

Mestrado em Engenharia Informática
Estágio
Relatório Final

Distribuição Contextualizada de conteúdos

Nuno Rafael Lopes Correia
nrlc@student.dei.uc.pt

Orientadores:
Professor Doutor Carlos Lisboa Bento
Eng. Miguel Machado

01 de Julho de 2013



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

Agradecimentos

Queria agradecer a todos os que me acompanharam durante este projecto. Uma palavra especial de apreço à minha família que sempre me apoiou e também aos meus orientadores que foram uma pedra basilar para o bom rumo deste projecto. Da mesma forma gostaria de agradecer a toda a equipa que contribuiu para o meu desenvolvimento pessoal e profissional aumentando em grande escala a minha gama de saberes. É também de realçar o apoio dado pelos meus amigos que nunca me deixaram desmotivar e se colocaram sempre ao dispor.

Histórico de revisões

Data	Alteração	Versão
01-10-2012	Escrita do capítulo 'Estado da Arte'	0.1.0
05-10-2012	Adição do planeamento	0.2.0
24-10-2012	Escrita do capítulo 'Introdução'	0.3.0
08-11-2012	Adição da metodologia	0.4.0
19-11-2012	Escrita do capítulo de requisitos	0.5.0
20-12-2012	Escrita do capítulo da arquitectura	0.6.0
02-01-2013	Escrita do capítulo 'Solução'	0.7.0
08-01-2013	Adição dos riscos	0.8.0
14-01-2013	Escrita do capítulo 'Avaliação'	0.9.0
16-01-2013	Escrita do capítulo 'Notas Finais'	0.10.0
16-01-2013	Adição dos anexos	0.11.0
22-01-2013	Escrita do resumo	0.12.0
28-01-2013	Entrega do relatório intermédio	1.0.0
17-06-2013	Adição do capítulo de validação	1.1.0
19-06-2013	Adições no capítulo da solução	1.2.0
23-06-2013	Alteração da estrutura	1.3.0
25-06-2013	Adições no capítulo da solução	1.4.0
01-07-2013	Adição do histórico de riscos	1.5.0
03-07-2013	Entrega do relatório final	2.0.0

A versão deste documento segue o formato X.Y.Z (Entrega.Adição.Revisão). A alteração de conteúdo existente implica o incremento da casa Z e não consta do histórico de revisões; a adição de uma secção impõe o incremento da casa Y; a entrega do documento para avaliação (fase intermédia e final) resulta no incremento da casa X. A versão actual é a 2.0.0. Este versionamento segue os critérios sugeridos por Tom Preston-Werner, co-fundador do github, conforme mencionado em semver.org

Tabela 1: Histórico de revisões

Reparos do júri

Reparo	Tratamento
Data sets e validação	Ver capítulo 6
Detalhes filtragem colaborativa	Ver capítulo 3
Técnicas estatísticas e métricas a utilizar	Ver capítulo 6
Dificuldade em perceber user stories	Modificação na estrutura do relatório e melhor organização dos requisitos

Esta tabela pretende listar os principais reparos do júri durante a defesa intermédia e o seu tratamento

Tabela 2: Reparos do júri

Resumo

Este estágio está enquadrado no projecto TICE.Mobilidade, um projecto no qual vinte e oito co-promotores e alguns organismos de investigação, em conjunto com o Instituto Pedro Nunes, unem esforços para mudar o paradigma da mobilidade urbana em Portugal.

Este projecto, TICE.mobilidade, alinha-se a nível de objetivos com projectos como o SMART em Singapura ou o projecto 'Ville, Mobilite, Energie' em Nice, França. O que se pretende com este estágio é desenvolver um sistema de distribuição de conteúdos que seja sensível ao contexto em que está inserido, para que possam ser disponibilizados conteúdos em ecrãs presentes nos transportes públicos. Estes conteúdos irão ser recomendados usando variáveis contextuais como a localização, o contexto temporal e o perfil do utilizador que está a visualizar a informação. Estes conteúdos podem ser notícias, eventos, publicidade, ou aplicações para mobilidade.

Apesar do foco do projecto se limitar à área dos transportes públicos, o resultado do estágio trará um produto com uma aplicabilidade mais extensa e poderá ser usado em contextos mais amplos, quer móveis (ex. smartphones) ou mesmo em situações sem mudança no contexto geográfico (ex. Placares de publicidade).

Palavras-chave: Mobilidade, TICE, BUSCA, Conteúdos, Contexto, sistemas de recomendação, CARS.

Conteúdo

1	Introdução	3
1.1	Enquadramento	3
1.2	Objectivos	3
1.3	Motivação	4
2	Estado da arte	5
2.1	Sistemas sensíveis ao contexto	5
2.2	Sistemas de recomendação	8
2.2.1	Filtragem Colaborativa	10
2.2.2	Filtragem baseada em conteúdo	10
2.2.3	Soluções híbridas	11
2.2.4	Abordagens contextuais	11
2.2.5	Conclusões	12
3	Solução proposta	15
3.1	Definição do problema	15
3.2	Advertiser dashboard	16
3.3	API	16
3.4	Sistema de permissões	19
3.5	Sistema de recomendação	19
3.5.1	Perfis de utilizador	20
3.5.2	Medição de semelhança entre perfis	20
3.5.3	Cálculo das previsões	20
4	Definição de requisitos	23
4.1	Actores	23
4.2	Requisitos	24
4.2.1	Advertiser Dashboard	24
4.2.2	API	26
4.2.3	Sistema	29

5	Arquitectura e design	31
5.1	Vistas da arquitectura	31
5.1.1	Vista física	31
5.1.2	Vista lógica	32
5.2	Tecnologias	34
6	Validação	35
6.1	Estratégia de desenvolvimento	35
6.2	Testes	35
6.2.1	Validação cruzada	36
7	Abordagem	39
7.1	Metodologia de Trabalho	39
7.1.1	Ferramentas adicionais	40
7.2	Plano de trabalhos	41
7.3	Análise de Riscos	47
7.3.1	Riscos do projecto	48
8	Conclusões	51
8.1	Trabalho realizado	51
8.2	Contribuições	54
8.3	Obstáculos	54
8.4	Trabalho futuro	55
A	Prototipagem	57
B	Monitorização de riscos	61

Lista de Acrónimos

AMPPQ	<i>Advanced Message Queuing Protocol</i>
API	<i>Application Programming Interface</i>
BDD	<i>Behaviour driven development</i>
BUSCA	<i>BUS Context Awareness</i>
CARS	<i>Context Aware Recommender System</i>
ELBA	<i>European Location Based Advertising</i>
GIS	<i>Geographic Information System</i>
IETF	<i>Internet Engineering Task Force</i>
IPN	<i>Instituto Pedro Nunes</i>
MAE	<i>Mean Absolute Error</i>
MSE	<i>Mean Squared Error</i>
OST	<i>One Stop Transport</i>
POA	<i>Point of Advertisement</i>
REST	<i>Representational State Transfer</i>
RMSE	<i>Root Mean Squared Error</i>
RFC	<i>Request for Comments</i>
SMART	<i>Singapore mit alliance for research and technology</i>
TDD	<i>Test driven development</i>
TICE	<i>Tecnologias de Informação, Comunicação e Electrónica</i>
TBQ	<i>Taxonomy-based questionnaire</i>

Tabela 3: Acrónimos

CONTEÚDO

Lista de Tabelas

1	Histórico de revisões	
2	Reparos do júri	
3	Acrónimos	
2.1	Comparação entre sistemas de recomendação	11
7.1	Riscos	49

LISTA DE TABELAS

Lista de Figuras

2.1	'Long-tail':instituições físicas só podem fornecer o que é popular, enquanto que instituições online podem fornecer tudo o que está disponível	9
2.2	Comparação entre as 3 abordagens contextuais	12
3.1	Interface exemplo da framework Swagger	18
3.2	Correlação de Pearson	20
5.1	Vista lógica de nível 1	33
6.1	Especificação do comportamento de uma tarefa	36
7.1	Scrum	40
7.2	Kanban Board	41
7.3	Planeamento do 1º semestre	43
7.4	Desvios ao planeamento no 1º semestre	44
7.5	Planeamento do 2º semestre	45
7.6	Desvios ao planeamento do 2º semestre	46
8.1	Interface inicial	53
8.2	Interface de criação de uma lista	53
8.3	Interface de gestão de conteúdos	54
A.1	Listas de conteúdos (Alta fidelidade)	58
A.2	Listas de conteúdos (Baixa fidelidade)	58
A.3	Lista de conteúdos (Baixa fidelidade)	59
A.4	Opções de filtragem(Baixa fidelidade)	60
B.1	Riscos do mês de Fevereiro	62
B.2	Riscos do mês de Março	62
B.3	Riscos do mês de Abril	63
B.4	Riscos do mês de Maio	63
B.5	Riscos do mês de Junho	64

Capítulo 1

Introdução

1.1 Enquadramento

O projecto "TICE.Mobilidade -Sistemas de Mobilidade Centrado no Utilizador", cuja missão passa pela exploração de novas soluções para o transporte urbano, mais eficientes e abrangentes, é um projecto que conta com o apoio de vinte e oito parceiros e alguns organismos de investigação empenhados em criar um novo paradigma para mobilidade urbana.

Um dos vários subprojectos do TICE.mobilidade é o projecto BUSCA ou BUS Context Awareness. Este projecto tem com cliente final a ARRIVA, uma empresa de transportes que irá instalar um sistema de distribuição contextualizada de conteúdos nos seus transportes públicos. Este sistema tem como finalidade cumprir vários objectivos. O primeiro desses objectivos é proporcionar aos seus utentes uma viagem mais confortável tentando, para isso, diminuir o "tempo psicológico" de viagem, ou seja, o tempo que a viagem está a demorar do ponto de vista do utente. Os outros objectivos passam por atrair mais pessoas para os transportes públicos, diminuindo assim a pegada ecológica de uma cidade, os gastos em transportes privados e também a obtenção de lucros através da distribuição de conteúdos publicitários que sejam do agrado do utilizador.

Este estágio surge como um subprojecto do projecto BUSCA e é focado na construção do sistema de distribuição e recomendação de conteúdos.

1.2 Objectivos

O âmbito deste estágio consiste no desenvolvimento de um sistema que seleccione os conteúdos mais apropriados para um grupo de pessoas tendo em conta o contexto em que estão inseridos. O uso do contexto é o foco principal

do estágio pois permite otimizar a recomendação feita. Este sistema permitirá distribuir variados conteúdos, desde publicidade a eventos passando por notícias. Relativamente ao contexto será tido em conta a posição geográfica do utilizador, o contexto temporal (ex. a hora do dia) e o perfil do grupo de utilizadores ou os interesses de utilizadores individuais. A aplicabilidade deste sistema surge no âmbito do projecto BUSCA mas poderá ser extensível a outras situações e contextos.

1.3 Motivação

Estamos a viver na Era da informação e do conhecimento. Para trás fica o tempo onde o poder, quer seja político, social, económico ou até mesmo militar era baseado essencialmente na posse de bens materiais ou de capital financeiro. As vantagens competitivas de alguém com acesso a boa informação são enormes. Passam desde uma maior agilidade e capacidade de resposta a eventuais problemas e desafios até a uma maior capacidade de evolução social e pessoal.

A quantidade de informação com que lidamos no dia-a-dia é simplesmente colossal. Para termos uma ideia, a empresa CISCO Systems, líder em soluções para redes e comunicações, estima que no ano 2013 teremos 667 exabytes de informação a circular pela internet [1]. Para uma noção mais concreta deste facto, se criássemos uma pilha de iPad's, com trinta e dois gyabytes cada um, esta seria três vezes maior que o monte Everest. Neste cenário, é fácil desperdiçar tempo e recursos com informação que não nos é relevante e também não nos traz qualquer valor. Para resolver este problema uma das soluções é adaptar a informação que recebemos ao contexto em que estamos inseridos.

A motivação para a escolha deste estágio, não depende só do meu interesse por aprender, desenvolver e contribuir para a área de "Context Awareness", mas também porque este estágio está inserido no projecto TICE.Mobilidade. Um projecto de grande relevo que está inserido numa área que tem ganho cada vez mais importância, a mobilidade.

Capítulo 2

Estado da arte

Um sistema de distribuição de informação contextualizada é um sistema que permite fazer chegar aos seus utilizadores conteúdos que sejam mais relevantes consoante o contexto em que este está inserido [2, 3, 4, 5, 6]. Com o aumento do número de sensores nos aparelhos que nos rodeiam, o elevado crescimento na área da mobilidade e a gigantesca quantidade de informação que se encontra disponível, a contextualização é uma solução para vários problemas, entre os quais a eficiência da publicidade e o tempo e recursos gastos com informação que não é relevante para a audiência em causa.

2.1 Sistemas sensíveis ao contexto

Nesta subsecção serão apresentados e descritos alguns dos serviços que são semelhantes ao que vai ser desenvolvido no âmbito deste estágio. Será dada uma maior relevância à forma como estes serviços lidam com o contexto.

ELBA

O ELBA é um projecto que têm como objectivo desenvolver uma abordagem inovadora para a publicidade baseada na localização. Este projecto têm três vertentes distintas. Uma das vertentes consiste na distribuição de publicidade nos transportes públicos, publicidade esta relativa às lojas ao redor da posição do transporte. Outra vertente consiste na distribuição de publicidade em dispositivos móveis após o utilizador fazer uma pesquisa sobre algo que este necessita, por exemplo, uma farmácia ou um restaurante italiano. A última vertente consiste na implementação de um serviço de publicidade em supermercados ou centros comerciais que disponibiliza informações de descontos, promoções ou outros conteúdos interessantes sobre lojas e produtos que estejam numa área relativamente próxima do utilizador.

Compass

O Compass é um serviço para smartphones, da Jiwire, que permite distribuir publicidade tendo em conta a localização do utilizador e os descontos e lojas nas proximidades. Este serviço mostra informações dos produtos e lojas e para além disso dá direcções das mesmas lojas. Uma característica bastante importante deste serviço é o facto de este criar um grafo que inter-relaciona os vários POA's de forma a melhor poder fazer recomendações. Este inter-relacionamento têm em conta as localizações passadas e a localização presente do utilizador de forma a por exemplo saber que, se um utilizador está no café tem uma probabilidade alta de num instante seguinte se dirigir a um cinema.

Gimbal

O Gimbal é uma plataforma sensível ao contexto para Android e iOS desenvolvida pela Qualcomm. Foi desenvolvida com o intuito de distribuir conteúdo relevante de uma maneira oportuna e personalizada para utilizadores de sistemas móveis. De todas os sistemas encontrados e analisados este é, sem dúvida, um dos mais abrangentes em termos contextuais. Isto é, este serviço é um dos que mais se destaca dos outros porque usa outros contextos sem ser a localização do utilizador.

Google Ads

O Google disponibiliza um serviço de publicidade contextualizada, o Google Ads. O funcionamento deste serviço é bastante simples. Aquando de uma procura o google verifica quais são os anúncios publicitários que correspondem à pesquisa que foi feita e mostra os mesmos ao utilizador. Estes mesmos anúncios também podem aparecer em sites cujo conteúdo se adequa ao anúncio. Para além disso, o google também conta com uma opção para publicitar dentro de uma certa área geográfica, usando o Google AdWords Express.

