngularJS*, que permite implementar lógica aplicacional do lado do cliente (JavaScript) e desenvolver o interface e estilo dos *templates* (HTML, CSS) de forma dinâmica. As funcionalidades desenvolvidas que permitem realizar os requisitos funcionais relacionados com a recomendação de trajectos foram:

- Recomendação do trajecto mais seguro, mais económico, mais curto e mais rápido para uma determinada rota e respectiva informação da rota:
 - Correspondente média da segurança dos trajectos.
 - Correspondente média da economia dos trajectos.
 - Correspondente distância média dos trajectos.
 - Correspondente tempo médio de viagem dos trajectos.

6.4 Reporting

Nesta secção serão demonstradas as funcionalidades implementadas, na vertente B2B do projecto EcoMobile, que permitem a criação de *dashboards*, relatórios e análises personalizadas.

A implementação destas funcionalidades tem como principal objectivo, permitir a um gestor de frotas ou a um *business analyst*, possuir uma perspectiva de 360 graus sobre a sua organização, através da disponibilização de um conjunto de indicadores, que avaliam o negócio, quase em tempo-real.

6.4.1 Modelo de dados

Como especificado na arquitectura do componente *Analytics* (Figura 5.8), o processamento dos dados é realizado assim que é recebida uma viagem. Para o desenvolvimento das funcionalidades que permitissem aos utilizadores realizar a agregação de indicadores-chave sobre diferentes perspectivas, foi numa primeira instância, analisado o negócio, os dados e a sua granularidade, para conseguir criar um modelo que permitisse representar os factos à granularidade desejada sobre diversas dimensões (Anexo F secção F.2.1).

Para a representação da informação utilizou-se um modelo de dados implementado no *MongoDB*. Esta base de dados possui a capacidade de lidar com um grande número de escritas, uma vez que é capaz de escalar horizontalmente. Possui também operações que permitem o incremento e actualização dos campos de forma consistente e disponibiliza uma *framework* de agregação que permite realizar a agregação da informação de forma muito eficiente.

Este representa os factos e dimensões relativos às viagens que são realizadas pelos condutores de uma organização num determinado dia (Figura 6.10). Estes factos são actualizados, de forma consistente assim que é recebida a viagem de um condutor, permitindo que a informação se encontre disponível praticamente quando é comunicada e à granularidade desejada, uma vez que é utilizada a *design pattern* de um documento por utilizador por dia [53].

```

    "_id" : "2014-12-30/8",
    "meta" : {
      "date" : "2014-12-30",
      "day" : 30,
      "month" : 12,
      "year" : 2014,
      "semester" : 2,
      "quarter" : 4,
      "week_of_month" : 4,
      "org_id" : 5,
      "org_name" : "Sentilant",
      "user_id" : 3,
      "user_email" : "irenedavis@sentilant.com"
    },
    "metrics" : {
      "ac" : 12,
      "hb" : 15,
      "hc" : 9,
      "total_trips" : 15,
      "safety" : 76,
      "economy" : 82,
      "time_trav" : 8988.8,
      "dist_trav" : 245.4,
      "productivity" : 70,
      "speed" : 60
    }
  }

```

Diagram illustrating the data model structure with annotations:

- Dimensões** (Dimensions):
 - Tempo** (Time): "date", "day", "month", "year", "semester", "quarter", "week_of_month"
 - Organização** (Organization): "org_id", "org_name"
 - Conductor** (Conductor): "user_id", "user_email"
- Factos** (Facts): "ac", "hb", "hc", "total_trips", "safety", "economy", "time_trav", "dist_trav", "productivity", "speed"

Figura 6.10: Modelo de dados para agregação de métricas de condução

Para poder realizar os agregados dos indicadores a diversos níveis de granularidade e por diversas perspectivas foram implementados *pipelines* de agregação, utilizando a *framework* de agregação [54] disponibilizada pela base de dados *MongoDB*. Este *pipeline* de agregação possui características semelhantes ao modelo de programação MapReduce, no entanto encontra-se desenvolvido no *core* do *MongoDB*, pelo que é mais optimizado para trabalhar com esta base de dados. De seguida será demonstrado um dos muitos *pipelines* de agregação desenvolvidos. O *pipeline* representado na figura 6.11

agrega a evolução da segurança de um condutor, por fatia de tempo à granularidade temporal do ano. Este tem a sua fase inicial (*match*), onde realiza uma procura pela chave (id) do utilizador, para posteriormente passar para a parte de elaboração da agregação, tendo por base os documentos filtrados na fase de *match*.

Na fase de agregação (*group*) este calcula a média da segurança para cada ano existente e por fim constrói os documentos com os resultados agregados.

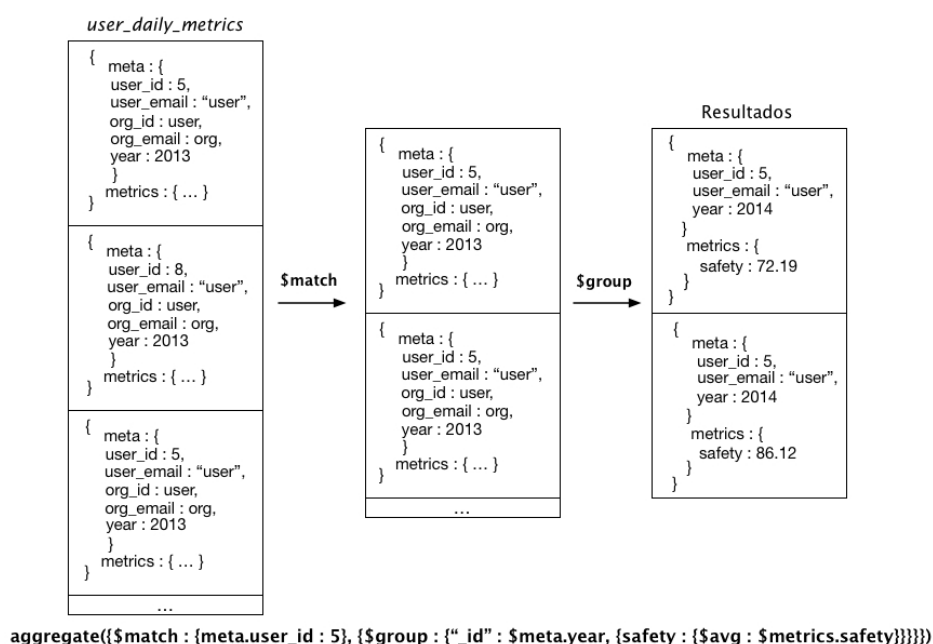


Figura 6.11: Exemplo de um *pipeline* de agregação

Como forma de otimizar os *pipelines* desenvolvidos, criaram-se índices para chaves das dimensões presentes no modelo de dados, por forma a otimizar a procura pelas chaves que são utilizadas mais frequentemente (e.g., *year*, *user_id*, *org_id*).

Na representação do modelo que permite ao utilizador criar análises, relatórios e *dashboards* personalizados foi implementado um modelo relacional no *PostgreSQL* (Figura 6.12). No desenho deste modelo existiu o cuidado de que este fosse o mais genérico e dinâmico possível, para que permitisse representar dinamicamente todo o conjunto de métricas e análises criadas pela organização, que permitem a criação dos *dashboards* e relatórios (Anexo F secção F.2.1).

