



Universidade do Minho

Mestrado Integrado em Engenharia Informática

Laboratório em Engenharia Informática

An Intelligent Tourist Guide (Route Planner)

Grupo 86

Luís Gomes (A78701)

Joel Rodrigues (A79068)

Elisa Valente (A79093)

25 de Junho de 2019

Resumo

O turismo é cada vez mais um setor de destaque na sociedade em que vivemos. Com o passar do tempo estão a ser atingidos níveis recorde do número de deslocamentos associados a esta atividade, demonstrando a crescente importância associada a este setor. Em modo exemplificativo, em Portugal, este setor é a maior atividade económica exportadora do país, contribuindo com 18.6% das exportações totais e 8.2% do PIB em 2018 [1].

Para dar resposta às necessidades impostas pela atividade turística têm vindo a ser criados sistemas com o objetivo de proporcionar melhor experiências aos viajantes. Indo de encontro a este problema, este projeto tem como propósito a criação de uma aplicação mobile que ofereça uma experiência personalizada de turismo de acordo com as preferências de cada pessoa. Assim sendo, a aplicação deve criar um plano de visita de uma cidade, tendo em conta o tempo e dinheiro que o utilizador tem disponível para o completar.

O grande foco deste projeto é a obtenção dos pontos que melhor se adaptam a cada utilizador e para isso são utilizados algoritmos genéticos, uma técnica de otimização que, tendo como *input* todos os pontos turísticos de uma cidade tem a capacidade de escolher aqueles que encaixam nas restrições e preferências impostas.

Conteúdo

1	Introdução	4
2	Estado de Arte	6
3	Requisitos	8
4	Análise e Armazenamento de Dados	9
4.1	Descrição do conjunto de dados	9
4.2	Armazenamento dos dados	10
5	Arquitetura da Solução Proposta	11
5.1	Estrutura externa	11
5.2	Estrutura interna	12
5.2.1	Algoritmo Genético	13
5.2.2	Heurística de seleção de restaurantes	14
6	Aplicação Desenvolvida	16
6.1	Ferramentas Utilizadas	16
6.2	Resultado Obtido	16
7	Conclusão	21

Figuras

1	Número de hóspedes em estabelecimentos hoteleiros portugueses.	6
2	Estrutura interna do sistema.	12
3	Registo, obtenção e atualização das preferências.	17
4	Página Inicial.	18
5	Planear uma viagem.	19
6	Realizar viagem.	19
7	Terminar viagem.	20
8	Apresentar Pontos de Interesse.	20

1 Introdução

Muitos visitantes recorrem aos serviços de guias turísticos quando pretendem conhecer uma nova cidade. Estes têm uma rota pré-estabelecida, na qual arrastam um grupo de pessoas aos locais que acham mais relevantes. Nestes grupos, cada pessoa tem as suas preferências que muitas vezes não são refletidas nas rotas criadas por estas entidades. Assim, podem acabar por visitar sítios que não sejam do seu interesse e deixar de fora outros mais apelativos.

Tendo este problema como ponto de partida, pretende-se criar um sistema em que cada pessoa sinta que a sua experiência turística é única e personalizada. Este sistema identifica-se como uma aplicação mobile que permita a criação de uma rota turística numa cidade, através de técnicas de *Machine Learning*, tendo por base o perfil de cada utilizador.

A utilização de sistemas de *Machine Learning* tem vindo a apresentar cada vez melhores resultados em diversas áreas de estudo. Estas melhorias devem-se em grande parte aos avanços do *hardware* mas também ao crescente interesse das comunidades de investigação desta área. Os Algoritmos Genéticos (*Genetic Algorithms - GA*) enquadram-se neste grupo de técnicas, mais propriamente na área de otimização, e utilizam conceitos da biologia relacionados com o cruzamento e mutações genéticas, bem como a evolução das espécies e seleção natural introduzida por *Charles Darwin*. Deste modo foram utilizados GA e outras heurísticas como forma de otimização da rota turística.

De forma a tirar partido do sistema desenvolvido, cada utilizador deve inserir na plataforma criada as informações necessárias para a realização de uma viagem e tendo por base estas informações são adquiridos um conjunto de pontos que serão passados a um GA para a seleção daqueles que melhor se enquadram no caso em questão, devolvendo uma rota turística.