Millennial Media

Millennial Media é um serviço de publicidade virado para plataformas mobile com um alcance de dezenas de milhares de sites e aplicações. Este serviço também já começa a adoptar serviços de contextualização disponibilizando, de momento, opções para direccionar os conteúdos para determinadas regiões geográficas, sistema operativo e tipo de dispositivo. Este serviço também permite aos seus utilizadores consultarem algumas métricas, tais como o número de visualizações da publicidade e a fill rate (percentagem de pedidos que correspondem a um determinado conteúdo).

Sapo Voucher

O Sapo Voucher é um serviço, português, que disponibiliza descontos e promoções bastante variados, desde jantares em restaurantes a actividades radicais. Estes descontos estão organizados por zona, no entanto, esta limitação geográfica é feita de uma maneira bastante alargada o que, aliado a este ser a única variável contextual, implica que os conteúdos que são mostrados ao utilizador possam divergir bastante do que este pretende.

Voucher Cloud

O Voucher Cloud é uma serviço bastante idêntico ao Sapo Voucher, na medida em que também disponibiliza descontos e promoções segundo um critério geográfico. O que o diferencia do Sapo Voucher é a possibilidade de introduzir conteúdos de uma forma hiperlocal, isto é, os perímetros geográficos com uma área muito mais reduzida, o que permite distribuir conteúdos que tenham uma grande probabilidade de serem relevantes.

Rovio

Rovio é uma empresa de jogos, filandesa, mais conhecida pelo seu famoso jogo Angry Birds, que conta com mais de 130 milhões de utilizadores mensais. Esta empresa permite que anúncios publicitários sejam incluídos no jogo Angry Birds. Por defeito, este conteúdo é mostrado a jogadores de todo o mundo, no entanto, recentemente foi incluída uma opção que permite direccionar a publicidade para países específicos.

PlaceCast

O PlaceCast conta com um plano de negócios bastante parecido com o da VoucherCloud ou da Sapo Voucher, baseando o seu negócio à volta da publicidade, principalmente descontos e promoções. O PlaceCast permite que sejam disponibilizados conteúdos tendo em conta a posição hiperlocal, do utilizador, os seus gostos e o perfil maximizando assim a relevância do conteúdo que chega ao utilizador.

Mobitto

Mobitto é uma aplicação desenvolvida pela startup portuguesa Mobile Hangar e financiada pela estrela de futebol, Cristiano Ronaldo, que têm como objectivo interligar os utilizadores da aplicação com marcas e lojas/locais fazendo para isso uso do contexto geográfico. De uma forma bastante simples

a Mobitto é uma aplicação que nos mostra quais são os melhores descontos e promoções à nossa volta.

Conclusões

No seguimento da pesquisa e análise feita sobre serviços semelhantes ao que se vai desenvolver, chega-se à conclusão que a maior parte desses serviços se centra no contexto geográfico dos conteúdos e utilizadores negligenciando outras variáveis contextuais importantes como o perfil do utilizador e a hora a que vão ser distribuídos/recomendados os conteúdos, limitando assim o potencial comercial da distribuição de publicidade.

Chega-se também à conclusão de que a forma como os conteúdos são contextualizados é bastante arcaica excluindo alguns casos como a geração de grafos que inter-relacionam localizações de forma a gerar perfis de utilizadores, abordagem esta usada na aplicação Compass.

2.2 Sistemas de recomendação

Um sistema de recomendação é um tipo de sistema de filtragem de informação. Um sistema de filtragem de informação tem como objectivo principal a gestão do excesso de informação, ou 'infobesidade', um termo cunhado por Alvin Toffler no seu livro 'The third wave'. Num sistema de recomendação o propósito é mais específico focando o seu funcionamento na apresentação ao utilizador de conteúdos em que este está mais interessado, quer sejam filmes, músicas ou mesmo livros. Como mencionado na 'Encyclopedia of machine learning', o objectivo de um sistema de recomendação é gerar recomendações significativas para um grupo de utilizadores sobre itens ou produtos que lhes possam interessar[7].

Os sistemas de recomendação estão a ficar cada vez mais populares e a proliferar de uma forma bastante rápida. Isto deve-se a vários fenómenos, talvez o mais importante seja o que é apelidado de 'long tail'[8], um termo utilizado na Estatística para identificar distribuições de dados como a curva de Pareto (ver figura 2.1), onde o volume de dados é classificado de forma decrescente. Um exemplo disto é o facto de uma livraria focar a maior parte das suas vendas numa quantidade muito pequena dos livros que têm em stock, pois é impossível para esta ter fisicamente disponível todos os livros. Adaptando este exemplo ao mundo online podemos perceber que os sistemas de recomendação surgem como uma maneira de combater este fenómeno, recomendando itens do 'fundo da cauda' e de interesse, que de outra maneira poderiam não chegar ao conhecimento do utilizador.



Figura 2.1: 'Long-tail': instituições físicas só podem fornecer o que é popular, enquanto que instituições online podem fornecer tudo o que está disponível

Um revendedor online que rapidamente se apercebeu disto foi a Amazon.com cujo sistema de recomendações é conhecido como o 'Rei', um sistema que está em constante evolução há mais de dez anos e onde já foram gastos muitos milhões de dólares. A Netflix também é bastante conhecida pelo seu sistema de recomendações, o CineMatch, principalmente pelo desafio que lançou onde premiava com um milhão de dólares quem conseguisse melhorar em 10 por cento os seus resultados. A partir destes dados podemos concluir que os sistemas de recomendação assumem um papel bastante importante nos sistemas de filtragem de informação.

Existem vários tipos de sistemas de recomendação e soluções híbridas que tentam conjugar várias abordagens de forma a otimizar os resultados. Os sistemas de recomendação podem ser de uma forma geral divididos em três tipos. Os sistemas de filtragem colaborativa que se baseiam nas apreciações passadas de todos os utilizadores, os sistemas de filtragem baseada no conteúdo, onde a recomendação é feita consoante as apreciações feitas no passado sobre conteúdos semelhantes ou que correspondam ao perfil do utilizador, e as soluções híbridas que consistem na combinação de várias abordagens.

Passaremos de seguida explicitar melhor cada um destes sistemas.

2.2.1 Filtragem Colaborativa

Os sistemas de recomendação baseados em filtragem colaborativa analisam as semelhanças que existem entre o parecer dos utilizadores, parecer este que normalmente se encontra sobre a forma de uma classificação numérica. Estes sistemas baseiam-se no pressuposto de que se um utilizador A tem interesses semelhantes a um utilizador B relativamente a um conjunto de conteúdos, é provável que o mesmo se venha a verificar para outro conjunto de conteúdos. De uma forma mais simples, esta abordagem baseia-se no facto de que utilizadores semelhantes tenham opiniões semelhantes, ou como explicado por Jonathan L. Herlocker, Joseph A. Konstan e John Riedl - "A tarefa dos sistemas de filtragem colaborativa passa por prever quanto é que um utilizador vai gostar de um item, dado um conjunto de pareceres referentes a utilizadores que partilhem as mesmas opiniões"[9] .

A filtragem colaborativa pode ser subdivida em duas abordagens. Uma abordagem denominada 'memory-based' e outra denominada 'model-based'. A abordagem 'memory-based', uma das mais populares, utiliza toda a matriz de opiniões 'utilizador-item' para poder gerar uma previsão. Os sistemas que adoptam esta abordagem usam técnicas estatísticas para encontrar os utilizadores mais semelhantes entre si, a 'vizinhança'. Depois desta vizinhança estar formada são usados algoritmos para gerar uma previsão através da agregação dos ratings da vizinhança. Nas abordagens 'model-based' são dadas recomendações usando modelos estatísticos que são criados usando redes bayesianas, clustering ou outras abordagens. Na tabela 2.1 é apresentada uma matriz que permite comparar as várias técnicas que podem ser usadas.

2.2.2 Filtragem baseada em conteúdo

Sistemas de filtragem baseada em conteúdo focam-se nas características dos itens a recomendar. De uma forma geral este tipo de sistemas recomendam itens que sejam similares aos que um utilizador expressou uma opinião positiva. Resumidamente estes sistemas trabalham com um vector de características dos itens que podem ou não ter um peso/importância associado que servem para calcular a semelhança entre itens. Existem técnicas simples baseadas apenas em médias ponderadas e também técnicas mais elaboradas que usam classificadores Bayesianos, técnicas de clustering, árvores de decisões e até redes neuronais, no entanto, estas abordagens não foram amplamente

Categoria	Técnicas	Vantagens	Desvantagens
Memory Based	Sistemas baseados na vizinhança	implementação fácil facilidade em adicionar novos dados não existe necessidade de considerar o conteúdo	problemas de escalabilidade com grandes datasets
Model Based	Redes bayesianas Técnicas de clustering Redução dimensional	melhor escalabilidade bons resultados	construção do modelo performance diminui com dados escassos

Tabela 2.1: Comparação entre sistemas de recomendação

explorada pois os moldes deste projecto não ditam qualquer relação entre conteúdos, sendo estes demasiado genéricos para poder inferir características.

2.2.3 Soluções híbridas

As soluções híbridas surgiram da necessidade de criar sistemas de recomendação mais precisos e fiáveis aliando os melhores aspectos de cada abordagem. Existem várias alternativas para aliar o melhor destes dois mundos. Uma delas consiste em criar listas de recomendações em paralelo usando a abordagem colaborativa e abordagem baseada em conteúdo e depois combinar as duas através de uma média ponderada onde o peso da abordagem colaborativa aumenta proporcionalmente ao número de utilizadores que avaliam o item. Outras abordagens incorporam os métodos colaborativos dentro de conjuntos com conteúdos semelhantes, um exemplo disto é um sistema que funcione com filtragem colaborativa só para o género musical jazz.

2.2.4 Abordagens contextuais

De forma a otimizar os resultados dos sistemas de recomendação a contextualização é uma boa abordagem para alguns cenários. Existem três maneiras de poder incorporar o contexto nas previsões feitas pelos sistemas de recomen-

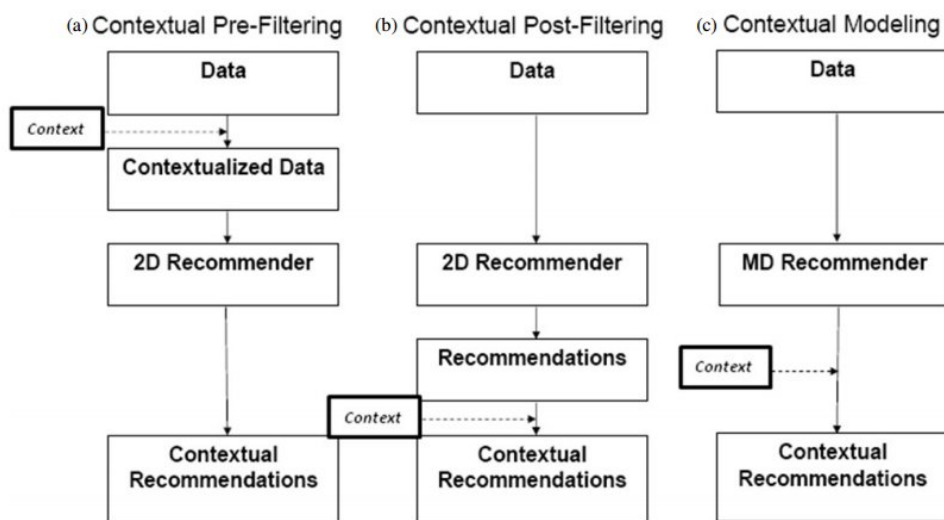


Figura 2.2: Comparação entre as 3 abordagens contextuais

dação, elas são a pré-filtragem, a pós-filtragem e a modelagem contextual. A principal diferença entre elas reside na altura em que é feita a contextualização como se pode ver na figura 2.2 retirada do artigo *Context-Aware Recommender Systems: A Comparison Of Three Approaches* [2]. Na pré-filtragem são filtrados todos os contextos semelhantes ao do utilizador e depois são aplicados métodos de recomendação usuais. Ao contrário da pré-filtragem, na pós-filtragem os métodos de recomendação normais são aplicados antes da filtragem contextual. Na modelagem contextual, o contexto é usado dentro dos algoritmos que vão gerar as recomendações.

2.2.5 Conclusões

A partir da análise ao estado da arte dos vários tipos de sistemas de recomendação existentes foram retiradas algumas conclusões. Primeiro ficou evidente que um sistema de recomendação baseado em conteúdo estaria fora de questão porque os conteúdos a serem recomendados são demasiado genéricos para se poder inferir características que permitam fazer uma comparação. Por outro lado também ficou evidente que os sistemas de recomendação baseados em memória apresentam melhores resultados que os sistemas baseados num modelo quando a quantidade de dados é pequena, sendo que essa diferença de performance desvanece-se quando o conjunto de dados é suficientemente grande. Relativamente às abordagens contextuais todas apresentam bons re-

sultados mas a abordagem de modelação contextual é a mais estável sendo que independentemente do dataset em questão a qualidade de resultados mantém-se estável, o que não acontece com as outras abordagens.

Capítulo 3

Solução proposta

“Every solution to every problem is simple. It’s the distance between the two where the mystery lies.”

Derek Landy

3.1 Definição do problema

Este estágio tem como principal objectivo lidar com o problema da monotonia experienciada pelos utilizadores dos transportes públicos. É normal que nestas viagens o tempo psicológico tenda a ser bastante mais longo que o tempo normal da viagem, visto que existem poucas distrações. Desta forma, no âmbito projecto BUSCA, serão introduzidos monitores no transportes públicos onde irão ser mostrados conteúdos, maioritariamente publicitários, de forma a combater esse problema. Por outro lado, isto também irá beneficiar potenciais interessados em distribuir publicidade.

De forma a garantir que os utilizadores dos transportes públicos realmente estão interessados nos conteúdos que lhes são mostrados chegou-se à conclusão que um sistema de recomendação contextualizado seria uma boa solução.

Neste capítulo pretende-se resumir de uma forma simples e directa a solução proposta. Como explicado na secção 1.2 este estágio tem como objectivo a criação de um sistema de distribuição e recomendação de conteúdos que seja sensível ao contexto. De forma a suportar o sistema de recomendação foi criado um sistema de gestão dos conteúdos que pode ser dividido em duas

partes. Um frontend, o "Advertiser Dashboard", que permite que o utilizador faça a gestão de conteúdos e uma API REST para que possa existir uma automação das acções do utilizador. Com estas componentes de gestão de conteúdos, surgiu um novo perfil de utilizador, que obrigou à criação de um sistema de permissões.

3.2 Advertiser dashboard

O "Advertiser dashboard" é um componente bastante importante do sistema de gestão de conteúdos, permitindo ao advertiser gerir os seus conteúdos usando a interface web da plataforma One.Stop.Transport (ver mockups presentes no apêndice A e o resultado final no capítulo 8) . De forma a que os conteúdos do utilizador permaneçam organizados e que as interações do utilizador com o sistema sejam mais simples foi tomada a decisão de associar sempre os conteúdos a listas. Estas listas poderão ser apenas uma forma de mais rapidamente identificar e gerir os conteúdos, como por exemplo dividir os conteúdos de várias campanhas publicitárias, ou rapidamente adicionar opções de filtragem contextual, como por exemplo, a criação de uma lista para conteúdos que sejam apenas disponibilizados para a zona de Coimbra durante o período da tarde.