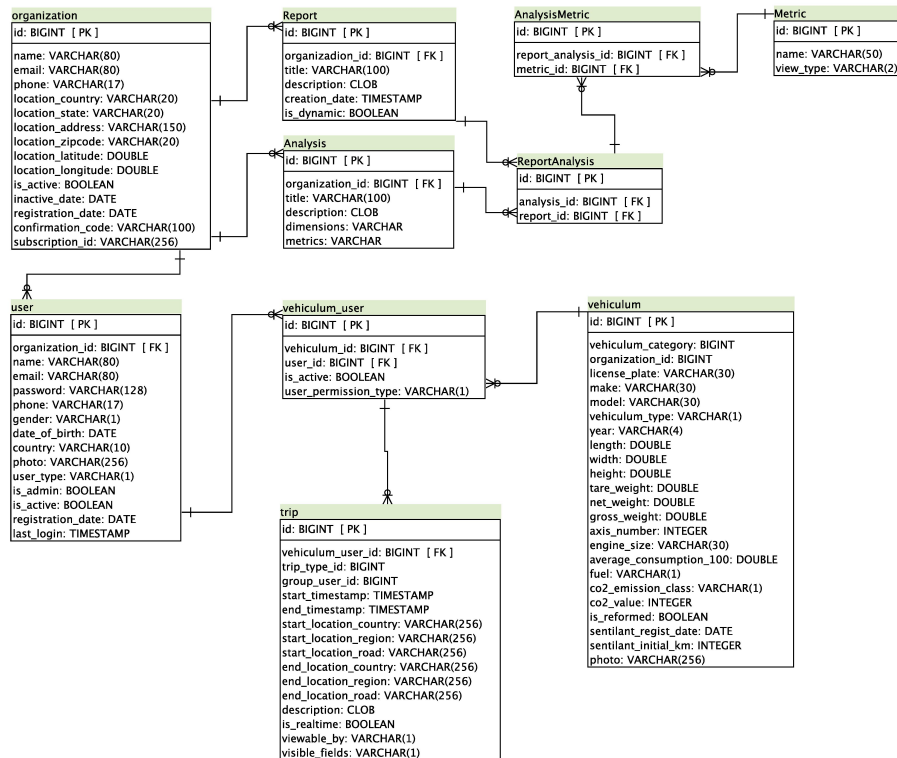


Figura 6.12: Modelo de dados da componente de *reporting* da versão B2B

Tabela 6.8: Descrição das entidades do modelo da componente de *reporting*

Entidade	Descrição
Organization	Representa uma organização.
User	Representa um utilizador da aplicação que pode ser um administrador ou um condutor.
Vehiculum	Representa um veículo da organização.
Trip	Representa uma viagem realizada por um determinado condutor num determinado veículo.
Analysis	Representa as análises personalizadas sobre os indicadores criadas para a organização.
Report	Representa os relatórios que foram criados para a organização.
Metric	Representa as métricas que são disponibilizadas pela aplicação ou criadas pela organização.

6.4.2 Interface e funcionalidades

Nesta secção serão demonstradas as funcionalidades implementadas na vertente B2B do projecto EcoMobile.

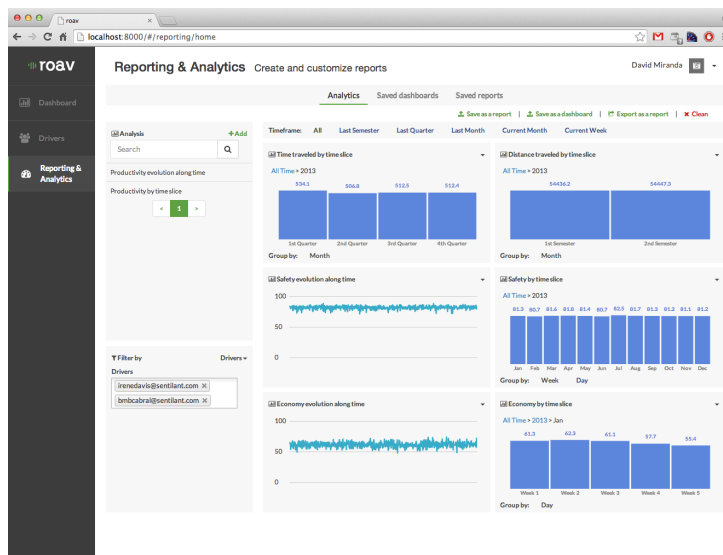


Figura 6.13: Interface que permite a criação de análises, relatórios e *dashboards* personalizados

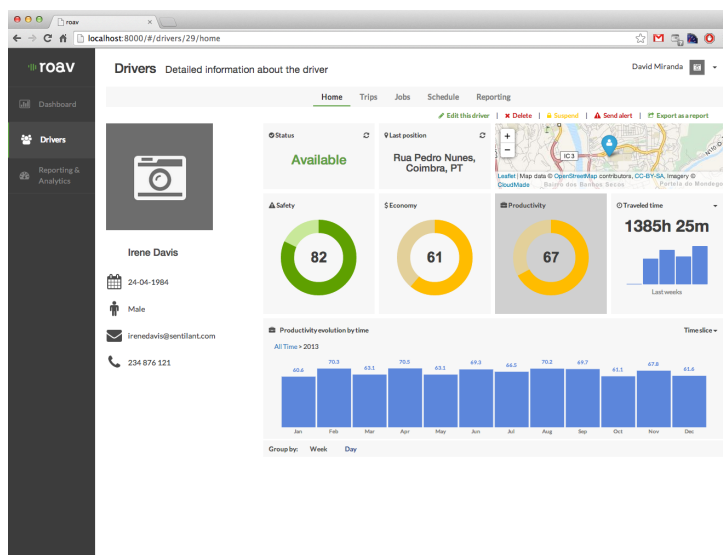


Figura 6.14: Funcionalidades de *reporting* incorporadas no perfil de cada condutor

Na figura 6.13 pode observar-se o interface que permite a um gestor de frotas analisar todo um conjunto de indicadores pelos diversos níveis de granularidade, aplicar filtros dimensionas (e.g., tempo, condutores, veículos), criar relatórios e *dashboards* personalizados e guardar estes como vistas pré-definidas, por forma a obter a informação mais pertinente nas situações mais oportunas.

A componente de *reporting* foi também incorporada no interface que permite visualizar a informação detalhada de cada condutor (Figura 6.14). Assim, quando o gestor acede ao perfil de um condutor, tem logo a possibilidade de visualizar e analisar a informação mais relevante respeitante ao condutor e criar imediatamente um relatório com a presente informação.

Funcionalidades desenvolvidas

Na secção da arquitectura (Secção 5.3.2) referiu-se que as funcionalidades de *reporting* encontram-se, ao nível do *backend*, implementadas nos módulos *Analytics* e *Reporting*. O primeiro encontra-se num *worker*, que é implementado com recurso à tecnologia *Celery* para o *Django*. Já o segundo encontra-se implementado no servidor *Django*, que possui toda a lógica que permite a criação das análises, *dashboards* e relatórios.

Para desenvolvimento da componente de *frontend*, implementada no módulo *Reporting* (Secção 5.3.2), foi utilizada a *framework AngularJS*, que permite implementar lógica aplicacional do lado do cliente (JavaScript) e desenvolver o interface e estilo dos *templates* (HTML, CSS) de forma dinâmica. As funcionalidades desenvolvidas que permitem realizar os requisitos funcionais relacionados com a criação de *dashboards* e relatórios foram:

- Procurar e seleccionar (*drag-and-drop*) as análises personalizadas que se desejam analisar.
- Criar relatórios e *dashboards*, a partir de análises personalizadas relativas às viagens dos condutores da organização.
- Visualizar e analisar, sobre diferentes perspectivas, as métricas de condução no perfil de cada condutor.

Capítulo 7

Testes

Nesta secção pretende-se demonstrar a validação das funcionalidades implementadas nas duas vertentes do projecto EcoMobile, através da realização de testes de aceitação com o *Product Owner*, ao longo de cada *sprint* de desenvolvimento e no final de cada *milestone*.

Realizaram-se também testes de carga com o intuito de validar os atributos de qualidade do sistema. Este género de testes foca-se em determinar a robustez, disponibilidade e confiabilidade de uma aplicação em condições extremas (*stress*) e identificar os componente que criam um *bottleneck* no sistema.

7.1 Testes de aceitação

Este género de testes tem como principal objectivo verificar se os requisitos funcionais implementados funcionam correctamente e da forma esperada pelo utilizador final. A elaboração destes testes ao longo do desenvolvimento deste estágio permitiu que os responsáveis pelo produto contribuíssem de forma constructiva para o progresso do estágio e no planeamento das *sprints* ao longo do seu desenvolvimento.

Os testes de aceitação realizados podem ser consultados no Anexo H secção H.2.

7.2 Testes de carga

No decorrer deste estágio não houve a possibilidade de realizar estes testes num verdadeiro ambiente *cloud*, no entanto estes foram realizados no ambiente de *staging* com as mesmas configurações do ambiente de produção.

Uma vez que a capacidade das máquinas presentes no ambiente de *staging* não é equivalente às do ambiente *cloud*, não se conseguiu alcançar os números pretendidos. No entanto, tendo em conta os números atingidos, existe a suposição de que com a dimensão dos recursos da infra-estrutura actual esta terá muito melhor *performance* no ambiente de produção, uma vez que a necessidade de escalar dependerá do número de clientes e não da arquitectura projectada.

7.2.1 Cenário

Visualizar e analisar viagens Após realizada a autenticação no sistema, aceder à lista e viagens e seleccionar viagens para visualizar e analisar a informação disponibilizada em cada página.