Deste modo, neste documento, na secção 2 é apresentado o estado de arte em aplicações referentes ao turismo e ao planeamento de rotas turísticas. De seguida, são apresentados

os requisitos de sistema na secção 3. Na secção 4 mostram-se as decisões relativas aos dados utilizados pelo sistema e na secção 5 a sua arquitetura. Por fim, na secção 6 encontra-se o resultado final do projeto com a aplicação desenvolvida.

2 Estado de Arte

Com a evolução tecnológica nas condições de mobilidade e no acesso a informação, surgiu um aumento do turismo a nível mundial. Como se pode verificar na figura 1, de 2002 a 2017 o número de hóspedes em estabelecimentos hoteleiros portugueses duplicou, refletindo o crescente interesse pela atividade turística. No final de 2017 foram registados mais de 20.6 milhões de turistas hospedados em Portugal.

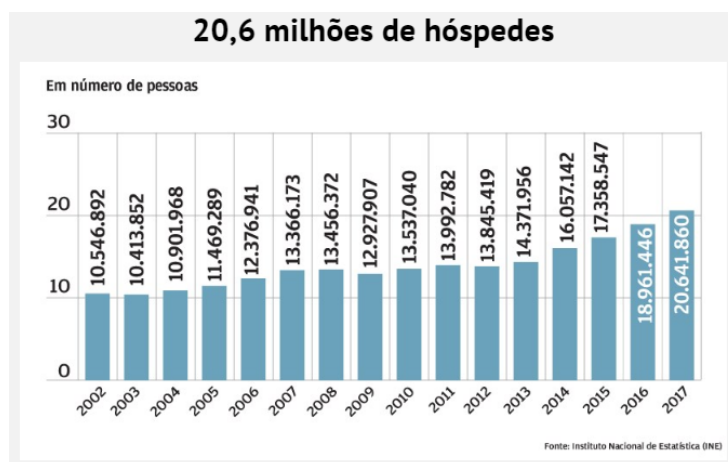


Figura 1: Número de hóspedes em estabelecimentos hoteleiros portugueses.

Indo de encontro a este crescimento, entra-se agora numa nova era onde o turismo passa a ser planeado através de buscas na *Web*, ao invés das típicas agências de viagem ou excursões. Muitas aplicações oferecem propostas para o planeamento e organização de uma viagem. As que mais se destacam neste campo são as seguintes:

- [Google Trips](#) - Esta aplicação permite planear uma viagem com antecedência, apresentando ao utilizador os pontos mais famosos, deixando-o escolher aqueles que pretende visitar. É ainda possível guardar viagens planeadas para mais tarde serem efetuadas.
- [Save Trip](#) - Permite ao utilizador planear uma viagem com antecedência, sendo este responsável pela escolha de todos os pontos a visitar. Não são apresentadas sugestões de locais a visitar.

- [Visit A City](#) - Apresenta sugestões de atividades e itinerários pré-feitos para o utilizador. O utilizador pode visualizar os principais pontos turísticos num mapa.

Apesar da existência de uma grande variedade de aplicações ligadas ao planeamento e organização de uma viagem, estas apresentam ainda bastantes limitações, especialmente no desenvolvimento de um plano de viagem de acordo com o perfil do utilizador, existindo um grande défice neste tipo de habilidade. O objetivo principal do desenvolvimento deste projeto é a combinação do tipo de características já existentes no mercado, como a possibilidade de efetuar rotas pré-feitas e a visualização do percurso em mapas, com a exploração da criação de rotas personalizadas e direcionadas para os gostos individuais de cada utilizador específico.

3 Requisitos

De modo a elaborar uma aplicação que vá de encontro às características pretendidas é necessário elaborar o levantamento de requisitos. Estes são os guias das funcionalidades que o sistema deve oferecer aos seus utilizadores e como tal são uma das partes mais importantes da fase de modelação de qualquer projeto.