Quando o utilizador acede à secção de gestão de conteúdos é lhe mostrado um ecrã com todas as suas listas e um botão no início da listagem que lhe permite de uma forma rápida e intuitiva adicionar uma nova lista.

Para adicionar uma nova lista o utilizador irá definir um nome, uma pequena descrição e algumas variáveis contextuais como um perímetro geográfico hiperlocal, um intervalo temporal, perfis de utilizadores, e rotas de transportes públicos de forma a direccionar os conteúdos de acordo com as suas preferências.

Aquando a criação da lista será possível ao utilizador adicionar então os conteúdos usando um mecanismo simples e intuitivo de "drag and drop".

3.3 API

De forma a que o processo de gestão e distribuição de conteúdos possa propiciar a integração com sistemas externos foi criada uma API RESTful. Esta API partilha as mesmas funcionalidades do Frontend mas também permite a inserção de classificações de conteúdos por parte de utilizadores autorizados.

Para documentar a API, de forma a que possa ser usada por outros programadores, optou-se por usar a framework Swagger, da empresa Reverb.

Esta framework permite descrever, visualizar e consumir uma API REST. Isto é útil porque em vez de o programador ter apenas linhas de texto e código para o orientarem no uso do API, tem uma interface gráfica, simples, directa, intuitiva e que lhe permite testar rapidamente a API (ver figura 3.1) usando a interface gráfica.

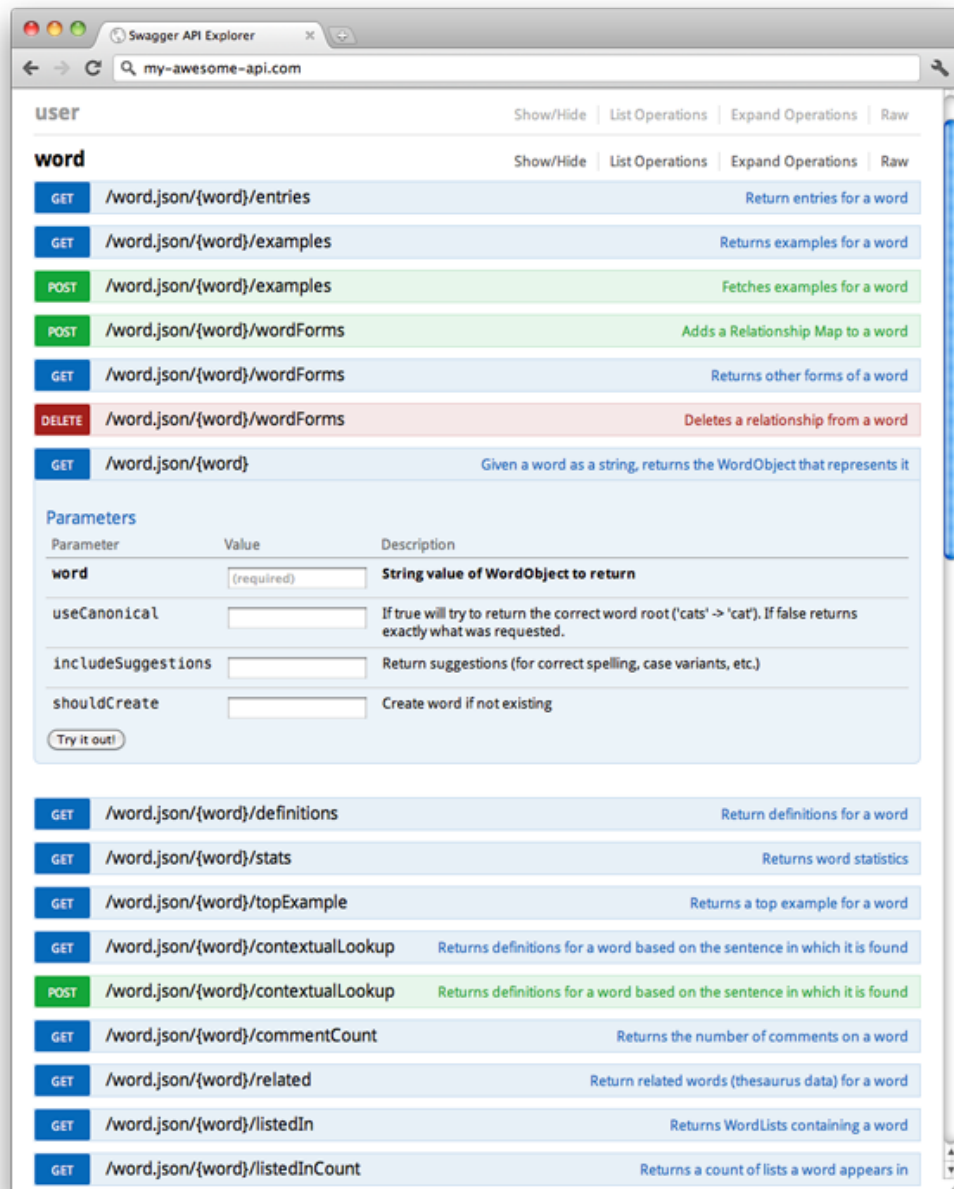


Figura 3.1: Interface exemplo da framework Swagger

3.4 Sistema de permissões

Para controlar os acessos quer à API quer ao "Advertiser Dashboard" foram usadas duas aplicações "open source". A aplicação django-user-roles serviu para controlar os níveis de acesso segundo o tipo de utilizador, neste caso só o novo perfil criado, o "advertiser" é que irá ter acesso às novas funcionalidades implementadas. Esta mesma aplicação, a 'django-user-roles', sofreu alguns melhoramentos que acabaram por se traduzir numa contribuição "open source" por parte do estagiário. A outra aplicação, a 'django-guardian', foi usada para controlar os acessos por objecto, de forma a que os utilizadores não conseguissem aceder e/ou modificar conteúdos que não lhe pertenciam.

3.5 Sistema de recomendação

Este trabalho pressupõe a construção de um sistema de recomendação, sistema este que difere do que é comum porque usa o contexto para melhor recomendar itens [5]. Como os conteúdos a ser recomendados por este sistema não têm nenhuma categoria associada, isto é, podem ser de qualquer tipo tem de se optar por uma abordagem colaborativa, como explicado na secção 2.2.1.

Das várias abordagens descritas em 2.2.4 a que foi escolhida para aplicar a este trabalho foi a modelagem contextual pois é a que obtém melhores resultados e ao mesmo tempo é a mais estável [2, 3, 4], no sentido em que independentemente do dataset a consistência nas qualidade previsões mantêm-se. A solução final é uma derivação do trabalho de Annie Chen, no seu paper "Context-Aware Collaborative Filtering System: Predicting the User's Preference in the Ubiquitous Computing Environment", onde esta sugere uma abordagem para um sistema de recomendação baseado em memória, colaborativo e contextual.

O que este sistema faz, de uma forma simples, consiste em usar os pareceres dos utilizadores acerca de conteúdos, pareceres estes sobre a forma de uma votação, e analisá-los para assim poder prever que conteúdos devem ser distribuídos para cada contexto. Como seria impossível gerar um dataset no tempo necessário para este estágio foi usado um dataset relativo a um projecto de investigação em Personalização Contextual, amavelmente facultado pelo professor Andrej Kosir da universidade de Ljubljana na Eslovénia.

O primeiro passo do sistema de recomendação é construir os perfis dos utilizadores para depois medir a semelhança entre eles e por fim calcular as previsões dos "ratings".

3.5.1 Perfis de utilizador

Normalmente a construção dos perfis de utilizador é gerada através da semelhança entre os pareceres (ratings) dos utilizadores. Neste caso como existe um projecto a par deste cujo intuito é de monitorizar e prever os perfis de utilizadores num determinado transporte público e num determinado horário a abordagem a usar não será essa. Como também ainda não existem resultados desse estágio, os perfis foram criados usando a faixa etária dos utilizadores.

3.5.2 Medição de semelhança entre perfis

Para medir a semelhança entre os perfis foi usado o coeficiente de correlação de Pearson. O que este coeficiente faz é medir o grau de relação linear entre duas variáveis (ver figura 3.2). Como se pode ver na seguinte equação :

$$w_{a,u} = \frac{\sum_{i=1}^m (r_{a,i} - \bar{r}_a) \cdot (r_{u,i} - \bar{r}_u)}{\sigma_a \cdot \sigma_u} \quad (3.1)$$

O que esta equação faz é combinar a semelhança dos ratings dados pelos perfis de utilizador a e u para devolver um valor entre -1 e 1, sendo que, -1 significa gostos opostos e 1 significa que os dois perfis possuem o mesmo gosto.

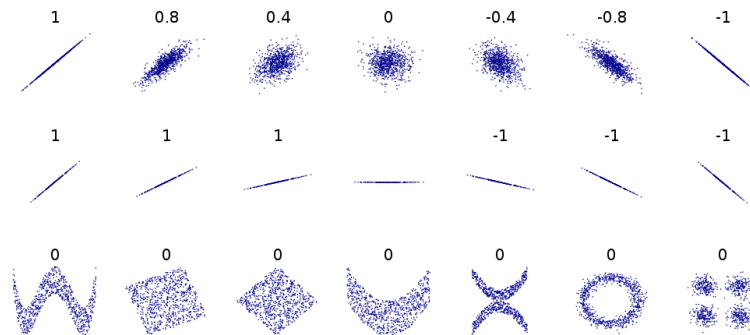


Figura 3.2: Correlação de Pearson

3.5.3 Cálculo das previsões

Caso este sistema não fosse contextual seria possível calcular as previsões usando a seguinte formula:

$$p_{a,i} = \bar{r}_a + k \sum_{u=1}^n (r_{u,i} - \bar{r}_u) \cdot w_{a,u} \quad (3.2)$$

onde a , é o perfil, i corresponde ao item, n é o numero de perfis e k é um factor de normalização.

Como este sistema é contextual torna-se necessário fazer algumas modificações a esta equação de forma a determinar o peso de cada rating tendo em conta a semelhança entre contextos.

Para isso redefine-se o novo rating como :

$$R_{u,i,c} = k \sum_{t=1}^z r_{u,i} \cdot sim(t, x) \quad (3.3)$$

onde u é o perfil, i o item, c o contexto actual e t a média dos contextos dos ratings do perfil u sobre o item i .

sendo assim obtemos a equação:

$$p_{a,i,c} = \bar{r}_a + k \sum_{u=1}^n (R_{u,i,c} - \bar{r}_u) \cdot w_{a,u} \quad (3.4)$$

Este cálculo combina todos os ratings ponderados tendo em conta a semelhança de contextos e os outros perfis de utilizador.

Esta abordagem deriva da abordagem de Annie Chen, sendo que a mesma era orientada à previsão de conteúdos para utilizadores singulares e esta nova abordagem implica que as previsões sejam relativas a grupos/perfis de utilizadores. Sendo assim foram feitas algumas alterações que acabaram por se traduzir numa melhor performance temporal do algoritmo e não afectam em grande medida a qualidade dos resultados. Por exemplo, o rating de um determinado perfil sobre um item corresponde à média dos ratings de todos os utilizadores pertencentes a esse perfil e o mesmo acontece em relação aos contextos onde foi feito o rating.

Capítulo 4

Definição de requisitos

”The most difficult part of requirements gathering is not the act of recording what the user wants, it is the exploratory development activity of helping users figure out what they want.”

SteveMcConnell

A especificação dos requisitos do projecto foi feita sobre a forma de *user stories*. User stories são pequenas descrições das funcionalidades que os actores gostariam de usar no projecto. Estas *user stories* são usualmente escritas em pequenos pedaços de papel de modo a que não se perca tempo e energia a fazer algo com um nível de detalhe tão baixo que por alguma pequena mudança no projecto se tenha de refazer trabalho. Outros benefícios das *user stories* é permitirem um planeamento mais simples e rápido, estimularem a comunicação na equipa e evitarem uma falsa percepção de exactidão[10].

Neste capítulo serão apresentados os actores no projectos e os requisitos, sob a forma *user stories*, para eles relevantes.

4.1 Actores

Os actores que vão de alguma forma intervir no sistema são o Anunciante e o Utente. O actor Anunciante representa um indivíduo ou entidade que deseje inserir conteúdos na plataforma de forma a que estes sejam mais tarde distribuídos para outras pessoas. O actor Utente representa alguém que pede e recebe conteúdos da plataforma.

4.2 Requisitos

Como referido na secção 4 os requisitos foram definidos sob a forma de *user stories*. Foram utilizadas *user stories* porque estas tratam essencialmente das necessidades do utilizador [11] lidam bem com imprevistos e mudanças, pois estas tem de ser independentes umas das outras, o que faz com que seja mais simples e menos demorado fazer um replaneamento. Para além disto as user stories são de fácil leitura para pessoas sem conhecimentos técnicos [11] e ajudam na definição de testes unitários quer seja num estilo de BDD (Behaviour Driven Development) ou TDD (Test driven development).

Jonathan Rasmusson, propôs um modelo para as *user stories* no seu livro "The Agile Samurai"[10] onde definia cada *user story* contendo três partes, o actor, a acção, e a razão pela qual vai ser feito. O template seria o seguinte:

Enquanto <actor do sistema>
Quero <acção>
De forma a <proposta de valor>.

De forma a priorizar os requisitos usa-se a abordagem recomendada pela IETF em que aconselha o uso das palavra-chave "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", e "OPTIONAL".

A interpretação destas palavra-chave é conforme o que está descrito no RFC 2119.

De seguida serão apresentados os requisitos do projecto. Divididos essencialmente em três partes para uma melhor organização. Duas para a gestão de conteúdos, sendo elas um frontend (Advertiser Dashboard), e uma API REST e outra que contém os requisitos do sistema de recomendação. Estas três componentes estão explicadas em maior detalhe no capítulo 3.