Condições do teste

Tabela 7.1: Condições do teste de visualizar e analisar viagens

<i>Workload</i>	<i>Think time</i>	<i>Duração</i>
100 utilizadores concorrentes	3 a 4 segundos	30 minutos

Dos 100 utilizadores concorrentes, cerca de 20% destes, acedeu à página de comparação de trajectos.

7.2.2 Resultados

Numa primeira iteração deste cenário de teste, conseguiu-se obter uma resposta média de 30 pedidos por segundo. Uma vez que este número não satisfazia os atributos mínimos efectuou-se algum *tuning* na infra-estrutura.

1. **Replicação da base de dados *PostgreSQL* e *MongoDB*:** com esta modificação conseguiu obter-se um tempo médio de resposta de 50 pedidos por segundo. No entanto o *bottleneck* não se encontrava nas bases de dados, mas sim na aplicação web.
2. **Replicação da *Web Stack*:** implementou-se balanceamento de carga e replicou-se a aplicação web, tendo levado a um aumento do tempo médio de resposta para 100 pedidos por segundo.

Capítulo 8

Conclusão

Ponto da situação

As funcionalidades desenvolvidas durante este estágio, para o projecto Eco-Mobile na vertente B2C, encontram-se estáveis e prontas para serem colocadas em produção. Pretende-se que em meados do mês de Agosto de 2014, o projecto EcoMobile seja lançado na *cloud* como um produto, denominado de Drivian.

Relativamente à vertente B2B, o início do desenvolvimento desta ocorreu em meados de Maio de 2014, tendo esta versão do produto um período de desenvolvimento apontado para 6 a 9 meses, sendo expectável que ao final de 4 meses de desenvolvimento seja possível disponibilizar uma versão do produto a uma empresa piloto.

Experiência adquirida

A integração do estagiário neste projecto foi, sem margem para dúvidas, um enorme contributo para a sua formação como futuro profissional na área de Engenharia de Software. Uma vez que o estagiário foi integrado numa equipa de desenvolvimento, foi necessário um processo de adaptação com todos os elementos, ao projecto e aos processos e metodologias de desenvolvimento presentes na empresa. Este processo de adaptação correu da melhor forma, sendo que o trabalho em equipa e o contacto com profissionais com um grau de experiência profissional superior teve um grande impacto na aquisição de novas competências pelo estagiário, que apenas possuía experiência académica.

Para além do conjunto de competências ligadas à Engenharia de Software, como metodologias de desenvolvimento ágeis, análise de requisitos e gestão de tarefas, este estágio permitiu adquirir um conjunto de competências técnicas

fortemente relacionadas com desenvolvimento web, bases de dados relacionais e não relacionais e princípios arquitecturais de escalabilidade.

Em suma, a experiência e conhecimentos adquiridos ao longo do estágio tornaram o estagiário um profissional com maiores capacidades e melhor preparado para o mercado de trabalho.

Principais obstáculos

No início deste estágio existiu um período de indefinição, quanto ao âmbito concreto do estágio e instabilidade dos requisitos gerais do projecto, o que durou algumas semanas até à sua concreta definição.

Uma vez que o estagiário não possuía qualquer experiência ao nível da utilização de metodologias ágeis para o desenvolvimento de projectos, houve inicialmente algumas dificuldades na adaptação do processo, tendo esta dificuldade sido ultrapassada à medida que o estágio foi avançando.

Outra dificuldade sentida durante o decorrer do estágio encontra-se relacionada com o processo de elaboração de testes de software e de qualidade. Esta era uma área em que o estagiário, praticamente, não possuía qualquer conhecimento e experiência. Não se revelou possível, devido a questões logísticas e de planeamento, realizar um bom plano de testes de carga, que permitisse elaborar um dimensionamento concreto dos recursos da infra-estrutura na *cloud*.

Trabalho futuro

Pretende-se nos próximos tempos continuar a elaboração do módulo de *Reporting* do produto na vertente B2B, por forma a permitir a criação de factos e análises exploratórias da informação relativa a veículos e grupos. No final do desenvolvimento deste módulo, pretende-se incorporar o módulo de *Reporting* no produto Drivian, como uma funcionalidade *premium* da aplicação.

Apêndice A

Artefactos da metodologia ágil

Como referido na secção 2.1 aqui são apresentados os artefactos, relativos à metodologia ágil, produzidos durante a elaboração deste estágio. Aqui serão apresentados os *Backlogs* associados a cada uma das *sprints* e os respectivos *Burndown Charts*.

A.1 *Business-to-consumer*

Sprint 1

Item <i>Backlog</i>	Tarefas	Estado	<i>Story points</i>
ECOMOBILE-184	<i>Mongo Geospatial model for trips</i>	✓	3
ECOMOBILE-185	<i>Experiments for similarity algorithm implementation</i>	✓	5

Sprint 2

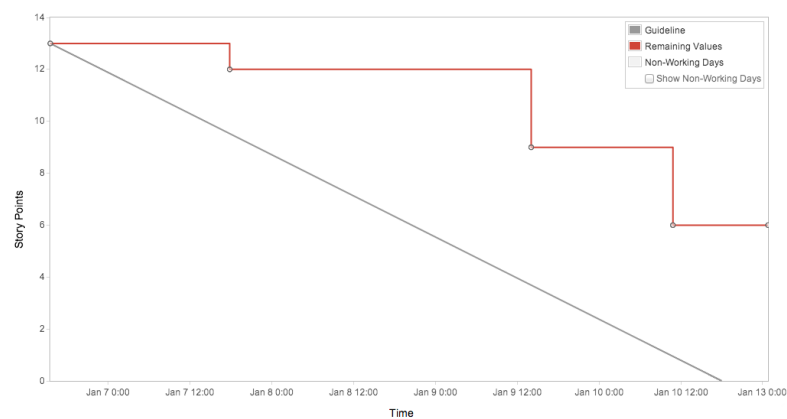
Item <i>Backlog</i>	Tarefas	Estado	<i>Story points</i>
ECOMOBILE-198	<i>API - Geospatial data</i>	✓	3
ECOMOBILE-199	<i>Geospatial worker</i>	✓	1
ECOMOBILE-200	<i>API - Get similar trips</i>	✓	3

Item <i>Backlog</i>	Tarefas	Estado	Story points
ECOMOBILE-201	<ul style="list-style-type: none"> -Redefine cluster algorithm -Similarity measure, distance measure and clustering validation -Redefine backend API for similar trips data -Front-end implementation 	X	6

Sprint 1 - Burndown



Sprint 2 - Burndown



Sprint 3

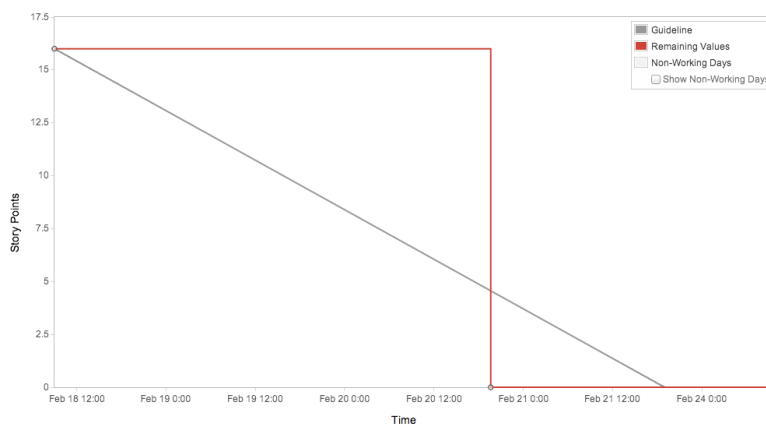
<i>Item Backlog</i>	<i>Tarefas</i>	<i>Estado</i>	<i>Story points</i>
ECOMOBILE-201	<ul style="list-style-type: none"> -Redefine cluster algorithm -Similarity measure, distance measure and clustering validation -Redefine backend API for similar trips data -Front-end implementation 	✓	6

Sprint 4

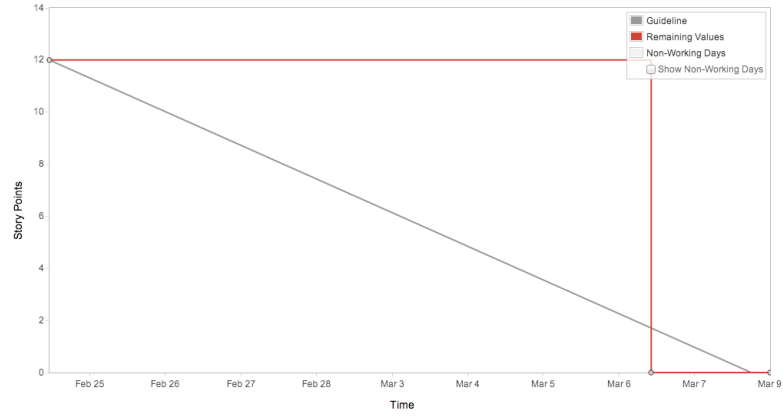
<i>Item Backlog</i>	<i>Tarefas</i>	<i>Estado</i>	<i>Story points</i>
ECOMOBILE-246	<ul style="list-style-type: none"> -Implementation of the recommendation route algorithm -Recommendation route API -Front-end page implementation 	✓	12