De forma sumariada, os requisitos associados à aplicação foram os seguintes:

1. Adquirir as preferências dos utilizadores e atualizá-las de acordo com as suas interações;
2. Permitir adicionar rotas pré-feitas;
3. Elaborar um plano turístico de acordo com as preferências do utilizador:
 - (a) Adquirir as características da viagem (cidade a visitar, duração, dinheiro, refeições);
 - (b) Apresentar os pontos de interesse mais populares segundo as características inseridas;
 - (c) Traçar a rota de acordo com os parâmetros definidos, tendo em conta a duração em cada um dos pontos a visitar;
4. Apresentar no mapa os pontos de cada viagem e permitir a sua navegação através dos mapas nativos do dispositivo;
5. Apresentar no mapa todos os pontos de uma cidade, por categoria;
6. Guardar uma viagem para mais tarde realizar;
7. Eliminar/Terminar viagens guardadas;
8. Pedir informações ao utilizador de cada ponto visitado (preços, tempo e classificação);

A implementação do sistema é feita com base nestes requisitos.

4 Análise e Armazenamento de Dados

Para dar suporte a um sistema que seja capaz de responder a pedidos sobre pontos de interesse de uma determinada zona é necessário que os dados utilizados sejam das áreas de geolocalização e turismo. No entanto, para estas áreas, não existe disponível nenhum conjunto de dados aberto que contenha todos os atributos exigidos para a concretização do trabalho proposto. Como tal, todos os dados utilizados tiveram de ser recolhidos e posteriormente processados.

4.1 Descrição do conjunto de dados

Para conjunto de dados utilizado é exigida a presença dos seguintes atributos: ID do POI, Nome do POI, Latitude, Longitude, Tag, Classificação, Preço e Duração.

Para a obtenção destes dados recorreu-se à ferramenta *OpenStreetMap* uma vez que é um mapa colaborativo, fornecendo dados de acesso livre a toda a comunidade. Para a sua obtenção foi utilizado o *Overpass*, uma ferramenta que permite a realização de *queries* ao OSM. No entanto, estes dados não contêm informação relativa à classificação, preço e duração do ponto de interesse. Desta forma, e para uma fase de testes inicial, foi necessário forjar os dados, adicionando as seguintes informações a cada local:

- **Classificação** - Classificações de um determinado ponto de interesse dadas por diferentes visitantes do local.
- **Duração** - Durações das visitas a um ponto de interesse, em minutos, dadas por diferentes visitantes.
- **Preço** - Dinheiro despendido por diferentes visitantes, em euros, associado ao ponto de interesse.

Estes dados foram gerados com o recurso a distribuições probabilísticas cujos valores obtidos pretendem representar categorias, simulando observações passadas de outros utilizadores.

4.2 Armazenamento dos dados

O armazenamento dos dados obtidos é assegurado através de uma base de dados *NoSQL* orientada a documentos, o *MongoDB*. Nesta são armazenadas três coleções:

- **Pontos de interesse** - São guardadas todas as informações sobre os pontos de interesse. Sempre que são inseridas novas informações sobre um ponto de interesse são inseridas nesta coleção;
- **Utilizadores** - São guardados os dados relativos aos utilizadores, como as suas preferências, as suas viagens (realizadas e planeadas) ou dados de autenticação.
- **Viagens** - São guardadas todas as viagens pré-feitas e os dados relativos a cada um dos pontos a ela pertencentes.

Em cada viagem são guardadas informações sobre o seu estado (Planeada, Terminada, Eliminada), dinheiro e tempo esperados e os pontos turísticos a visitar.

Optou-se por esta base de dados por ter um tipo de esquema é flexível e permitir uma comunicação eficiente com a estrutura da aplicação.

5 Arquitetura da Solução Proposta

A arquitetura do sistema é dividida em duas partes distintas. A estrutura externa corresponde à parte da aplicação que utiliza as funcionalidades admitidas pelo *backend* e efetua a comunicação com o utilizador através do *frontend*, apresentando de uma forma organizada e funcional a informação necessária. A estrutura interna é correspondente à organização do *backend* e ao processamento que é efetuado para a obtenção de um conjunto de pontos que satisfaça os desejos do utilizador.

5.1 Estrutura externa

A estrutura externa do sistema engloba o fluxo de dados, quer de entrada quer de saída do sistema. Esta parte do sistema é desenvolvida em *NodeJS* e *React-Native*. Os dados recebidos pelo utilizador são convertidos em formato *JSON* para a comunicação com os componentes da estrutura interna e base de dados.