4.2.1 Advertiser Dashboard

De seguida serão apresentadas as user stories referentes à componente de gestão de conteúdos através do frontend da plataforma. De forma a melhor compreender as "user stories" convém relembrar os seguintes conceitos:

Perímetros geográficos: Poligonos que delimitam uma área geográfica

Variáveis de contexto: Tempo, Localização, perfil

Grupos de conteúdos: Listas de conteúdos com algo em comum

User Story WEB-CLIENT-01-CRUD
Enquanto: Anunciante
Quero: adicionar conteudos
Versão: 1 Prioridade: MUST Estado: Concluida

User Story WEB-CLIENT-02-CRUD
Enquanto: Anunciante
Quero: remover conteudos
Versão: 1 Prioridade: MUST Estado: Concluida

User Story WEB-CLIENT-03-CRUD
Enquanto: Anunciante
Quero: listar conteudos
Versão: 1 Prioridade: MUST Estado: Concluida

User Story WEB-CLIENT-04-CRUD
Enquanto: Anunciante
Quero: associar perimetros geográficos ao conteudo
De forma: a limitar a distribuição de conteúdo
Versão: 1 Prioridade: MUST Estado: Concluida

User Story WEB-CLIENT-04-CRUD
Enquanto: Anunciante
Quero: associar perimetros geográficos ao conteudo
De forma: a limitar a distribuição de conteúdo
Versão: 1 Prioridade: MUST Estado: Concluida

User Story WEB-CLIENT-05-CRUD
Enquanto: Anunciante
Quero: associar intervalos temporais ao conteudo
De forma: a limitar a distribuição de conteúdo
Versão: 1 Prioridade: MUST Estado: Concluida

User Story WEB-CLIENT-06-CRUD
Enquanto: Anunciante
Quero: associar perfis de utilizador ao conteudo
De forma: a limitar a distribuição de conteúdo
Versão: 1 Prioridade: MUST Estado: Concluida

User Story WEB-CLIENT-07-CRUD
Enquanto: Anunciante
Quero: associar rotas de transportes publicos ao conteudo
De forma: a limitar a distribuição de conteúdo
Versão: 1 Prioridade: MUST Estado: Concluida

User Story WEB-CLIENT-08-CRUD
Enquanto: Anunciante
Quero: criar grupos de conteúdos
De forma: a organizar os conteúdos e a facilitar a associação a variáveis de contexto
Versão: 1 Prioridade: MUST Estado: Concluida

User Story WEB-CLIENT-09-CRUD
Enquanto: Anunciante
Quero: remover grupos de conteúdos
De forma: a organizar os conteúdos e a facilitar a associação a variáveis de contexto
Versão: 1 Prioridade: MUST Estado: Concluida

User Story WEB-CLIENT-01-ANALYTICS
Enquanto: Anunciante
Quero: visualizar dados referentes à distribuição da publicidade e ao gosto dos utilizadores
De forma: a poder ajustar o tipo de publicidade ou até o modelo de negócio
Versão: 1 Prioridade: MAY Estado: Não Concluida
Este requisito não foi iniciado porque se decidiu que a sua implementação deveria ser adiada

4.2.2 API

De seguida serão apresentados as user stories referentes à componente de gestão de conteúdos através de uma API. De forma a melhor compreender as "user stories" convém lembrar os seguintes conceitos:

Perímetros geográficos: Poligonos que delimitam uma área geográfica

Variáveis de contexto: Tempo, Localização, perfil, ...

Grupos de conteúdos: Listas de conteúdos com algo em comum

User Story API-CLIENT-01-CRUD
Enquanto: Anunciante
Quero: adicionar conteúdos
Versão: 1 Prioridade: MUST Estado: Concluida

User Story API-CLIENT-02-CRUD
Enquanto: Anunciante
Quero: remover conteúdos
Versão: 1 Prioridade: MUST Estado: Concluida

User Story API-CLIENT-03-CRUD
Enquanto: Anunciante
Quero: listar conteúdos
Versão: 1 Prioridade: MUST Estado: Concluida

User Story API-CLIENT-04-CRUD
Enquanto: Anunciante
Quero: associar perimetros geográficos ao conteudo
De forma: a limitar a distribuição de conteúdo
Versão: 1 Prioridade: MUST Estado: Concluida

User Story API-CLIENT-05-CRUD
Enquanto: Anunciante
Quero: associar intervalos temporais ao conteudo
De forma: a limitar a distribuição de conteúdo
Versão: 1 Prioridade: MUST Estado: Concluida

User Story API-CLIENT-06-CRUD
Enquanto: Anunciante
Quero: associar rotas de transportes publicos ao conteudo
De forma: a limitar a distribuição de conteúdo
Versão: 1 Prioridade: MUST Estado: Concluida

User Story API-CLIENT-07-CRUD
Enquanto: Anunciante
Quero: associar perfis de utilizadores ao conteudo
De forma: a limitar a distribuição de conteúdo
Versão: 1 Prioridade: MUST Estado: Concluida

User Story API-CLIENT-08-CRUD
Enquanto: Anunciante
Quero: criar grupos de conteúdos
Versão: 1 Prioridade: MUST Estado: Concluída

User Story API-CLIENT-09-CRUD
Enquanto: Anunciante
Quero: remover grupos de conteúdos
Versão: 1 Prioridade: MUST Estado: Concluída

User Story API-CLIENT-10-RECOMMENDATION
Enquanto: Anunciante
Quero: enviar feedback relativo à opinião de um utilizador acerca de um conteúdo
De forma: recomendar itens a outros utilizadores e priorizar conteúdos
Versão: 1 Prioridade: MUST Estado: Concluída

User Story SUB-01-BASIC
Enquanto: subscritor
Quero: pedir conteúdos à plataforma para uma determinada rota
Versão: 1 Prioridade: MUST Estado: Concluída

User Story SUB-02-BASIC
Enquanto: subscritor
Quero: pedir conteúdos à plataforma para um determinado perfil
Versão: 1 Prioridade: MUST Estado: Concluída

User Story SUB-03-BASIC
Enquanto: subscritor
Quero: pedir conteúdos à plataforma para um determinada posição
Versão: 1 Prioridade: MUST Estado: Concluída

User Story SUB-04-BASIC
Enquanto: subscritor
Quero: pedir conteúdos à plataforma para um determinado intervalo temporal
Versão: 1 Prioridade: MUST Estado: Concluída

User Story SUB-05-BASIC
Enquanto: subscritor
Quero: pedir conteúdos à plataforma para um determinado intervalo utilizador
Versão: 1 Prioridade: SHOULD Estado: Concluída

User Story SUB-07-STREAMING
Enquanto: subscritor
Quero: pedir ao sistema para criar um canal de streaming
De forma: a poder recomendar/priorizar outros conteúdos para ele e outros utilizadores
Versão: 1 Prioridade: MAY Estado: Não Concluída
Este requisito não foi iniciado porque não ter existido tempo para acabar o mesmo

4.2.3 Sistema

De seguida serão apresentados as user stories referentes ao sistema de recomendação e distribuição de conteúdos. De forma a melhor compreender as "user stories" convém relembrar os seguintes conceitos:

Perímetros geográficos: Polígonos que delimitam uma área geográfica

Variáveis de contexto: Tempo, Localização, perfil, ...

Grupos de conteúdos: Listas de conteúdos com algo em comum

User Story SYSTEM-01-FILTERING
Enquanto: sistema
Quero: Filtrar conteúdos consoante o posicionamento geográfico
De forma: contextualizar a publicidade
Versão: 1 Prioridade: MUST Estado: Concluída

User Story SYSTEM-03-FILTERING
Enquanto: sistema
Quero: Filtrar conteúdos consoante intervalo temporal ou instante
De forma: contextualizar a publicidade
Versão: 1 Prioridade: MUST Estado: Concluída

User Story SYSTEM-04-FILTERING
Enquanto: sistema
Quero: Filtrar conteúdos para uma determinada rota de um transporte público
De forma: contextualizar a publicidade
Versão: 1 Prioridade: MUST Estado: Concluída

User Story SYSTEM-05-PLAYLIST
Enquanto: sistema
Quero: Criar playlist de conteúdos consoante a filtragem
De forma: enviar conteúdos para um determinado intervalo temporal
Versão: 1 Prioridade: MUST Estado: Concluída

User Story SYSTEM-06-PLAYLIST
Enquanto: sistema
Quero: Exportar playlist de conteúdos para vários formatos
De forma: responder às necessidades de cada Anunciante
Versão: 1 Prioridade: MAY Estado: Concluída

User Story SYSTEM-07-STREAMING
Enquanto: sistema
Quero: Exportar playlist de conteúdos para vários formatos
De forma: responder às necessidades de cada Anunciante
Versão: 1 Prioridade: MAY Estado: Não Concluída

User Story SYSTEM-08-RECOMMENDATION
Enquanto: sistema
Quero: priorizar os conteúdos para um determinado utilizador
De forma: a melhor satisfazer o subscritor
Versão: 1 Prioridade: SHOULD Estado: Concluída

Lista de User Stories

Capítulo 5

Arquitectura e design

”Architecture is basically a container of something. I hope they will enjoy not so much the teacup, but the tea.”

Yoshio Taniguchi

Este capítulo tem como objectivo descrever a arquitectura do sistema que se pretende desenvolver. Como este trabalho vai ser integrado numa plataforma, a OST (One.Stop.Transport), com uma arquitectura já bem definida e um conjunto de tecnologias já associados à mesma, para a definição da arquitectura deste trabalho teve de ser ter em consideração a arquitectura da plataforma.

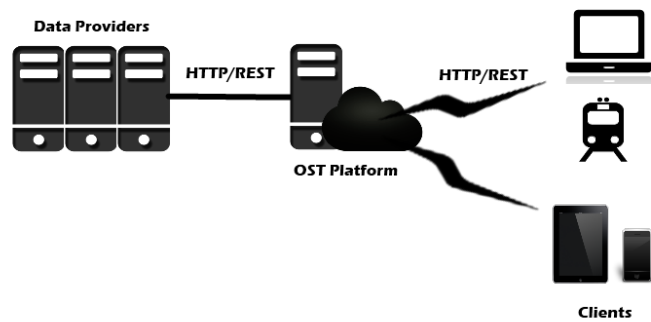
5.1 Vistas da arquitectura

De seguida serão apresentadas as vistas físicas e lógicas da arquitectura, onde se poderão ver as interacções entre os vários componentes da plataforma. Será feita uma explicação sobre o que é o trabalho do estagiário e qual é o trabalho que já existe feito pela equipa. Existem casos em que serão alterados módulos e esse reparo também será feito mais à frente neste documento.

5.1.1 Vista física

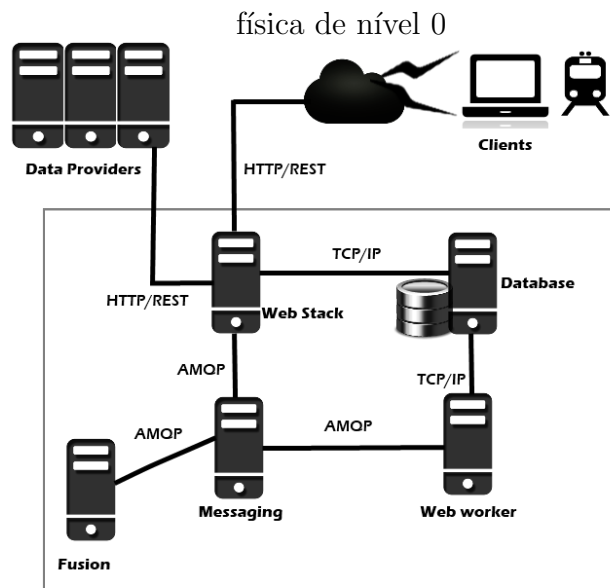
Na figura 5.1.1 podemos ver a perspectiva de alto nível do que é a arquitectura física da plataforma. Podemos ver deste diagrama que a plataforma serve como intermediária entre fornecedores de dados e clientes, tratando os dados dos fornecedores e enviando-os para os clientes de várias formas

e em vários contextos, respondendo às necessidades de cada subprojecto do TICE.mobilidade. Na visão de mais baixo nível (figura 5.1.1) podemos ver os vários componentes da plataforma OST.



figure[!h]

Vista



figure[!h]

Vista

física de nível 1

5.1.2 Vista lógica

Numa perspectiva lógica podemos dizer que o sistema se divide em dois módulos essenciais (existem outros na plataforma mas não fazem parte deste

projecto), são eles a Web Stack e o Fusion (Figura 5.1).

Web Stack - inclui todas as funcionalidades oferecidas aos Fornecedores de dados e Clientes, nomeadamente o portal da plataforma, as APIs de recepção e disponibilização de dados. Será alterado o módulo Web que corresponde ao frontend da plataforma para adicionar novas funcionalidades. Será também alterada a API para adição de mais funcionalidades.

Fusion - este módulo contém workers responsáveis por executar tarefas específicas como aceder a APIs dos Fornecedores de dados para adquirir novos dados ou o Parsing desses mesmos dados. Serão adicionados novos workers ligados ao sistema de recomendação de forma a poder escalar. Estes workers terão a responsabilidade de pré-computar alguns dados que depois serão pedidos através da API.

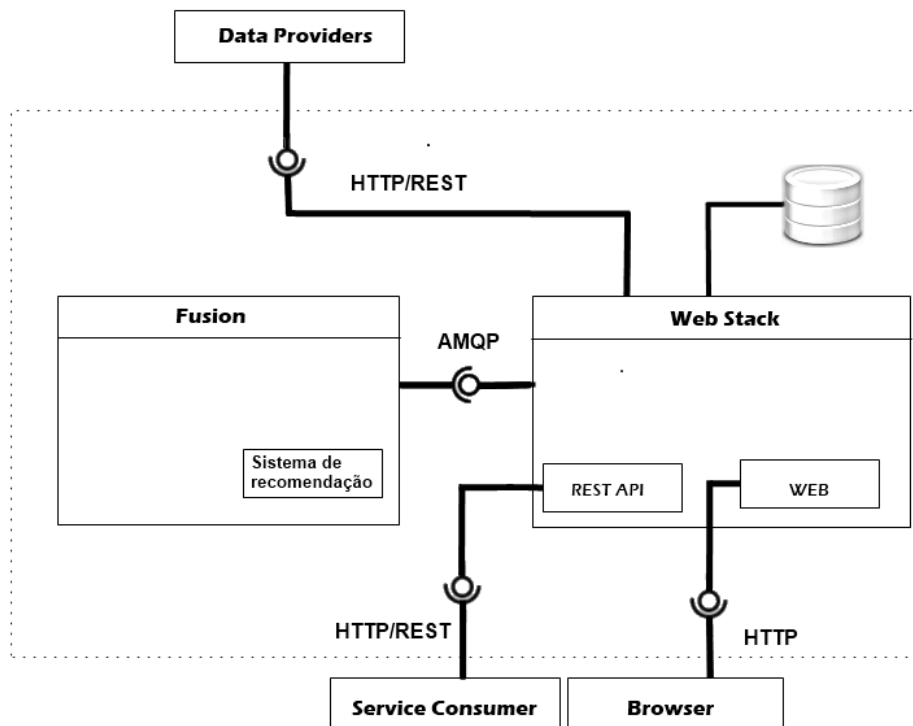


Figura 5.1: Vista lógica de nível 1

5.2 Tecnologias

Neste trabalho serão utilizadas algumas das tecnologias já em uso na plataforma em que o trabalho será integrado, no entanto, houve o cuidado pela parte do estagiário de verificar se as tecnologias em uso cumpriam o que era necessário para o estágio e se respondiam de forma adequada. De forma a enquadrar melhor o trabalho e perceber melhor a arquitectura vale a pena mencionar as mais importantes sendo elas as seguintes

Celery e RabbitMQ

Estas duas tecnologias são usadas para efeitos de "messaging" e serão usadas para escalar o sistema, principalmente para computações relacionadas com o sistema de recomendação.

Django

A framework MVC em que é desenvolvida a plataforma é a Django. Se a compararmos com outras plataformas como Ruby on Rails ou por exemplo a Play Framework o uso da framework Django é uma boa opção porque tem uma comunidade grande, activa e sendo a linguagem utilizada Python permitirá um desenvolvimento rápido.