Sprint 5

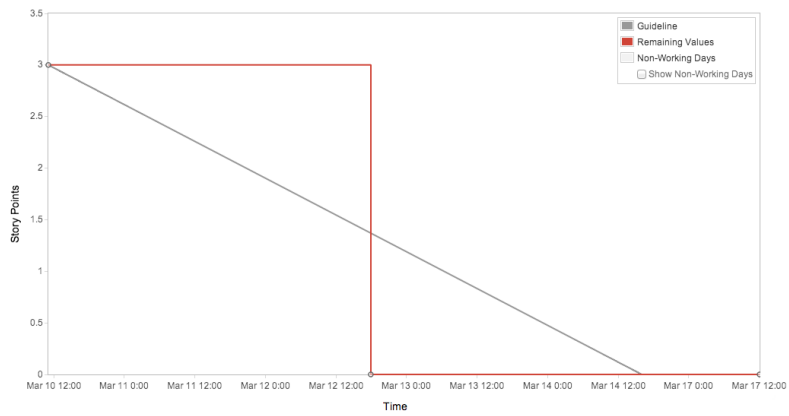
<i>Item Backlog</i>	<i>Tarefas</i>	<i>Estado</i>	<i>Story points</i>
ECOMOBILE-285	Select other trips to compare	✓	6

Sprint 3 - Burndown

Sprint 4 - Burndown



Sprint 5 - Burndown



A.2 *Business-to-business*

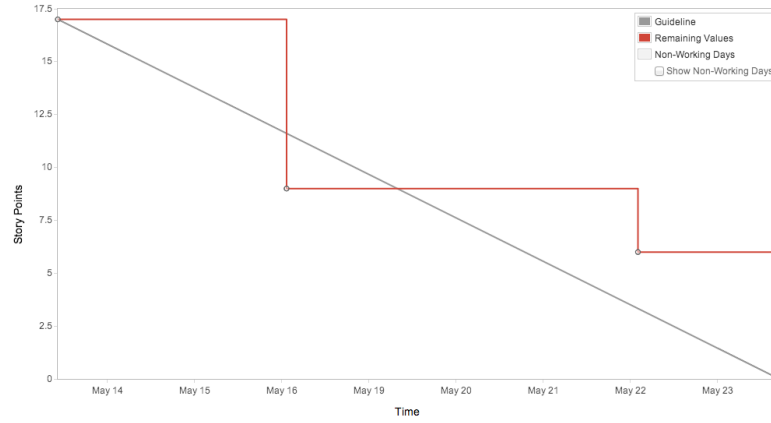
Sprint 1

<i>Item Backlog</i>	<i>Tarefas</i>	<i>Estado</i>	<i>Story points</i>
EB-16	<ul style="list-style-type: none"> -Implement backend and data model to store user metrics relative to the done trips -Implement backend workers to process and store user metrics/events relative to the done trips -Implement user API to provide the user profile information and global metrics -Implement frontend that shows the user profile information and global metrics 	✓	8
EB-17	Implement frontend that shows the user evolution along time	✓	3
EB-107	Create custom dashboards and reports	✗	6

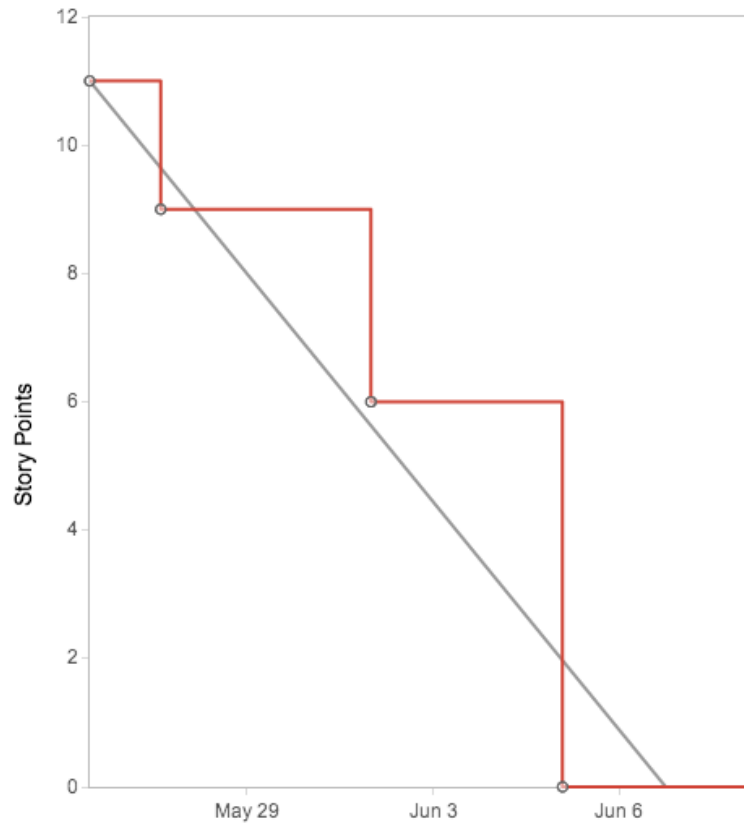
Sprint 2

<i>Item Backlog</i>	<i>Tarefas</i>	<i>Estado</i>	<i>Story points</i>
EB-21	Driver profile analytics	✓	6
EB-107	Create custom dashboards and reports	✓	2
EB-115	Filter current view by general timeframe	✓	3

Sprint 1 - Burndown



Sprint 2 - Burndown



Apêndice B

Análise de riscos

Nenhum projecto de Engenharia de Software está livre da ocorrência de imprevistos. Estes acontecimentos não previstos podem traduzir-se no incumprimento de um ou mais objectivos definidos para o projecto. Assim existiu um esforço para detectar atempadamente o acontecimento destes imprevistos.

De seguida serão identificados os riscos que poderiam ocorrer ao longo do projecto, e a sua ocorrência ou não ocorrência, bem como o seu impacto, probabilidade de ocorrência e as estratégias de mitigação para o risco.

Risco 01	Tipo Recurso	Impacto Crítico	Probabilidade Alta
Descrição Durante a janela temporal deste estágio a aquisição de um ambiente <i>cloud</i> para a elaboração de testes de carga reais pode não acontecer, o que não permitirá realizar testes de carga num ambiente totalmente real.			
Estratégia de mitigação Virtualizar o ambiente de <i>staging</i> e efectuar os teste de carga sobre esse ambiente, por forma a realizar alguns testes e pressupor que num ambiente <i>cloud</i> os resultados serão muito melhores.			
A ocorrência deste risco verificou-se e foi utilizada a estratégia de mitigação apresentada como forma de realizar testes de carga e conseguir uma percepção da dimensão mínima do sistema.			

Risco 02	Tipo Projecto	Impacto Crítico	Probabilidade Alta
Descrição A visão abstracta e a pouca estabilidade dos requisitos relacionados com a versão B2B, podem levar a uma redefinição dos requisitos previstos para a componente a desenvolver para esta versão neste estágio.			
Estratégia de mitigação Acompanhar durante as reuniões de projecto se existem alterações das necessidades de negócio da vertente B2B, por forma a adaptar os novos requisitos ao plano do estágio.			
A ocorrência deste risco verificou-se tendo sido utilizada a estratégia de mitigação apresentada como forma de minimizar o impacto deste risco no estágio.			

Risco 03	Tipo Projecto	Impacto Crítico	Probabilidade Alta
Descrição A disponibilidade da equipa de desenvolvimento para início da implementação da vertente B2B, aquando o planeado para o estágio, pode não ser total ou mesmo impossível.			
Estratégia de mitigação Reutilizar a solução tecnológica base da vertente B2C e adaptar para a vertente B2B por forma a garantir o cumprimento dos objectivos planeados para o estágio neste vertente.			
A ocorrência deste risco verificou-se tendo sido utilizada a estratégia de mitigação apresentada como forma de minimizar o impacto deste risco no estágio.			