Para o planeamento de uma viagem são necessárias as preferências do utilizador, que já se encontram armazenadas na base de dados, e os seguintes elementos:

- **Cidade** - Cidade a visitar;
- **Número de dias** - Número de dias que o utilizador pretende disponibilizar para completar o roteiro turístico;
- **Hora inicial** - Hora em que o utilizador pretende iniciar o roteiro turístico;
- **Hora final** - Hora em que o utilizador pretende terminar o roteiro turístico;
- **Almoço** - Interesse em incluir restaurantes à hora de almoço;
- **Jantar** - Interesse em incluir restaurantes na hora de jantar;
- **Custo** - Orçamento máximo para a realização da visita. O somatório dos preços de cada ponto não deve exceder este valor;

5.2 Estrutura interna

Após a obtenção das informações para a realização da viagem é desencadeado um conjunto de ações que culminam na produção da melhor rota:

1. **Pedido à Base de Dados** - É realizado um pedido à base de dados que filtra os pontos pela cidade pretendida.
2. **Tratamento dos dados** - Os dados são divididos em 2 partes, referentes a pontos turísticos propriamente ditos e estabelecimentos de restauração. Esta separação permite ter em conta as necessidades do utilizador no que toca às suas refeições.
3. **Algoritmo Genético** - O Algoritmo Genético otimiza o conjunto de pontos turísticos a retornar.
4. **Heurística de seleção de restaurantes** - Após selecionados os pontos turísticos a visitar, através do Algoritmo Genético, é realizada uma heurística que seleciona o melhor restaurante para almoçar e/ou jantar, tendo em conta a rota selecionada.

A figura 2 representa a estrutura interna do sistema.

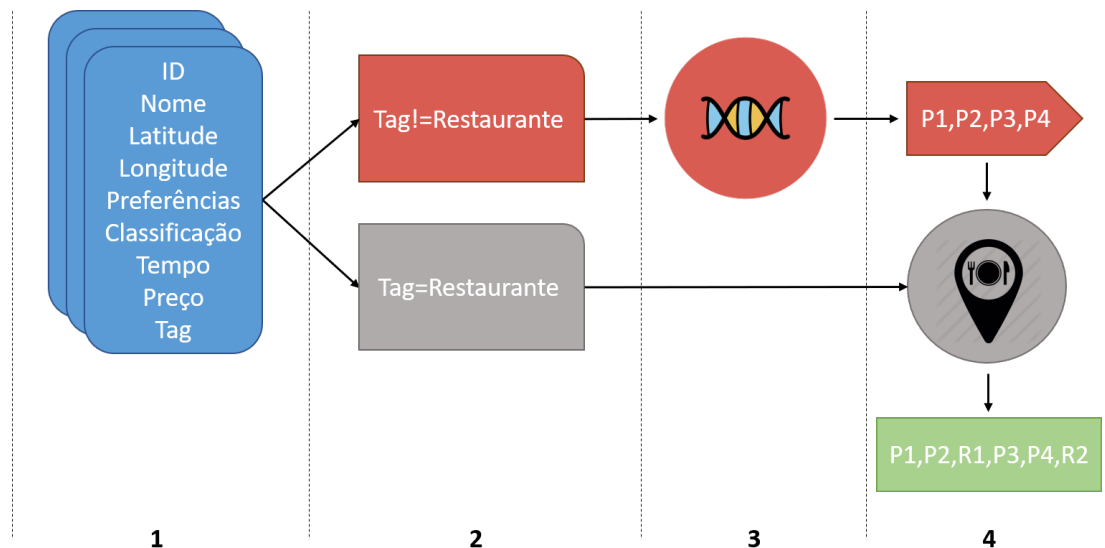


Figura 2: Estrutura interna do sistema.