PostgreSQL e Postgis

A base de dados a utilizar será uma base de dados PostgreSQL porque pode ser usada com a extensão PostGis, uma das melhores extensões para o armazenamento de dados espaciais.

Vagrant

Vagrant permite construir e replicar ambientes de desenvolvimento completos. Isto é uma mais-valia porque diminui o tempo perdido na construção do ambiente de desenvolvimento e garante que o resultado é idêntico ao ambiente de produção. O uso de Vagrant neste projecto torna-se assim essencial devido ao estado avançado da plataforma One.Stop.Transport e da quantidade de tecnologias nela presentes.

Capítulo 6

Validação

”Pay attention to zeros. If there is a zero, someone will divide by it”

Cem Kaner

Este capítulo tem como objectivo explicar em que é que vai constituir a estratégia de validação assim como a abordagem de desenvolvimento intrínseca a essa validação.

6.1 Estratégia de desenvolvimento

Neste estágio optou-se por seguir a abordagem Behavior Driven Development (BDD). A BDD é considerada uma evolução do TDD (Test-driven development) e pode resumir-se como a construção de uma descrição do comportamento pretendido (ver figura 6.1), normalmente em linguagem natural, para uma determinada funcionalidade que depois será mapeada em testes. Esta opção de usar BDD surge no seguimento das 'user-stories' usadas para definir os requisitos do sistema e enquadra-se com a metodologia ágil de desenvolvimento de software usada neste estágio (ver capítulo 7.1).

6.2 Testes

No que toca aos testes unitários estes serão feitos seguindo a abordagem de BDD referida na secção 6.1, no entanto não serão feitos qualquer tipo de testes à interface visual visto que esta está em fase de mudança, mudança esta a cargo de uma empresa contratada. A validação da qualidade do sistema de

```
Feature: Addition
  In order to avoid silly mistakes
  As a math idiot
  I want to be told the sum of two numbers

Scenario: Add two numbers
  Given I have entered 50 into the calculator
  And I have entered 70 into the calculator
  When I press add
  Then the result should be 120 on the screen
```

Figura 6.1: Especificação do comportamento de uma tarefa

recomendação basear-se-á principalmente na sua precisão, medida usando a diferença entre a predição feita pelo sistema e a avaliação real do utilizador. Para isto será usado o que se designa por "cross fold validation" ou validação cruzada.

6.2.1 Validação cruzada

A validação cruzada, ou "cross fold validation", é uma técnica de validação amplamente usada em análise estatística onde o objectivo é a previsão de resultados.

O método mais usado de validação cruzada é o "10 fold cross validation", que consiste em dividir o dataset em dez subconjuntos que serão depois usados, um de cada vez, para validação. A razão de em "k fold cross validation", ser maioritariamente usado $k=10$ é porque valores perto de 10 são suficientemente altos para reduzir a variância e ao mesmo tempo baixos de forma a não consumir muito tempo de testes[12].

De forma a medir a qualidade dos resultados do sistema de recomendação foram usadas diversas métricas. Foram usadas algumas métricas que medem o erro do sistema de recomendação e outras que medem a precisão e a quantidade de conteúdos relevantes para os utilizadores que lhes são recomendados, o "recall"

As métricas que medem o erro pretendem traduzir quão próxima uma previsão está do seu verdadeiro valor. Uma das métricas de erro usadas foi a MSE ou "Mean Squared Error" e sabendo que "p" é a previsão e

”a” trata-se do valor actual, MSE define-se por:

$$MSE = \frac{1}{n} \sum_{i=1}^n (p_i - a_i)^2 \quad (6.1)$$

Esta métrica é bastante fácil de calcular no entanto tende a exagerar os efeitos de valores atípicos [13]. Por isso também se utiliza outra variação a RMSE, ou ”Root Mean Squared Error”, definida por:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (p_i - a_i)^2} \quad (6.2)$$

Outra métrica que ainda é menos sensível a disparidades, sendo por isso uma das mais usadas [13], é a MAE ou ”Mean Absolute Error”, definida por:

$$MAE = \frac{1}{n} \sum_{i=1}^n |p_i - a_i| \quad (6.3)$$

Para avaliar o comportamento do sistema quando este tem recomendar uma porção pequena dos conteúdos são usadas mais duas métricas que irão medir a precisão do sistema e o recall [13].

Sendo VP um verdadeiro positivo, FP um falso positivo e FN um falso negativo. Temos para a precisão:

$$prec = \frac{VP}{VP + FP} \quad (6.4)$$

e para o recall:

$$recall = \frac{VP}{VP + FN} \quad (6.5)$$

Existe também uma métrica que combina os valores da precisão e de recall numa única medida e é denominada de F-Measure, definida por:

$$F - measure = \frac{2 * recall * precision}{recall + precision} = \frac{2 * VP}{2 * VP + FN + FP} \quad (6.6)$$

Capítulo 7

Abordagem

Este capítulo apresenta o plano de estágio, a metodologia de desenvolvimento de software aplicada, e ainda a análise dos riscos identificados durante o estágio.

7.1 Metodologia de Trabalho

A metodologia de desenvolvimento a ser usada é uma metodologia ágil baseada em Scrum a que a equipa adicionou algumas ferramentas como o Kanban Board, do Kanban, e o Test-driven development e o Pair Programming [14]. O facto de ser uma metodologia ágil traz grandes vantagens pois estamos a lidar com um projecto de grande dimensão onde existe colaboração entre vários parceiros não co-localizados o que dificulta a comunicação e proporciona uma maior instabilidade dos requisitos, as metodologias ágeis lidam com este problema aceitando a mudança como parte do processo de desenvolvimento e fazendo mudanças ao planeamento sempre que necessário [10].

O termo scrum está relacionado com o rugby e refere-se ao momento em que o jogo tem de ser recomeçado depois de algum acontecimento ter feito com que o jogo fosse interrompido. Um dos principais artefactos do scrum são os sprints. As sprints consistem num conjunto de iterações com uma duração, normalmente, entre duas a quatro semanas onde são executadas um pequeno conjunto de tarefas que estão no *sprint backlog*, isto permite que sejam disponibilizado ao cliente várias versões funcionais do produto de uma forma iterativa e incremental . Outros artefactos importantes são também o product backlog e a daily scrum meeting. Para uma melhor compreensão de como funciona o scrum ver figura 7.1 (página 40).

Este estágio foi dividido de acordo com as três fases do Scrum - Pré-Game, Game e Pós Game. Tanto a fase Pré-Game como a fase Pós-Game não são

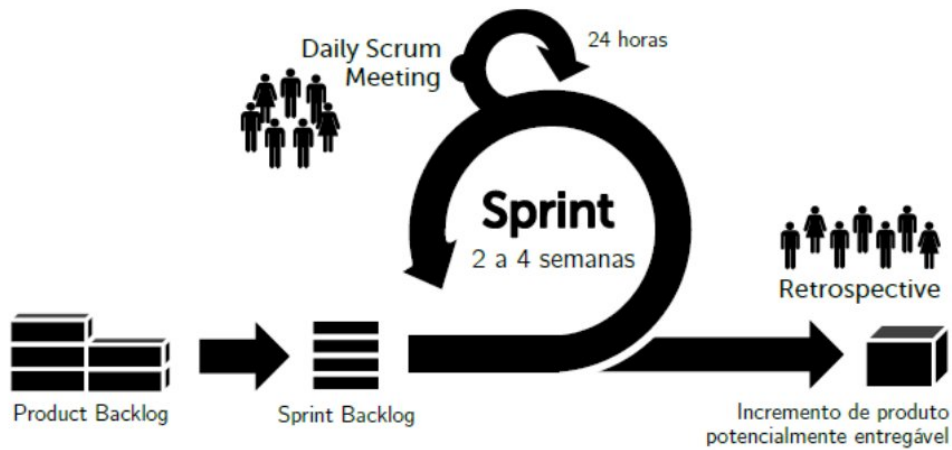


Figura 7.1: Scrum

divididas em Sprints, e portanto não têm backlogs associados. Os sprints terão a duração de duas semanas, conforme o que é normal na equipa do IPN. Os artefactos serão depois apresentados no relatório final. No contexto deste projecto, o Scrum Master é o Engenheiro Miguel Machado, coordenador técnico do projecto e orientador deste estágio por parte do IPN, e o Product Owner é o Engenheiro Alcides Marques, gestor dos projectos TICE. Falta acrescentar que a definição de "Done" associado a cada requisito implica que o código esteja completo, e que os testes tenham sido escritos e executados.

7.1.1 Ferramentas adicionais

Como foi referido anteriormente, serão usadas outras ferramentas de outras metodologias que não o Scrum. Uma dessas ferramentas é o Kanban Board, originário da metodologia Kanban, que consiste num quadro, dividido em secções, onde são colocados os cartões correspondentes às user stories. Com esta ferramenta é possível ter uma melhor percepção do estado do projecto e rapidamente detectar bottlenecks que possam estar a atrasar o projecto. Na figura 7.2 está representado o quadro usado pela equipa.

Para além do Kanban são usadas mais duas ferramentas, o Pair Programming e o Behaviour-Driven Development. O Pair-Programming consiste na produção de código num computador partilhado por dois elementos de uma equipa o driver e o observer, sendo o primeiro o que escreve o código, e o outro o que está atento ao código que está a ser produzido. As principais



Figura 7.2: Kanban Board

vantagens desta ferramenta resumem-se a uma maior e mais rápida disseminação de conhecimento, e a um aumento da qualidade do código produzido [15] [16]. Como este trabalho está enquadrado num estágio o uso de Pair-Programming foi usado pontualmente e apenas com o objectivo de potenciar a disseminação de conhecimento quer da plataforma quer das tecnologias usadas, onde o estagiário assumirá, maioritariamente, o papel de driver. O Behaviour-Driven Development é um processo de desenvolvimento baseado no Test-Driven Development e que será usado em conjunto com as "story tests" criadas a partir das "user stories".

7.2 Plano de trabalhos

"It always takes longer than you expect, even when you take into account Hofstadter's Law"

Hofstadter

Este subcapítulo serve para apresentar o planeamento para as diferentes fases da metodologia, *Pré Game*, *Game* e *Pós Game* através de diagramas de Gantt.

Na 7.3 podemos ver o que estava planeado para o primeiro semestre. Este planeamento não se concretizou em pleno e na figura 7.4 podemos ver quais é que foram os desvios associados ao plano inicial. O maior desvio foi o desenvolvimento de requisitos que se prolongou um pouco mais do que o planeado e retirou algum tempo à Prototipagem, o que não afectou muito o projecto pois o frontend deste trabalho vai ser baseado no frontend já existente da plataforma OST, de forma a manter a coerência entre interfaces.

Para o segundo semestre foi prevista uma fase inicial para acabar a análise a algumas tecnologias, seguida da fase de desenvolvimento e, por fim, uma fase final (Pós-Game) para documentar o trabalho feito no relatório.