Risco 04	Tipo Recurso	Impacto Médio	Probabilidade Alta
Descrição Durante a janela temporal deste estágio o lançamento do produto pode não acontecer, o que se traduz numa menor quantidade de dados sensoriais para efeitos de experimentação e validação dos algoritmos a implementar.			
Estratégia de mitigação Gerar dados sensoriais artificiais, ou multiplicar os dados sensoriais existentes, para conseguir realizar uma experimentação com qualidade.			
A ocorrência deste risco verificou-se, no entanto ao longo do estágio o impacto deste veio diminuindo, uma vez que conseguiu-se internamente gerar um número aceitável de viagens reais para efeitos de experimentação e testes.			

Risco 05	Tipo Técnico	Impacto Crítico	Probabilidade Baixa
Descrição Integração num projecto que possui um desenho arquitectural já implementado, pode afectar a integração da solução pretendida com este estágio.			
Estratégia de mitigação Fazer um estudo profundo das tecnologias arquitecturais utilizadas e manter o constante diálogo com o arquitecto da solução.			
A ocorrência deste risco não se verificou.			

Apêndice C

Análise de funcionalidades e custos

Documento de estudo de mercado: Sentilant-Estudo_Produtos_Concorrentes_v2.0.pdf

Apêndice D

Análise das necessidades de negócio

Documento de análise de requisitos de negócio: Sentilant-Necessidades_Negocio_v2.0.pdf

Apêndice E

Análise de requisitos

Para a priorização dos requisitos foram utilizadas as palavras-chave, definidas no RFC 2119, "*Must*", "*Should*", "*Could*" e "*Optional*".

E.1 Requisitos funcionais

A especificação dos requisitos funcionais a desenvolver foi feito através de *user stories*, utilizando o seguinte formato:

- **Enquanto** <actor do sistema> (Quem?)
- **Quero** <acção> (O quê?)
- **De forma a** <benefício> (Porquê?)

E.1.1 *Business-to-consumer*

<i>ID</i> B2C-01	Prioridade <i>MUST</i>
Enquanto utilizador web	
Quero escolher uma viagem realizada por mim	
De forma a poder aceder à componente de comparação de trajectos e visualizar qual é média da minha segurança ao longo dessa rota	

ID B2C-02	Prioridade <i>MUST</i>
Enquanto utilizador web	
Quero escolher uma viagem realizada por mim	
De forma a poder aceder à componente de comparação de trajectos e visualizar qual é média da minha economia ao longo dessa rota	

ID B2C-03	Prioridade <i>MUST</i>
Enquanto utilizador web	
Quero escolher uma viagem realizada por mim	
De forma a poder aceder à componente de comparação de trajectos e visualizar graficamente a evolução da minha segurança ao longo dessa rota	

ID B2C-04	Prioridade <i>MUST</i>
Enquanto utilizador web	
Quero escolher uma viagem realizada por mim	
De forma a poder aceder à componente de comparação de trajectos e visualizar graficamente a evolução da minha economia ao longo dessa rota	

ID B2C-05	Prioridade <i>MUST</i>
Enquanto utilizador web	
Quero escolher uma viagem realizada por mim	
De forma a poder aceder à componente de comparação de trajectos e comparar o trajecto dessa viagem, com o trajecto de viagens semelhantes realizadas por mim	

ID B2C-06	Prioridade <i>MUST</i>
Enquanto utilizador web	
Quero escolher uma viagem realizada por mim	
De forma a poder aceder à componente de comparação de trajectos e comparar o trajecto dessa viagem, com o trajecto de viagens semelhantes realizadas pelos meus amigos da aplicação	

ID B2C-07	Prioridade <i>MUST</i>
Enquanto utilizador web	
Quero escolher uma viagem realizada por mim	
De forma a poder aceder à componente de comparação de trajectos e comparar o trajecto dessa viagem, com o trajecto de viagens que possuem o mesmo inicio e fim, realizadas por mim	

ID B2C-08	Prioridade <i>MUST</i>
Enquanto utilizador web	
Quero escolher uma viagem realizada por mim	
De forma a poder aceder à componente de comparação de trajectos e comparar o trajecto dessa viagem, com o trajecto de viagens que possuem o mesmo inicio e fim, realizadas pelos meus amigos da aplicação	

ID B2C-09	Prioridade <i>MUST</i>
Enquanto utilizador web	
Quero escolher uma viagem realizada por mim	
De forma a poder aceder à componente de comparação de trajectos e escolher qual o indicador-chave a comparar globalmente entre dois trajectos	

ID B2C-10	Prioridade <i>MUST</i>
Enquanto utilizador web	
Quero escolher uma viagem realizada por mim	
De forma a poder aceder à componente de comparação de trajectos e escolher qual o indicador-chave a comparar entre dois trajectos, temporalmente ao longo dos trajectos	

ID B2C-11	Prioridade <i>MUST</i>
Enquanto utilizador web	
Quero escolher uma viagem realizada por mim	
De forma a poder aceder à componente de comparação de trajectos e visualizar no mapa a rota dos dois trajectos a comparar	

ID B2C-12	Prioridade <i>MUST</i>
Enquanto utilizador web	
Quero escolher uma viagem realizada por mim	
De forma a poder aceder à componente de comparação de trajectos e visualizar no mapa os eventos de condução dos dois trajectos a comparar	

ID B2C-13	Prioridade <i>SHOULD</i>
Enquanto utilizador web	
Quero escolher uma viagem realizada por mim	
De forma a poder aceder à componente de recomendação de trajectos e visualizar a rota do trajecto mais económico e respectiva duração, distância, economia e segurança média	

ID B2C-14	Prioridade <i>SHOULD</i>
<p>Enquanto utilizador web</p> <p>Quero escolher uma viagem realizada por mim</p> <p>De forma a poder aceder à componente de recomendação de trajectos e visualizar a rota do trajecto mais seguro e respectiva duração, distância, economia e segurança média</p>	

ID B2C-15	Prioridade <i>SHOULD</i>
<p>Enquanto utilizador web</p> <p>Quero escolher uma viagem realizada por mim</p> <p>De forma a poder aceder à componente de recomendação de trajectos e visualizar a rota do trajecto mais rápido e respectiva duração, distância, economia e segurança média</p>	

ID B2C-16	Prioridade <i>SHOULD</i>
<p>Enquanto utilizador web</p> <p>Quero escolher uma viagem realizada por mim</p> <p>De forma a poder aceder à componente de recomendação de trajectos e visualizar a rota do trajecto mais curto e respectiva duração, distância, economia e segurança média</p>	

E.1.2 *Business-to-business*

ID B2B-01	Prioridade <i>SHOULD</i>
<p>Enquanto gestor operacional</p> <p>Quero aceder ao componente de <i>reporting</i></p> <p>De forma a poder criar análises personalizadas, através do <i>drag-and-drop</i> das dimensões e indicadores disponíveis para a minha organização</p>	

ID B2B-02	Prioridade <i>MUST</i>
Enquanto gestor operacional	
Quero aceder ao componente de <i>reporting</i>	
De forma a poder pesquisar as análises, através do seu título, que foram previamente criadas ou pré-definidas pela aplicação	

ID B2B-03	Prioridade <i>MUST</i>
Enquanto gestor operacional	
Quero aceder ao componente de <i>reporting</i>	
De forma a poder criar uma vista que contém diversas análises previamente criadas ou pré-definidas pela aplicação, através do <i>drag-and-drop</i> das análises para a vista actual	

ID B2B-04	Prioridade <i>MUST</i>
Enquanto gestor operacional	
Quero aceder ao componente de <i>reporting</i>	
De forma a poder filtrar temporalmente, pelo último semestre, trimestre, mês ou semana, todas as análises que se encontram na vista actual	

ID B2B-05	Prioridade <i>MUST</i>
Enquanto gestor operacional	
Quero aceder ao componente de <i>reporting</i>	
De forma a poder fazer o <i>drill-down</i> e <i>drill-up</i> , numa das análises da vista actual, por forma a visualizar a informação com níveis diferentes de granularidade	

ID B2B-06	Prioridade <i>MUST</i>
Enquanto gestor operacional	
Quero aceder ao componente de <i>reporting</i>	
De forma a poder escolher o tipo de agrupamento dos dados a fazer, aquando o <i>drill-down</i> , numa das análises da vista actual	

ID B2B-07	Prioridade <i>MUST</i>
Enquanto gestor operacional	
Quero aceder ao componente de <i>reporting</i>	
De forma a poder remover uma das análises da vista actual	