5.2.1 Algoritmo Genético

Para a utilização desta técnica é necessário representar o problema num cromossoma, onde cada gene representa uma característica do problema. Neste caso, existem $N + 1$ genes, onde N é o número máximo de pontos que uma rota pode apresentar. O gene adicional determina o número de pontos a ter em conta para uma determinada rota. Assim, o cromossoma $\boxed{3 \mid X \mid Y \mid Z \mid W}$ mostra um indivíduo da população que representa uma rota com 3 pontos (X, Y, Z) e onde o ponto W não é considerado (podendo no entanto entrar no processo de cruzamento). Os valores possíveis para cada um dos genes são: um valor inteiro no intervalo $[1, N]$ para o primeiro gene; um valor inteiro representativo do índice de um ponto no *dataframe* para cada um dos restantes genes. De notar que os índices iniciais são escolhidos sem reposição, ou seja, não existem pontos repetidos na população inicial.

Para calcular o *fitness* de cada indivíduo da população, é utilizada a seguinte função:

$$\begin{aligned} & T_{factor} * |T_{available} - \sum (T_{visit} + T_{path})| \\ & + M_{factor} * |M_{available} - \sum M_{visit}| \\ & - C_{factor} * \sum C_{visit} \\ & - P_{factor} * \sum P_{visita} \end{aligned}$$

onde,

T : Tempo (*Time*)

M : Dinheiro (*Money*)

C : Classificação (*Classification*)

P : Preferências (*Preferences*)

As informações necessárias a esta função são obtidas de acordo com os índices e a respetiva associação no *dataframe*. Posteriormente são escolhidos os indivíduos que apresentam os valores de *fitness* mais baixos para o cruzamento. O cruzamento é efetuado nos pontos turísticos de forma aleatória, havendo sempre o cuidado de não efetuar trocas

se algum dos pontos já se encontrar no cromossoma. Como resultado do cruzamento, existe um aumento no tamanho da população que é compensado com um envelhecimento individual e a partir do qual, se após algumas gerações o indivíduo não se encontrar entre os melhores, será eliminado.

No que diz respeito à mutação, esta pode ocorrer a 2 níveis: no primeiro gene, havendo uma alteração do número de pontos a considerar dentro do limite máximo; nos restantes genes, na ordem dos pontos, existindo a possibilidade da entrada/saída de pontos da rota.

Como resultado final deste algoritmo é obtida uma rota constituída por qualquer tipo de ponto turístico exceto restaurantes.

5.2.2 Heurística de seleção de restaurantes

Quando o utilizador demonstra o interesse em ter presentes na sua rota estabelecimentos para a realização das suas refeições é importante a existência de medidas que consigam escolher entre os restaurantes disponíveis. Nestes casos, após a concretização do algoritmo genético é utilizada uma heurística que tem como propósito a obtenção desses pontos.

O objetivo é colocar o almoço num horário próximo do intervalo 12h-14h e o jantar numa hora que se aproxime do intervalo 19h-21h. Para isto, tendo em conta a hora de início de percurso são calculados os pontos que são visitados antes e depois da refeição em causa.

De seguida cada estabelecimento é classificado de acordo com o seguinte conjuntos de fatores:

- **Distância aos pontos** que antecedem e sucedem o horário de refeição. Pretendem-se encontrar estabelecimentos que não afastem demasiado o utilizador da sua rota.
- **Classificação** do restaurante fornecido por outros utilizadores.
- **Preço** médio gasto no estabelecimento. Permite o controlo dos gastos do utilizador.

- **Preferências do utilizador** em relação à alimentação.
- **Tempo de permanência** no estabelecimento, evitando que exceda o tempo disponível pelo utilizador.

Após a classificação de todos os pontos com a ponderação destes fatores é escolhido o que apresenta maior *score* para cada uma das refeições selecionadas.

6 Aplicação Desenvolvida

Nesta secção são apresentadas as ferramentas utilizadas para o desenvolvimento do sistema bem como o resultado final obtido, com a aplicação funcional.

6.1 Ferramentas Utilizadas

Para o desenvolvimento do sistema foram utilizadas diferentes ferramentas de acordo com as funcionalidades a ser implementadas. Para o *frontend* e a computação *client-side* foi utilizada uma *framework* de JavaScript, o *React-Native*. Para além da curva de aprendizagem acessível, esta *framework* é utilizada em larga escala no desenvolvimento de aplicações, existindo uma forte comunidade de apoio e um vasto conjunto de bibliotecas que facilitam a implementação de muitas funcionalidades.