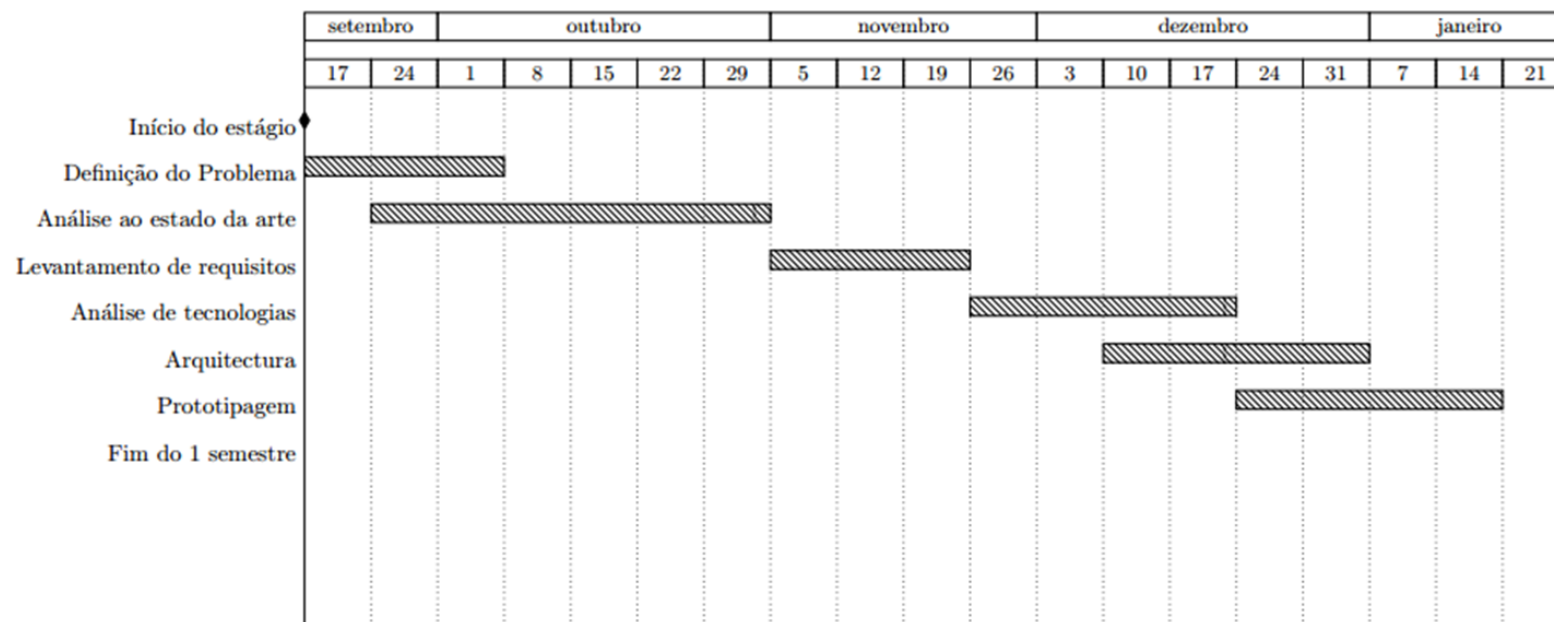


Figura 7.3: Planeamento do 1º semestre

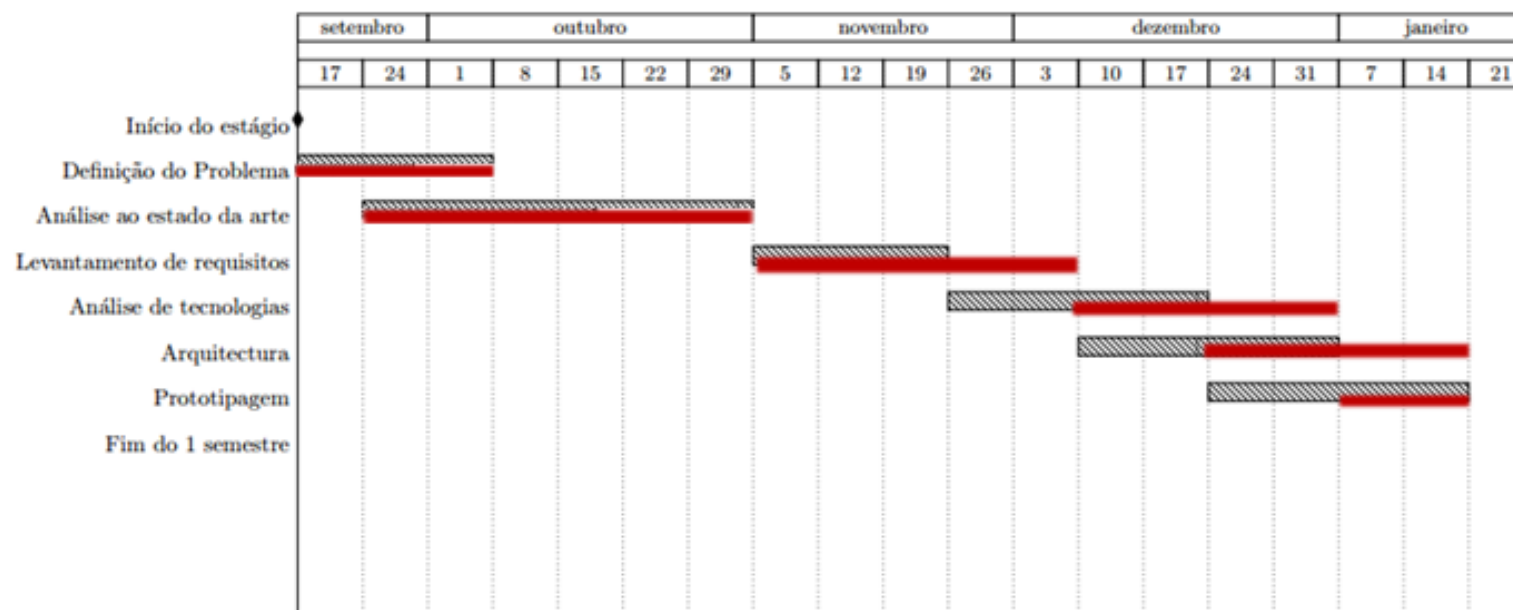


Figura 7.4: Desvios ao planeamento no 1º semestre

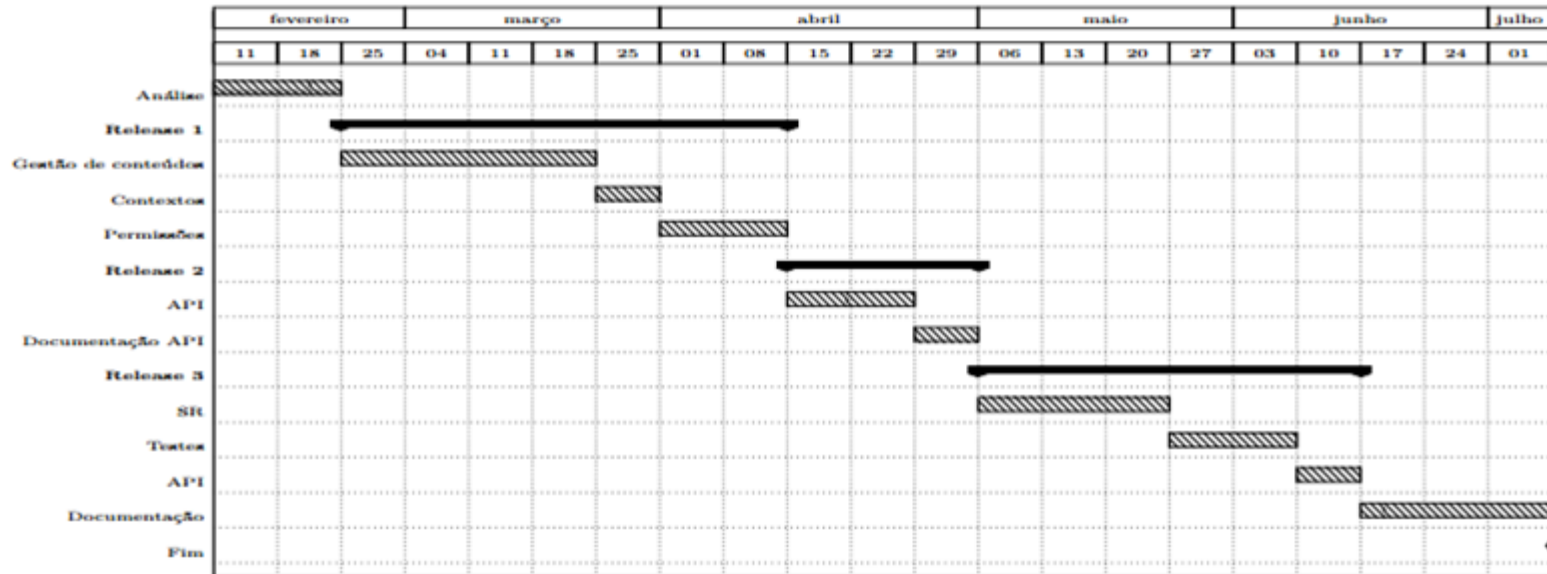


Figura 7.5: Planejamento do 2º semestre

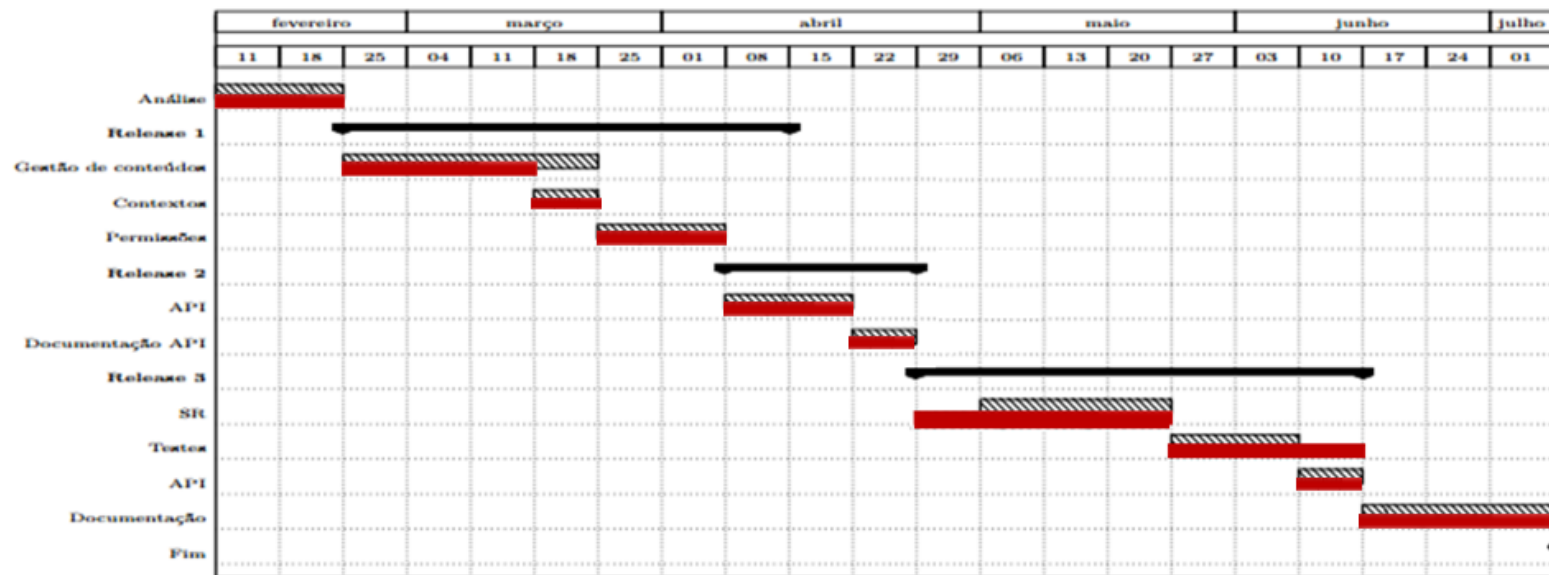


Figura 7.6: Desvios ao planeamento do 2º semestre

7.3 Análise de Riscos

"If there is no risk, there is no reward"

Unknown

Um risco é um acontecimento não desejado que tem algum tipo de prejuízo associado. Nenhum projecto está isento de riscos, por isso, é necessário fazer uma boa gestão, desde o início, para evitar que estes aconteçam. Este processo de gestão está dividido em quatro partes, sendo elas as seguintes:

Identificação - É nesta fase que é feita a identificação dos riscos. Esta tarefa pode ser feita pelo gestor de riscos ou pode ser discutida em equipa. Neste caso o estagiário optou por fazer esta tarefa sozinho e sem a ajuda instrumentos especiais para este efeito, tais como o TBQ (Taxonomy-based questionnaire) pois implicaria um esforço adicional que acabaria por não compensar no caso deste projecto. Os riscos devem ser concisos, serem baseados numa condição que é verdadeira no presente e terem uma consequência que não seja ambígua.

Análise - Esta fase é crucial para uma boa gestão de riscos. Aqui são atribuídos graus à probabilidade e ao impacto do risco no projecto. Normalmente esta tarefa é feita melhor com um gestor de riscos experiente.

Planeamento - O planeamento consistem em criar uma estratégia para gerir o risco, quer seja uma que evite o risco minimizando a sua probabilidade, ou uma estratégia que minimize o impacto do risco ou até mesmo a criação de um plano de contingência para se estar preparado quando o risco se materializar.

Monitorização - Os riscos não são estáticos, eles evoluem ao longo do tempo. Esta fase serve para actualizar a probabilidade e o impacto do risco e também para verificar se o risco ainda existe conferindo as suas premissas iniciais.

Neste capítulo são apresentados os principais riscos que foram encontrados no projecto.

7.3.1 Riscos do projecto

Na tabela 7.1 podemos ver uma listagem dos riscos do projecto que foram recolhidos até ao fim do primeiro semestre. Nesta tabela estão apresentados as quatro principais características dos riscos. O seu impacto e probabilidade que servem para priorizar os riscos e saber quais é que merecem mais atenção (estas métricas também podem ser usadas para criar uma matriz de riscos [17]), a sua descrição e a forma como o risco vai ser gerido.

Risco 01	Impacto Médio	Probabilidade Média
Descrição: A falta de um plano de negócio concreto associado ao projecto BUSCA pode levar a que o projecto possa sofrer atrasos e/ou possa vir a sofrer alterações nos seus requisitos		
Gestão: De forma a evitar o impacto deste risco, desde início se optou pelo uso de uma metodologia de desenvolvimento ágil que permitisse uma resposta rápida e eficaz a possíveis mudanças no projecto.		
Risco 02	Impacto Médio	Probabilidade Média
Descrição: A necessidade de desenvolver uma plataforma já existente e com a qual não se está familiarizado pode atrasar a implementação.		
Gestão: De forma a evitar o impacto deste risco, optou-se por adoptar uma ferramenta do Extreme Programming, o Pair-Programming, de modo a que a aquisição de conhecimentos sobre a plataforma seja feita a um ritmo mais elevado.		
Risco 03	Impacto Médio	Probabilidade Baixa
Descrição: A existência de vários parceiros e organismos de investigação, não co-localizados e em diferentes equipas, pode levar a que existam problemas de integração entre os trabalhos que estão a ser desenvolvidos.		
Gestão: De forma a evitar este risco, têm sido efectuadas reuniões que permitem perceber o estado dos projectos e qual a direcção que estes estão a tomar de forma a alinhar objectivos.		
Risco 04	Impacto Alto	Probabilidade Média
Descrição: O facto de, no âmbito do projecto BUSCA, não se conseguir obter feedback da opinião dos utilizadores pode levar a que o sistema de recomendação não seja efectivo.		
Gestão: Serão utilizados perfis de utilizadores resultantes de outro projecto que está a decorrer em conjunto com o uso de duas aplicações de mobilidade que também estão a ser desenvolvidas, de forma a obter um perfil colectivo e perfis individuais, respectivamente.		

Risco 05	Impacto Médio	Probabilidade Baixa
Descrição: O facto do hardware(para o projecto BUSCA) estar a ser desenvolvido por outro parceiro pode levar a que venham a surgir incompatibilidades ou dependências ainda não encontradas.		
Gestão: Serão efectuadas reuniões de forma a poderem ser validados os progressos efectuados.		

Tabela 7.1: Riscos

A monitorização dos riscos do projecto foi feita todos os meses e é apresentada no apêndice B sob a forma de matrizes de riscos

Capítulo 8

Conclusões

Terminado o estágio há que fazer uma análise crítica ao trabalho realizado e analisar que contribuições foram feitas.

8.1 Trabalho realizado

A primeira fase do estágio foi dedicada a analisar e a decidir qual seria o rumo que o projecto deveria tomar. Esta análise tentou responder a várias perguntas. Perguntas essenciais como o porquê (motivação), o como (metodologia, solução,...), o quando (planeamento) e o quê (requisitos, arquitectura, riscos...). A segunda fase deste estágio ficou associada à implementação e testes da solução encontrada.

O resultado final foi um enriquecimento do projecto TICE.Mobilidade através da criação de uma componente que gere a publicidade e que beneficia ao mesmo tempo os anunciantes e os utentes dos transportes públicos. Existiu um grande nível de aprendizagem a nível de tecnologias e métodos de trabalhos proporcionado pela equipa aquando da criação da componente de gestão de conteúdos e uma oportunidade de aprofundar os conhecimentos na área dos sistemas de recomendação e 'context-awareness' aquando da pesquisa e análise do que é o estado da arte.

Relativamente à componente de gestão de conteúdos pode-se dizer que o Advertiser dashboard é uma peça fundamental de todo o projecto de estágio, sendo que serve de suporte ao sistema de recomendação. É de interesse relembrar as principais funcionalidades deste 'Advertiser dashboard', sendo elas a criação de listas, como se pode ver na figura ?? e na figura 8.2, sendo que o propósito destas listas é essencialmente a organização e a facilidade na gestão dos conteúdos. Outra funcionalidade é a gestão de conteúdos , como se pode ver na figura 8.3, que mostra uma interface simples, que usa o 'drag

and drop' para a fácil adição de conteúdos.

A API também é bastante importante porque irá propiciar a integração com sistemas externos no futuro, de forma a facilitar o trabalho de quem vier a usar esta API foi usada a framework Swagger pelos motivos referidos em 3.3.

Por fim podemos falar do sistema de recomendação que é o principal componente diferenciador deste projecto, sendo que irá possibilitar uma melhor satisfação quer aos utentes dos transportes públicos quer aos anunciantes que irão ver os seus conteúdos serem distribuídos de uma forma mais dirigida e assim cativar mais pessoas. Os resultados da validação do sistema de recomendação também foram animadores sendo que as métricas usadas apresentaram resultados encorajadores, sendo eles os seguintes (ver capítulo 6 para compreender estas métricas):

- MSE = 1.08071654587
- RMSE= 1.03957517567
- MAE= 0.875120209685
- Precisão= 0.9021692 ...

Para melhor compreender estes resultados é importante referir que estes são referentes a uma escala de votação entre 1 e 5.