ID B2B-08	Prioridade <i>SHOULD</i>
Enquanto gestor operacional	
Quero aceder ao componente de <i>reporting</i>	
De forma a poder alterar o tipo de vista, das análises que se encontram na vista actual, para um gráfico ou tabela	

ID B2B-09	Prioridade <i>SHOULD</i>
Enquanto gestor operacional	
Quero aceder ao componente de <i>reporting</i>	
De forma a poder exportar uma das análises da vista actual para o formato de PDF ou CSV	

ID B2B-10	Prioridade <i>SHOULD</i>
Enquanto gestor operacional	
Quero aceder ao componente de <i>reporting</i>	
De forma a poder filtrar os indicadores das análises da vista actual por um determinado conjunto de condutores	

ID B2B-11	Prioridade <i>SHOULD</i>
Enquanto gestor operacional	
Quero aceder ao componente de <i>reporting</i>	
De forma a poder filtrar os indicadores das análises da vista actual por um determinado conjunto de veículos	

ID B2B-12	Prioridade <i>SHOULD</i>
Enquanto gestor operacional	
Quero aceder ao componente de <i>reporting</i>	
De forma a poder filtrar os indicadores das análises da vista actual por um determinado conjunto de grupos da organização	

ID B2B-13	Prioridade <i>SHOULD</i>
Enquanto gestor operacional	
Quero aceder ao componente de <i>reporting</i>	
De forma a poder exportar a vista actual para o formato PDF	

ID B2B-14	Prioridade <i>SHOULD</i>
Enquanto gestor operacional	
Quero aceder ao componente de <i>reporting</i>	
De forma a poder criar uma vista personalizada e guardar como um <i>dashboard</i> , para poder de forma periódica visualizar a informação	

ID B2B-15	Prioridade <i>SHOULD</i>
Enquanto gestor operacional	
Quero aceder ao componente de <i>reporting</i>	
De forma a poder criar uma vista personalizada e guardar como um relatório, para poder partilhar com outros elementos da organização	

ID B2B-16	Prioridade <i>MUST</i>
Enquanto gestor operacional	
Quero aceder ao componente de <i>reporting</i>	
De forma a poder limpar a vista actual, caso possua análises seleccionadas	

ID B2B-17	Prioridade <i>SHOULD</i>
Enquanto gestor operacional	
Quero aceder ao componente de <i>reporting</i>	
De forma a poder aceder à lista de <i>dashboards</i> previamente criados e seleccionar um <i>dashboard</i> para analisar a informação presente nele	

ID B2B-18	Prioridade <i>SHOULD</i>
Enquanto gestor operacional	
Quero aceder ao componente de <i>reporting</i>	
De forma a poder aceder à lista de relatórios previamente criados e seleccionar um relatório para analisar ou exportar	

Apêndice F

Modelo de dados

F.1 Classificação e recomendação

F.1.1 Colecção *trips_geospatial*

trips_geospatial
trip_id: BIGINT [PK]
cluster_id: BIGINT start_coordinates: STRUCT end_coordinates: STRUCT global_metrics: STRUCT positions: ARRAY

Figura F.1: Modelo de dados da colecção *trips_geospatial*

Tabela F.1: Modelo de dados *trips_geospatial*

Campo	Tipo dados	Índice	Descrição
trip_id	BIGINT	<i>Single field</i>	Identifica a viagem.
cluster_id	BIGINT	<i>Single field</i>	Identifica o <i>cluster</i> onde se encontra a viagem.
start_coordinates	GeoJSON	<i>Geospatial</i>	Coordenadas GPS de início da viagem.

Campo	Tipo dados	Índice	Descrição
end_coordinates	GeoJSON	<i>Geospatial</i>	Coordenadas GPS do final da viagem.
global_metrics	Documento	Não	Indicadores-chave globais da viagem.
positions	Array de documentos	Não	Posições GPS do trajecto agrupadas.

F.1.2 Colecção *trips_clusters*

trips_clusters
cluster_id: BIGINT [PK]
start_coordinates: STRUCT
end_coordinates: STRUCT
trips: ARRAY
cluster_measures: STRUCT
best_routes: STRUCT

Figura F.2: Modelo de dados da colecção *trips_clusters*

Tabela F.2: Modelo de dados *trips_clusters*

Campo	Tipo dados	Índice	Descrição
cluster_id	BIGINT	<i>Single field</i>	Identifica a o cluster.
start_coordinates	GeoJSON	<i>Geospatial</i>	Coordenadas GPS de início das viagens do <i>cluster</i> .
end_coordinates	GeoJSON	<i>Geospatial</i>	Coordenadas GPS do final das viagens do <i>cluster</i> .
trips	Array de documentos	Não	Contém os identificadores das viagens e respectivos indicadores-chave.

Campo	Tipo dados	Índice	Descrição
clus- ter_measures	Documento	Não	Indicadores-chave globais das viagens do <i>cluster</i> .
best_routes	Documento	Não	Contém os identificadores dos melhores trajectos em termos de segurança, economia, distância e tempo.

F.2 Reporting

F.2.1 Colecção *user_daily_metrics*

Esta colecção é responsável pela representação e armazenamento dos agregados relativos às viagens realizadas por um condutor. De forma a dar ao leitor uma melhor percepção do modelo de dados é de seguida demonstrado um documento, que representa os factos das viagens de um condutor num determinado dia.

```

    "_id" : "2014-12-30/8",
    "meta" : {
      "date" : "2014-12-30",
      "day" : 30,
      "month" : 12,
      "year" : 2014,
      "semester" : 2,
      "quarter" : 4,
      "week_of_month" : 4,
      "org_id" : 5,
      "org_name" : "Sentilant",
      "user_id" : 3,
      "user_email" : "irenedavis@sentilant.com",
    },
    "metrics" : {
      "ac" : 12,
      "hb" : 15,
      "hc" : 9,
      "total_trips" : 15,
      "safety" : 76,
      "economy" : 82,
      "time_trav" : 8988.8,
      "dist_trav" : 245.4,
      "productivity" : 70,
      "speed" : 60
    }
  }

```

No sub-documento *meta* é onde se encontram representadas as dimensões descritas de seguida:

Tabela F.3: Dimensões do modelo de dados *user_daily_metrics*

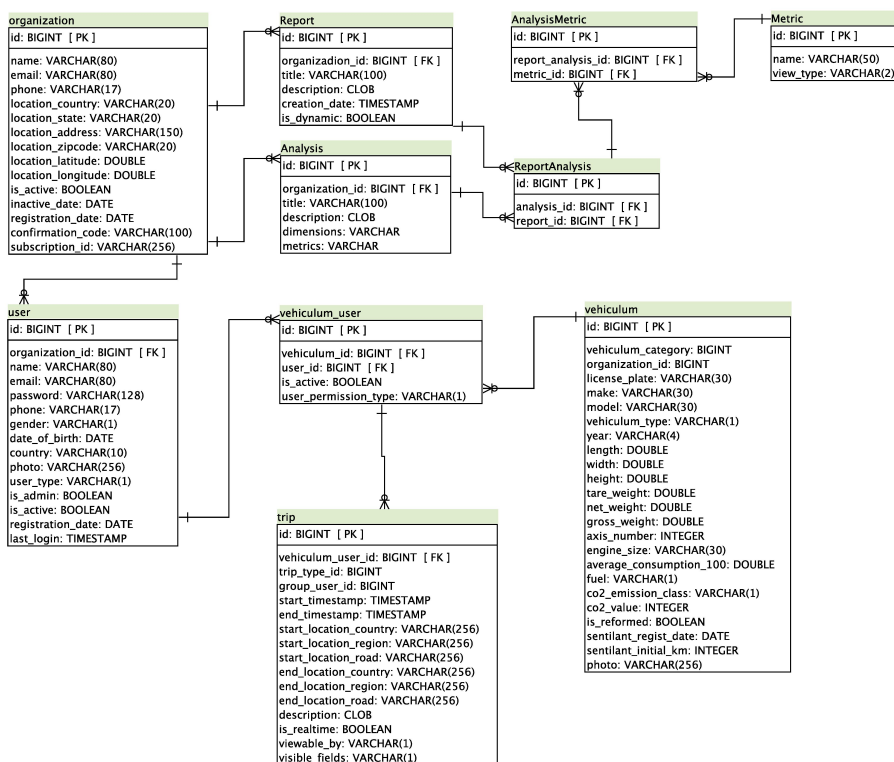
Campo	Dimensão	Índice	Descrição
date	Tempo	Não	Identifica a data completa.
day	Tempo	Não	Identifica o dia.
month	Tempo	Não	Identifica o mês.
year	Tempo	Não	Identifica o ano.
semester	Tempo	Não	Identifica o semestre.
quarter	Tempo	Não	Identifica o trimestre.
week_of_month	Tempo	Não	Identifica a semana do mês.
org_id	Organização	<i>Single field</i>	Identificador da organização.
org_name	Organização	Não	Nome da organização.
user_id	Utilizador	<i>Single field</i>	Identificador do utilizador.
user_email	Utilizador	<i>Single field</i>	Email do utilizador.