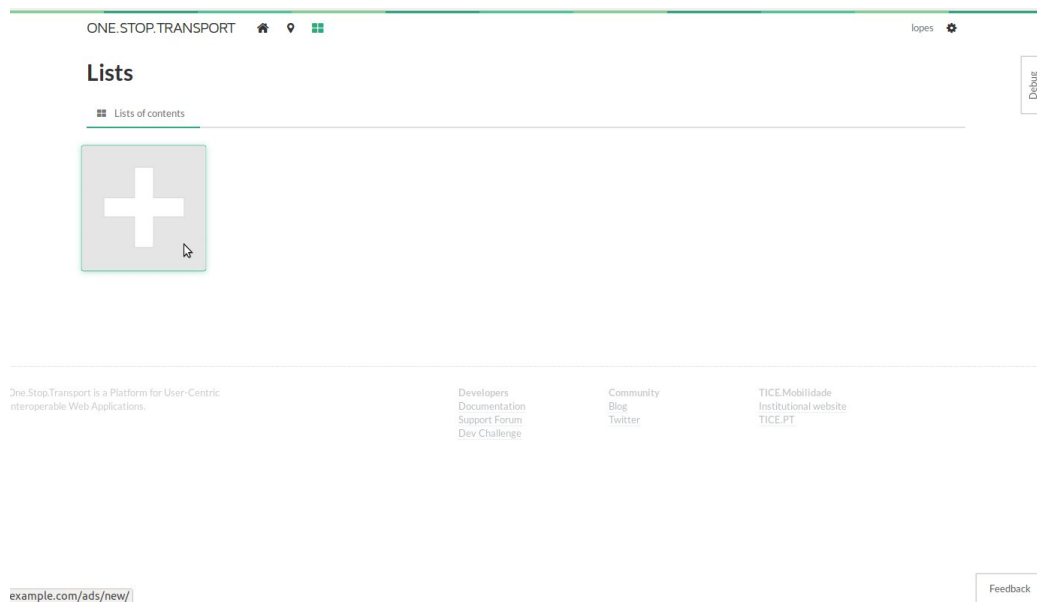


Figura 8.1: Interface inicial

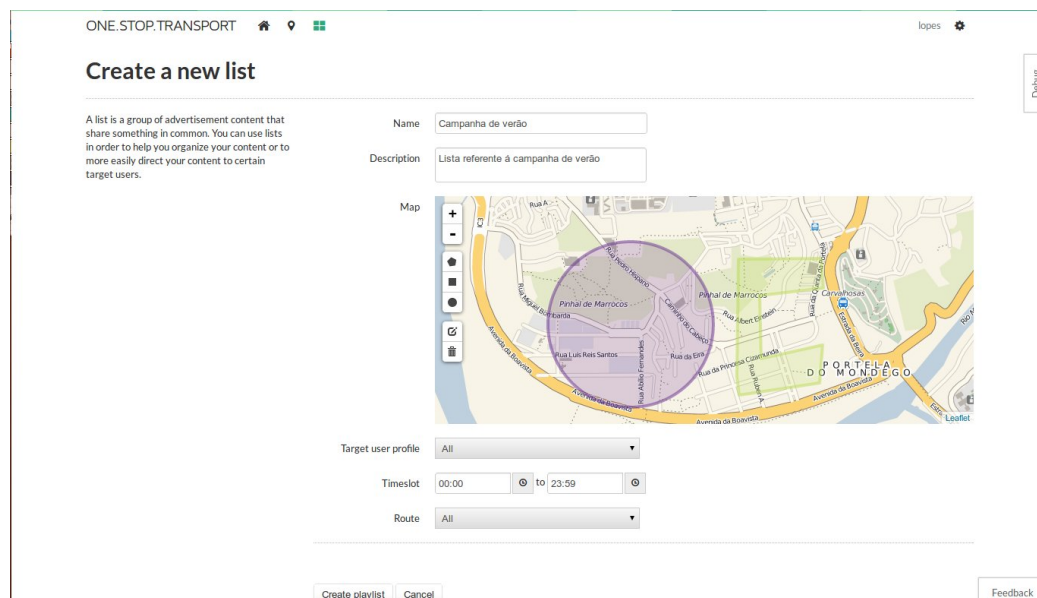


Figura 8.2: Interface de criação de uma lista

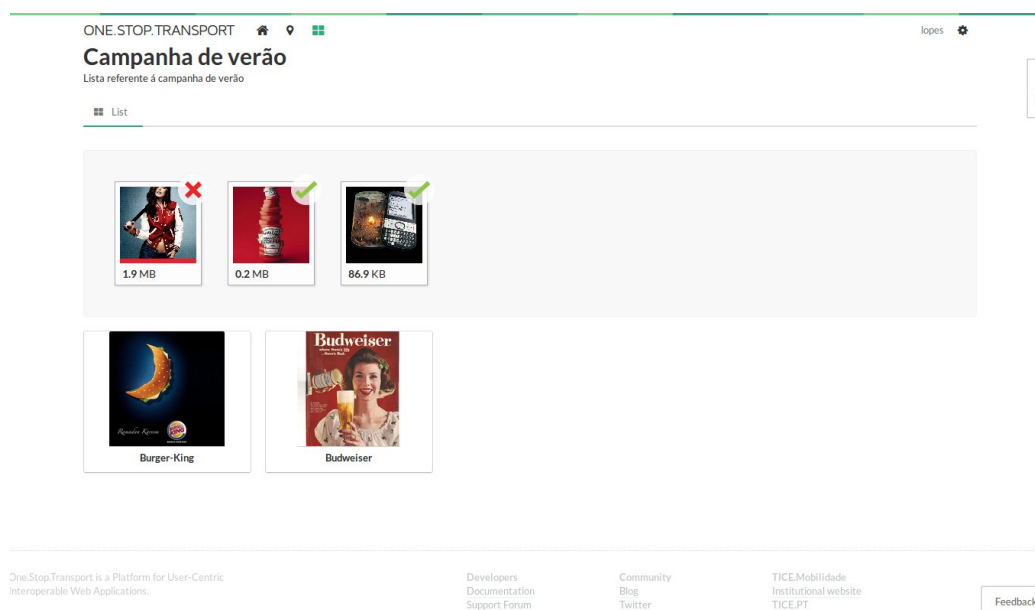


Figura 8.3: Interface de gestão de conteúdos

8.2 Contribuições

Durante este estágio, o estagiário teve a oportunidade de fazer uma contribuição 'open source' para uma aplicação que permite fazer gestão de permissões, a qual está disponibilizada para toda a comunidade de programadores Django. Para além disso também foi formulada uma nova abordagem do sistema de recomendação colaborativo e contextual proposto por Annie Chen, usando para validar o trabalho um dataset fornecido pelo Professor Andrej Kosir da Universidade de Ljubljana. Resultando tudo num trabalho com um objectivo bastante comum mas como uma abordagem diferente. Tudo isto integrado num projecto de grande dimensão e relevo.

8.3 Obstáculos

Durante a primeira fase do estágio existiram algumas dificuldades na definição do problema, o que dificultou o levantamento de requisitos e acabou por tirar algum tempo à prototipagem da interface. Isto não atrasou o andamento do projecto visto que o design da plataforma irá sofrer alterações, alterações estas que serão da responsabilidade de uma empresa contratada, fazendo com que as preocupações ao nível de design não tivessem de ser profundamente

ponderados.

Na segunda parte do estágio existiu alguma dificuldade na adaptação a algumas tecnologias existentes no projecto, por que estas nunca tinham sido usadas pelo estagiário, este problema foi combatido com o uso de Pair-Programming para ajudar a adaptação a estas tecnologias. Convém mencionar que o trabalho resultante do uso de Pair-Programming não faz parte do estágio e esta técnica foi usada apenas de forma a acelerar o processo de aprendizagem. Também existiram alguns problemas ao nível da validação do sistema de recomendação visto que foi necessário conseguir um dataset contextual, o que se revelou bastante difícil porque os trabalhos que giram à volta desta área são relativamente escassos e a maior parte dos que existem usam datasets que se encontram sobre acordos de confidencialidade.

8.4 Trabalho futuro

No futuro está previsto que este trabalho seja integrado com outro trabalho que permitirá prever os perfis que irão estar num determinado transporte público a uma determinada hora, deixando assim de se usar perfis baseados na faixa etária e passando a usar os perfis provenientes desse trabalho. Também está previsto adicionar uma componente que permitirá dar 'feedback' aos utilizadores do sistema de modo a que estes saibam como estão a ser distribuídos os seus conteúdos, quer seja em termos de localização, quer seja na dimensão temporal ou para que perfis de utilizadores.

Apêndice A

Prototipagem

Foram desenvolvidos alguns protótipos para ajudar a definir os requisitos do estágio, a que todos os intervenientes do projecto tivessem uma visão clara do que era proposto e todas as ideias estivessem alinhadas. Não foi dada muita importância aos protótipos de alta-fidelidade pois estes seguirão os moldes do frontend da plataforma.

Na figura A.1 podemos ver um protótipo de alta fidelidade, e na figura A.2 um protótipo de baixa fidelidade do que será o ecrã com as listas criadas por um determinado utilizador.

Estas listas são agregados de conteúdos e partilham entre si opções de filtragem de conteúdos. Na figura A.3 podemos ver os conteúdos de cada lista, e adicionar e remover conteúdos.

Para editar as opções de filtragem de conteúdos (ex. os conteúdos só podem passar na região de Coimbra) o protótipo representado pela figura A.4 demonstra como isso poderá ser feito.

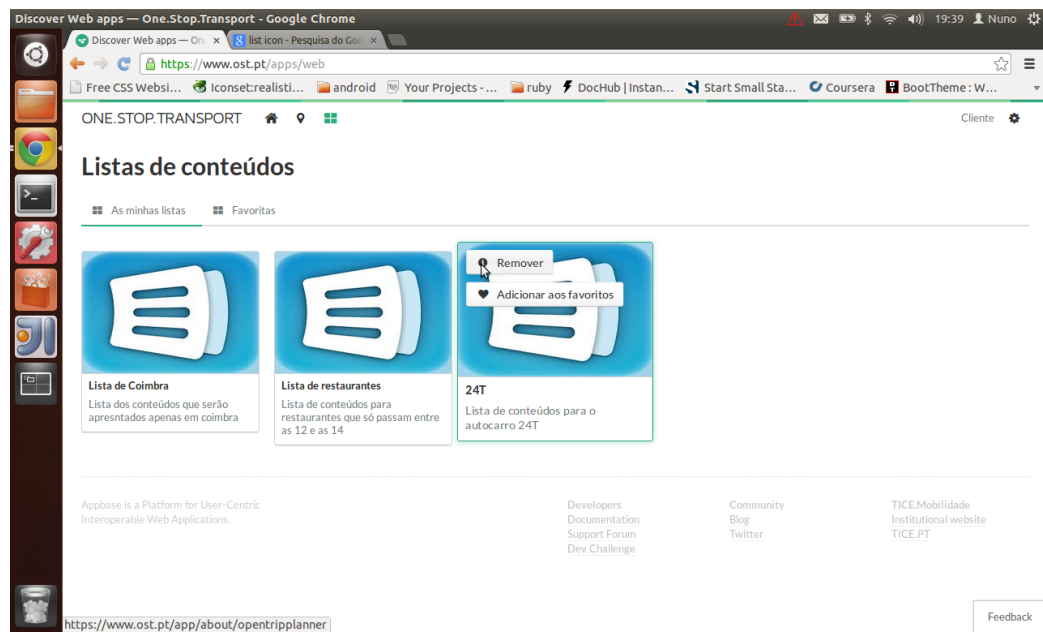


Figura A.1: Listas de conteúdos (Alta fidelidade)

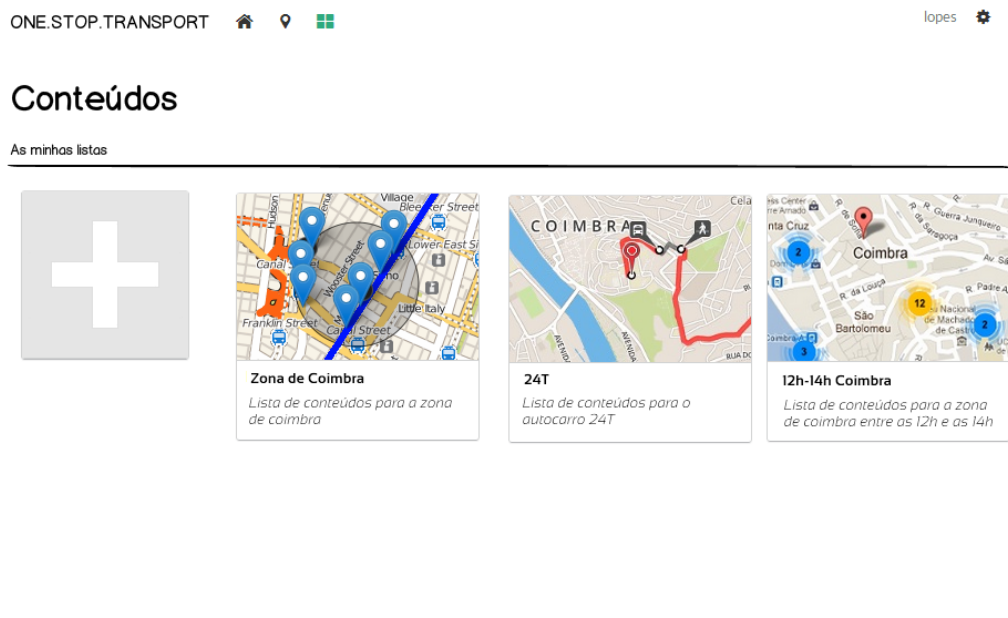


Figura A.2: Listas de conteúdos (Baixa fidelidade)

ONE.STOP.TRANSPORT

Zona de Coimbra

< Voltar Editar opções de filtragem

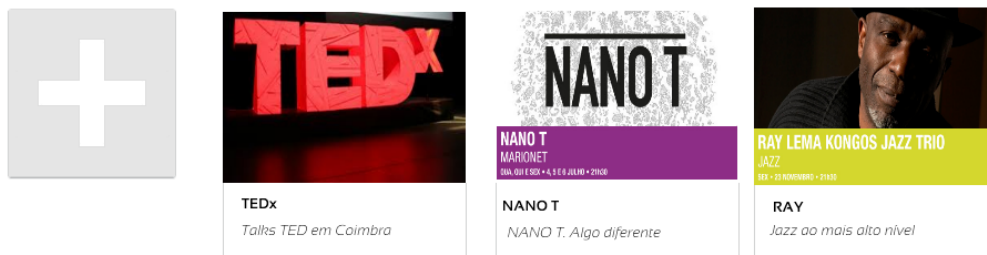


Figura A.3: Lista de conteúdos (Baixa fidelidade)

ONE.STOP.TRANSPORT

Zona de Coimbra

< Voltar

Perímetro geográfico



Perfil da pessoa

- Estudante
- Turista
- Reformado
- ...

Linha de transporte

- 24T
- 28
- 34
- 38
- 7
- ...

Figura A.4: Opções de filtragem(Baixa fidelidade)

Apêndice B

Monitorização de riscos

Este apêndice tem como objectivo apresentar a monitorização dos riscos deste projecto. Esta apresentação está sob a forma de uma matriz de riscos, de forma a que, rapidamente se possa definir quais os riscos com que nos devemos preocupar mais.