No sub-documento *metrics* é onde se encontram representados os factos descritos de seguida:

Tabela F.4: Descrição dos factos pré-calculados respeitantes às viagens de um condutor

Campo	Descrição
safety	Segurança média das viagens do condutor no dia.
economy	Economia média das viagens do condutor no dia.
speed	Velocidade média das viagens do condutor no dia.

Campo	Descrição
productivity	Productividade média das viagens do condutor no dia.
time_trav	Tempo total em andamento do condutor no dia.
dist_trav	Distância percorrida pelo condutor no dia.
total_trips	Total de viagens realizadas pelo condutor no dia.
hb	Total de travagens bruscas do condutor no dia.
ac	Total de acelerações bruscas do condutor no dia.
hc	Total de curvas bruscas do condutor no dia.

F.2.2 Modelo *reporting* B2BFigura F.3: Modelo de dados da componente de *reporting* da versão B2BTabela F.5: Descrição das entidades do modelo da componente de *reporting*

Entidade	Descrição
Organization	Representa uma organização.
User	Representa um utilizador da aplicação que pode ser um administrador ou um condutor.
Vehiculum	Representa um veículo da organização.
Trip	Representa uma viagem realizada por um determinado condutor num determinado veículo.

Entidade	Descrição
Analysis	Representa as análises personalizadas sobre os indicadores criadas para a organização.
Report	Representa os relatórios que foram criados para a organização.
Metric	Representa as métricas que são disponibilizadas pela aplicação ou criadas pela organização.

Todo o modelo de dados da versão B2B foi desenhado pelo estagiário. No entanto, uma vez que este possui uma grande complexidade, foi colocado aqui apenas a parte directamente relacionada com o trabalho realizado neste estágio. Na tabela anterior encontra-se a descrição das entidades do modelo de dados, à excepção das entidades fracas.

Apêndice G

Trabalho desenvolvido

G.1 Algoritmo de recomendação

Algorithm 3 Algoritmo de recomendação de trajectos

Pré-condições: *trajecto* - Coordenadas GPS da rota

Pós-condições: Retorna lista melhores trajectos

```

function RECOMENDA(trajecto)
  raio_procura ← 400 metros
  len_trajecto ← tamanho(trajecto)
  coordenadas_iniciais ← trajecto[0]
  coordenadas_finais ← trajecto[len_trajecto]
  candidatos ← obter_clusters_candidatos(coordenadas_iniciais,
coordenadas_finais, raio_procura)
  melhores_trajectos ← ∅
  actual_economia_trajecto ← ∅
  actual_seguranca_trajecto ← ∅
  actual_distancia_trajecto ← ∅
  actual_duracao_trajecto ← ∅
  for candidato in candidatos do
    economia_melhor_trajecto ← obter_economia(candidato)
    seguranca_melhor_trajecto ← obter_seguranca(candidato)
    distancia_melhor_trajecto ← obter_distancia(candidato)
    duracao_melhor_trajecto ← obter_duracao(candidato)
    if economia_melhor_trajecto > actual_economia_trajecto then
      actual_economia_trajecto ← economia_melhor_trajecto
      actualiza_melhores_trajectos(candidato, economia_melhor_trajecto)
    end if
    if seguranca_melhor_trajecto > actual_seguranca_trajecto then
      actual_seguranca_trajecto ← seguranca_melhor_trajecto
      actualiza_melhores_trajectos(candidato, seguranca_melhor_trajecto)
    end if
    if distancia_melhor_trajecto > actual_distancia_trajecto then
      actual_distancia_trajecto ← distancia_melhor_trajecto
      actualiza_melhores_trajectos(candidato, distancia_melhor_trajecto)
    end if
    if duracao_melhor_trajecto > actual_duracao_trajecto then
      actual_duracao_trajecto ← duracao_melhor_trajecto
      actualiza_melhores_trajectos(candidato, duracao_melhor_trajecto)
    end if
  end for
  return melhores_trajectos
end function

```

Apêndice H

Testes e validação

H.1 Validação *clusters*

Tabela H.1: Avaliação da coesão e separação dos *clusters*

<i>Cluster id</i>	Total viagens	Coesão	Separação
1	2	0,1	0,96
2	2	0,1	0,96
3	4	0,12	0,95
4	6	0,1	0,95
5	4	0,13	0,93
6	2	0	0,96
7	2	0	0,98
8	2	0,2	0,96
9	2	0,1	0,95
10	3	0,17	0,94
11	2	0	0,98

<i>Cluster id</i>	Total viagens	Coesão	Separação
12	14	0,12	0,94
13	2	0,2	0,94
14	2	0,1	0,95
15	3	0,07	0,94
16	4	0,1	0,97
17	2	0,1	0,95
18	2	0,1	0,94
19	2	0,1	0,98
20	2	0,1	0,97
21	5	0	0,94
22	4	0,08	0,97
23	2	0	0,97
24	2	0	0,98
25	2	0,1	0,94
26	2	0,2	0,95
27	2	0	0,98
28	3	0	0,98
29	2	0,2	0,96
30	2	0,2	0,98
31	2	0,2	0,93
32	5	0,13	0,96

<i>Cluster id</i>	Total viagens	Coesão	Separação
33	2	0,2	0,98
34	35	0	0,96
35	17	0,12	0,94
36	2	0,2	0,96
37	2	0,1	0,97
38	2	0,2	0,96
39	2	0,2	0,98
40	2	0,1	0,97
41	11	0	0,98
42	2	0	0,94
43	2	0,2	0,96
44	2	0	0,94
45	2	0,2	0,97
46	2	0	0,97
47	3	0	0,98
48	6	0	0,98
49	12	0,17	0,94
50	7	0,08	0,95
51	382	0	0,94
52	2	0,1	0,94

H.2 Testes de aceitação das funcionalidades *business-to-consumer*

Tabela H.2: Conjunto de testes de aceitação da vertente B2C

Requisito	Descrição do teste	Validação
B2C-01	Seleccionar um trajecto e visualizar a segurança média ao longo das viagens da rota	✓
B2C-02	Seleccionar um trajecto e visualizar a economia média ao longo das viagens da rota	✓
B2C-03	Seleccionar a métrica de segurança e visualizar graficamente a evolução ao longo das viagens da rota	✓
B2C-04	Seleccionar a métrica de economia e visualizar graficamente a evolução ao longo das viagens da rota	✓
B2C-05	Aplicar um filtro por viagens semelhantes e seleccionar um dos trajectos disponíveis	✓
B2C-06	Aplicar um filtro por viagens semelhantes dos meus amigos e seleccionar um dos trajectos disponíveis	✓
B2C-07	Aplicar um filtro por viagens com o mesmo início e fim e seleccionar um dos trajectos disponíveis	✓
B2C-08	Aplicar um filtro por viagens, dos meus amigos, com o mesmo início e fim e seleccionar um dos trajectos disponíveis	✓
B2C-09	Seleccionar um dos indicadores-chave disponíveis e actualizar o gráfico de comparação global	✓
B2C-10	Seleccionar um dos indicadores-chave disponíveis e actualizar o gráfico de comparação ao longo do trajecto	✓

Requisito	Descrição do teste	Validação
B2C-11	Seleccionar um trajecto para comparar e visualizar os trajectos no mapa	✓
B2C-12	Seleccionar no mapa os eventos de condução a visualizar	✓
B2C-13	Aceder a recomendação de trajecto e visualizar a rota do trajecto mais económico, caso exista	✓
B2C-14	Aceder a recomendação de trajecto e visualizar a rota do trajecto mais seguro, caso exista	✓
B2C-15	Aceder a recomendação de trajecto e visualizar a rota do trajecto mais rápido, caso exista	✓
B2C-16	Aceder a recomendação de trajecto e visualizar a rota do trajecto mais curto, caso exista	✓

H.3 Testes de aceitação das funcionalidades *business-to-business*

Tabela H.3: Conjunto de testes de aceitação da vertente B2B

Requisito	Descrição do teste	Validação
B2B-01	Seleccionar os factos e dimensões disponibilizados para criar uma nova análise	✗ ¹
B2B-02	Pesquisar por uma análise na lista de análises disponíveis	✓
B2B-03	Arrastar uma análise para um <i>widget</i> e disponibilizar a informação da análise	✓

¹Requisito de baixa prioridade durante o decorrer do estágio.