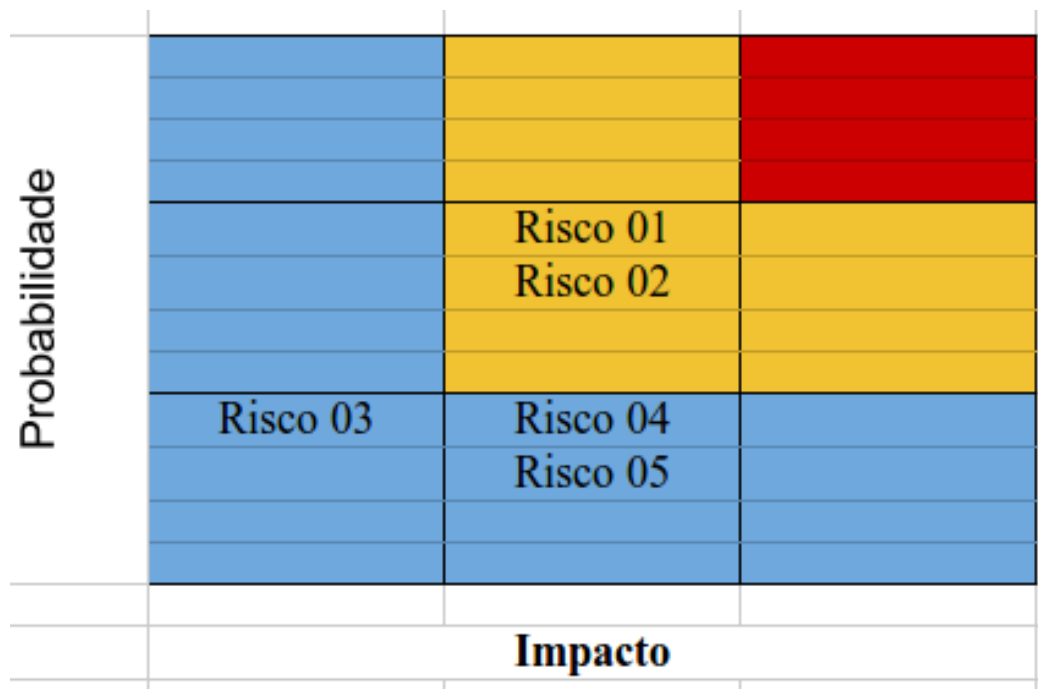


Figura B.1: Riscos do mês de Fevereiro

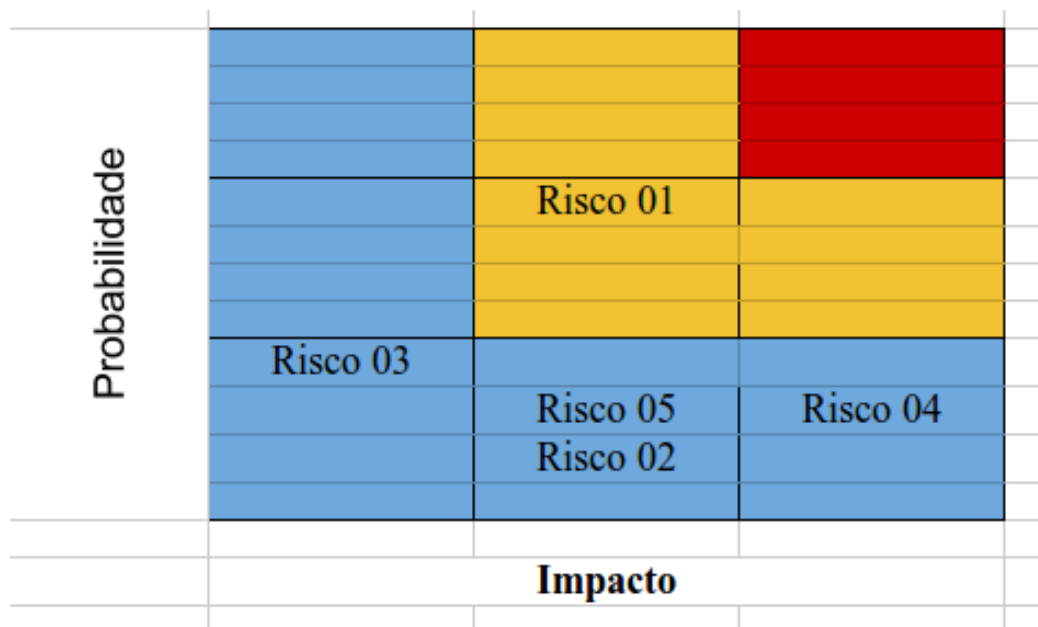


Figura B.2: Riscos do mês de Março

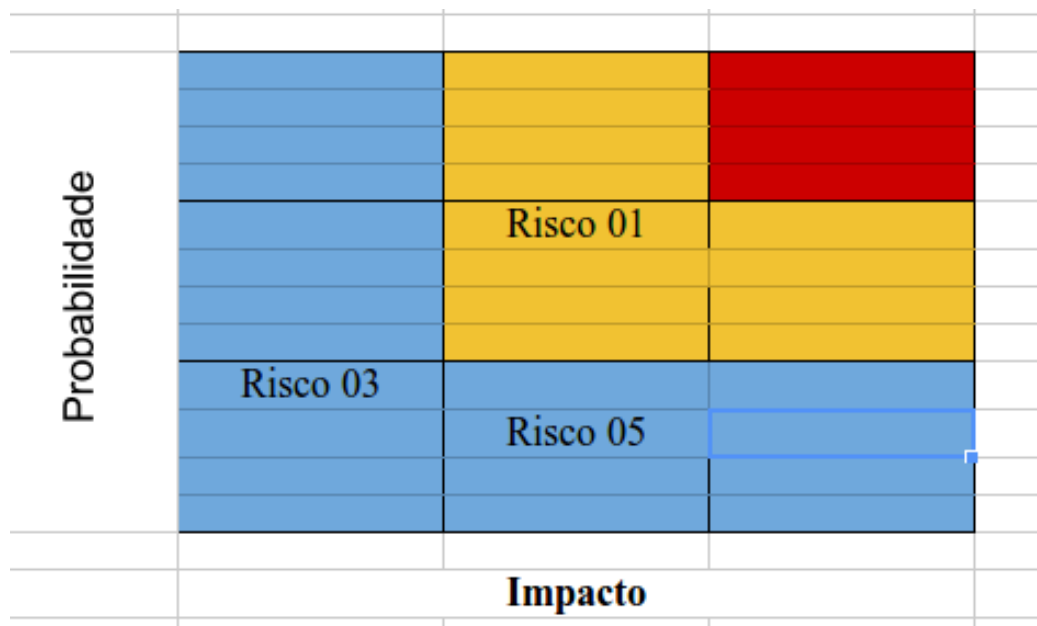


Figura B.3: Riscos do mês de Abril

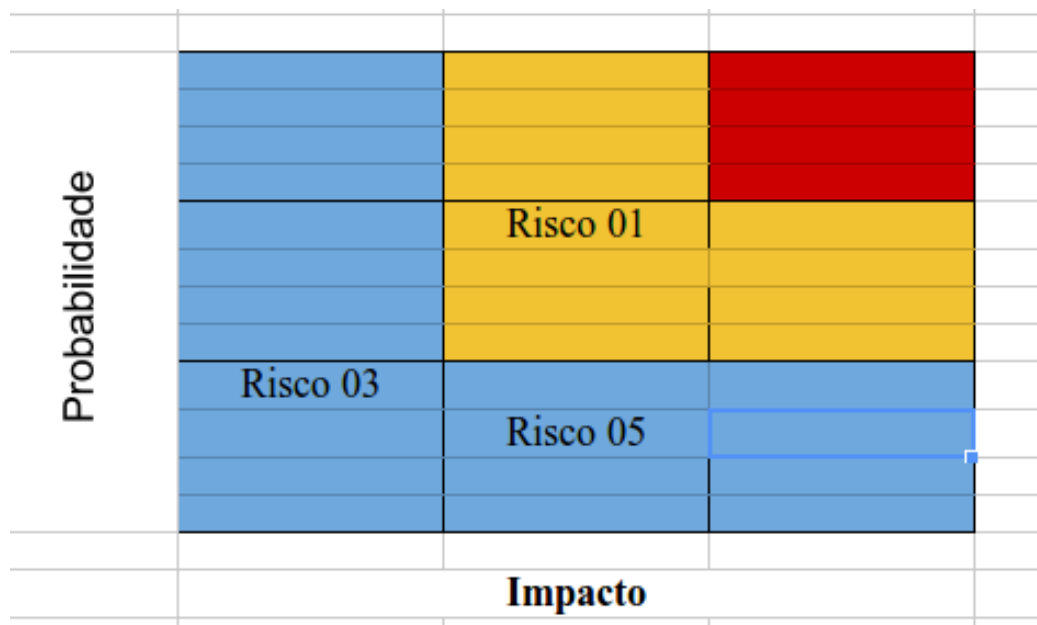


Figura B.4: Riscos do mês de Maio

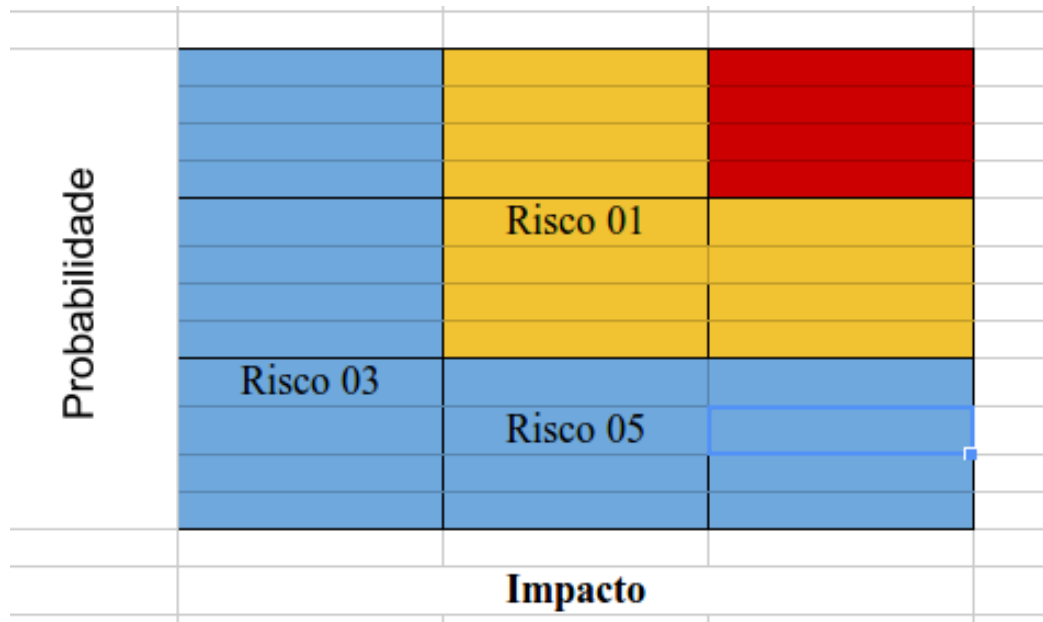


Figura B.5: Riscos do mês de Junho

Bibliografia

- [1] Lisa Gansky. *The Mesh: Why the Future of Business Is Sharing*. Penguin, 2010.
- [2] Michele Gorgoglione Umberto Panniello. Context-aware recommender systems: A comparison of three approaches.
- [3] Alexander Tuzhilin Umberto Panniello, Michele Gorgoglione. Comparing context-aware recommender systems in terms of accuracy and diversity: Which contextual modeling, pre-filtering and post-filtering methods perform the best.
- [4] Alexander Tuzhilin Cosimo Palmisano Anto Pedone Umberto Panniello, Michele Gorgoglione. Experimental comparison of pre- vs. post-filtering approaches in context-aware recommender systems.
- [5] Alexander Tuzhilin Gediminas Adomavicius. Context-aware recommender systems.
- [6] Michele Gorgoglione Cosimo Palmisano Anto Pedone Umberto Panniello, Alexander Tuzhilin. Experimental comparison of pre- vs. post-filtering approaches in context-aware recommender systems.
- [7] Geoffrey I; Sammut, Claude; Webb. *Encyclopedia of Machine Learning*. Springer, 2010.
- [8] Jerey D. Ullman Anand Rajaraman, Jure Leskovec. *Mining of Massive Datasets*. Cambridge University Press, 2012.
- [9] V. Kotlyar M.S. Viveros S.S. Duri R.D. Lawrence, G.S. Almasi. Personalization of supermarket product recommendations. data mining and knowledge discovery. 2001.
- [10] Jonathan Rasmusson. *The Agile Samurai: How Agile Masters Deliver Great Software*. The pragmatic bookshelf, 2011.

- [11] Jennifer Greene Andrew Stellman. *Applied Software Project Management*. O'Reilly, 2005.
- [12] Ron Kohavi. A study of crossvalidation and bootstrap for accuracy estimation and model selection. 1995.
- [13] Eugenio Lentini Paolo Cremonesi, Roberto Turrin and Matteo Matteucci. An evaluation methodology for collaborative recommender systems. 2010.
- [14] Henrik Kniberg. *Lean from the Trenches*. The pragmatic bookshelf.
- [15] Laurie Williams Alistair Cockburn. The costs and benefits of pair-programming. 2001.
- [16] Ward Cunningham Ron Jeffries Laurie Williams, Robert R Kessler. Strengthening the case for pair-programming. 2002.
- [17] Tim Lozier. Risk assessment: Creating a risk matrix, 2011.
- [18] A. Borchers J. L. Herlocker, J. A. Konstan and J. Riedl. An algorithmic framework for performing collaborative filtering. 1999.
- [19] Annie Chen. Context-aware collaborative filtering system: Predicting the user's preference in the ubiquitous computing environment. 2005.
- [20] Brent Smith Greg Linden and Jeremy York. Item-to-item collaborative filtering. 2003.
- [21] Simon Sinek. *Start with why, How great leader inspire everyone to take action*. Penguin, 2009.
- [22] Taghi M. Khoshgoftaar Xiaoyuan Su. A survey of collaborative filtering techniques.
- [23] Michael Jahrer Andreas Tosche. The bigchaos solution to the netix grand prize.
- [24] Chris Volinsky Robert M. Bell, Yehuda Koren. The bellkor 2008 solution to the netflix prize.
- [25] Michael Jahrer Andreas Toscher. The bigchaos solution to the netfix prize 2008.
- [26] Yehuda Korenr. The bellkor solution to the netix grand prize.

- [27] Junning Liu James Davidson, Benjamin Liebald. The youtube video recommendation system.
- [28] John Riedl J. Ben Schafer, Joseph Konstan. Recommender systems in e-commerce.
- [29] Qing Yang Liang Xiang. Time-dependent models in collaborative filtering based recommender system.
- [30] Anne Boyer Armelle Brun, Geoffray Bonnin. History dependent recommender systems based on partial matching.
- [31] Expresso. Os alicerces da cidade inteligente, Outubro 2012.