Requisito	Descrição do teste	Validação
B2B-04	Aplicar um filtro temporal global (último semestre, último trimestre, último mês, mês actual, semana actual) e actualizar todos os <i>widgets</i>	✓
B2B-05	Fazer <i>drill-down</i> e <i>drill-up</i> numa das análises	✓
B2B-06	Escolher qual o tipo de agrupamento temporal se deseja agregar, aquando o <i>drill-down</i>	✓
B2B-07	Seleccionar a opção "Limpar" num <i>widget</i> e remover a análise	✓
B2B-08	Seleccionar a opção "Mudar vista" num <i>widget</i> e alterar o tipo de vista sobre a análise	✗ ¹
B2B-09	Seleccionar a opção "Exportar" num <i>widget</i> e exportar a análise para PDF/CSV	✗ ¹
B2B-10	Escolher os condutores que se desejam e actualizar a vista filtrada pelos condutores	✓
B2B-11	Escolher os veículos que se desejam e actualizar a vista filtrada pelos condutores	✗ ²
B2B-12	Escolher os grupos que se desejam e actualizar a vista filtrada pelos grupos	✗ ²
B2B-13	Seleccionar a opção de "Exportar para PDF" e exportar a actual vista para PDF	✗ ¹
B2B-14	Seleccionar a opção de "Guardar como <i>dashboard</i> " e guardar a actual vista como um <i>dashboard</i>	✗ ²
B2B-15	Seleccionar a opção de "Guardar como relatório" e guardar a actual vista como um relatório	✗ ²
B2B-16	Seleccionar a opção de "Limpar" e remover todas as análises da actual vista	✓

²Requisito não implementado devido a alterações no planeamento.

H.3. TESTES DE ACEITAÇÃO DAS FUNCIONALIDADES BUSINESS-TO-BUSINESS139

Requisito	Descrição do teste	Validação
B2B-17	Seleccionar a opção "Dashboards guardados" e visualizar a lista de <i>dashboards</i>	X ²
B2B-18	Seleccionar a opção "Relatórios guardados" e visualizar a lista de relatórios	X ²

Bibliografia

- [1] F. Williams Engdahi. Behind Oil Price Rise: Peak Oil or Wall Street Speculation?
- [2] IMTT. Manual de eco-condução. <http://goo.gl/fdnYne>. Acedido em 11 de Janeiro de 2014.
- [3] Roger S. Pressman. Software Engineering - A Practitioner's Approach.
- [4] Rui Chicória. Real-time architecture and workflow for the One.Stop.Transport platform.
- [5] Mike Cohn. What is planning poker. <http://goo.gl/TgHygm>. Acedido em 19 de Janeiro de 2014.
- [6] Pair Programming. Pair programming. <http://goo.gl/FzahVf>.
- [7] David J. Anderson. Kanban - successful evolutionary change for your technology business. <http://goo.gl/DwLPgb>. Acedido a 24 de Junho de 2014.
- [8] Git branching - basic branching and merging. <http://goo.gl/nFUkoa>. Acedido em 19 de Junho de 2014.
- [9] J.P. Shim e Merril Warkentin e James F. Courtney e Daniel J. Power e Ramesh Sharda e Christer Carlsson. Past, present, and future of decision support technology.
- [10] D. J. Power. A brief history of decision support systems.
- [11] R. Kimball. The data warehouse toolkit.
- [12] R. Kimball e Joe Caserta. The data warehouse etl toolkit.
- [13] Colin White. The next generation of business intelligence: Operational bi.

- [14] Joseph Guerra e David Andrews. Why you need a data warehouse.
- [15] Thomas H. Davenport. Competing on analytics.
- [16] Jeffrey Dean e Sanjay Ghemawat. Mapreduce: Simplified data processing on large clusters.
- [17] Suresh Anthony Ashish Thusoo, Zheng Shao.
- [18] Usame Fayyad e Gregory Piatetsky-Shapiro e Padhraic Smyth. From data mining to knowledge discovery in databases.
- [19] Ian H. Witten e Eibe Frank e Mark A. Hall. Data mining: Practical machine learning tools and techniques.
- [20] Rüdiger Wirth. Crisp-dm: Towards a standard process model for data mining.
- [21] University of California at Berkeley. How much information? <http://goo.gl/01wgXY>.
- [22] M. Anderberg. Cluster analysis for applications.
- [23] Anand Rajaraman e Jeff Ullman. Mining of massive datasets.
- [24] Rui Xu e Donald Wunsch. Survey of clustering algorithms.
- [25] A. Jain e R. Dubes. Algorithms for clustering data.
- [26] Joseph M. Hellerstein e Michael Stonebraker e James Hamilton. Architecture of a database system.
- [27] Rick Cattell. Scalable sql and nosql data stores.
- [28] Seth Gilbert e Nancy A. Lynch. Perspectives on the cap theorem.
- [29] High Scalability. Facebook's new realtime analytics system: Hbase to process 20 billion events per day. <http://goo.gl/4YCLzG>. Acedido em 12 de Junho de 2014.
- [30] Twitter. Realtime analytics with storm and hadoop. <http://goo.gl/5R1s9L>. Acedido em 12 de Junho de 2014.
- [31] What's the Big Data. Big data analytics and data science at netflix. <http://zite.to/ZXTbiI>. Acedido em 12 de Junho de 2014.
- [32] Joseph O. Chan. An architecture for big data analytics.

- [33] D.W Cearley. Cloud computing: key initiative overview.
- [34] Zaigham Mahmood e Richard Hill. Cloud computing for enterprise architectures.
- [35] J.M Willis. Cloud computing and the enterprise.
- [36] Gartner. It glossary: Multi-tenancy. <http://goo.gl/JC7NMw>. Acedido a 12 de Junho de 2014.
- [37] Qi Zhang e Lu Cheng e Raouf Boutaba. Cloud computing: state-of-the-art and research challenges.
- [38] TomTom. TomTom. <http://goo.gl/cLYVo6>. Acedido em 11 de Janeiro de 2014.
- [39] TomTom Business. TomTom Business - Gestão de Frotas. <http://goo.gl/wb2F91>. Acedido em 11 de Janeiro de 2014.
- [40] TomTom Business. TomTom WEBFLEET. <http://goo.gl/GMvXZC>. Acedido em 11 de Janeiro de 2014.
- [41] Fleetmatics. GPS Fleet Tracking. <http://goo.gl/ZnTZq1>. Acedido em 11 de Janeiro de 2014.
- [42] Frotcom Lusitana. Soluções de localização por GPS. <http://goo.gl/jEgvR1>. Acedido em 11 de Janeiro de 2014.
- [43] Inc. Automatic Labs. Automatic. <http://www.automatic.com/>. Acedido em 11 de Janeiro de 2014.
- [44] Slash Idea s.r.o. MyCarTracks. <http://www.mycartracks.com/>. Acedido em 11 de Janeiro de 2014.
- [45] Mike Cohn. User Stories. <http://goo.gl/KvKHR9>. Acedido em 19 de Janeiro de 2014.
- [46] Mike Cohn. Advantages of the “As a user, I want” user story template. <http://goo.gl/5msjSu>. Acedido em 19 de Janeiro de 2014.
- [47] Martin L. Abbott and Michael T. Fisher. Scalability Rules - 50 Principles for Scaling Web Sites.
- [48] Ping Yan e Daniel D. Zeng. Spatial movement pattern discovery with lcs based path similarity measure.

- [49] Ruoming Jin. Cluster validation. <http://goo.gl/RpnD1N>. Acedido em 25 de Junho de 2014.
- [50] Michael S. Rosenberg. Sequence alignment - concepts and history.
- [51] Stephen F. Altschul. Global and local sequence alignment.
- [52] R. W. Wibowo. Pyssa. <http://goo.gl/51i7uz>.
- [53] MongoDB Inc. Pre-aggregated reports. <http://goo.gl/2HI1Q1>. Acedido em 24 de Junho de 2014.
- [54] MongoDB Inc. Aggregation framework introduction. <http://goo.gl/CcrLA7>.
- [55] Vipin Kumar. Introduction to data mining.
- [56] Young Jung, In-Chul e S. Know. A Sequence Pattern Matching Approach to Shopping Path Clustering.
- [57] Markus Liu, Hechen e Schneider. Similarity Measurement of Moving Object Trajectories.