Para a parte do servidor, o NodeJS foi a *framework* escolhida, pela compatibilidade com o *frontend* e a facilidade de conexão com o tipo de BD escolhida (MongoDB). Da mesma forma, as bibliotecas existentes facilitam o trabalho de desenvolvimento. Ainda no lado do servidor, o processo do algoritmo genético foi efetuado em *Python* pela facilidade de manipulação de dados fornecido por esta linguagem e pela sua simplicidade de uso. A comunicação entre estes componentes é efetuada através da troca de mensagens JSON.

Os dados relativos aos pontos turísticos foram adquiridos através da utilização do *Overpass*, uma ferramenta que permite efetuar *queries* ao *OpenStreetMap*.

6.2 Resultado Obtido

Como resultado do trabalho realizado, apresenta-se uma aplicação *mobile*, compatível com *Android* e *IOS*, que conhece o utilizador, se adapta às suas escolhas e lhe apresenta percursos turísticos que vão de encontro aos seus interesses. De seguida são apresentadas as funcionalidades implementadas.

- **Obtenção de preferências**

Aquando do registo de um utilizador, para além das informações básicas (nome, email e password), e sendo este o primeiro contacto com o utilizador, torna-se necessário obter informação sobre as suas preferências. Assim, após o registo, são apresentadas ao utilizador um conjunto de categorias de preferências das quais ele deve seleccionar no máximo 10. Estas preferências podem ser atualizadas mais tarde no seu perfil.

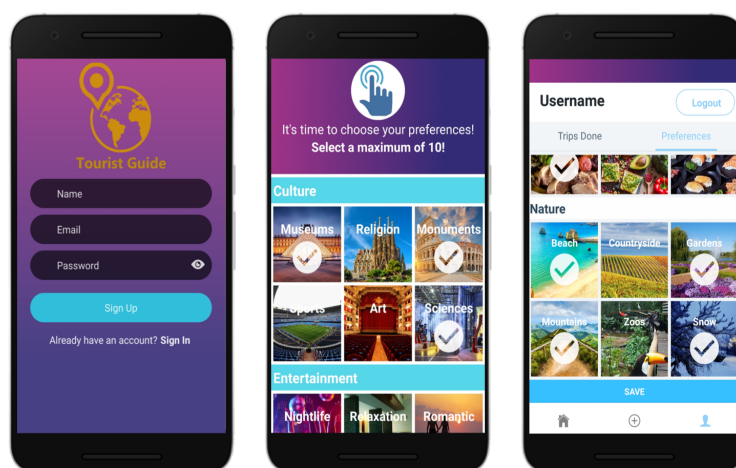


Figura 3: Registo, obtenção e atualização das preferências.

- **Lista de Viagens**

São apresentadas ao utilizador todas as viagens que ele adicionou à sua lista de viagens ainda por realizar, permitindo-lhe aceder às suas informações (número de dias da viagem, preço e a rota para cada um dos dias) e eventualmente iniciar o percurso turístico. São-lhe ainda sugeridas novas viagens que já se encontram pré-feitas que podem ser adicionadas à lista de viagens por realizar.

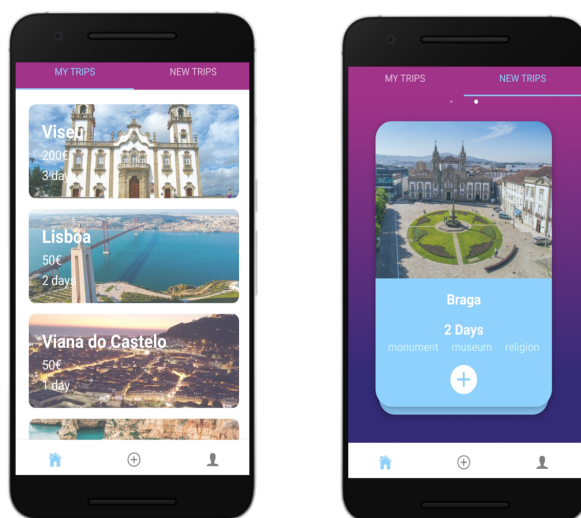


Figura 4: Página Inicial.

• Planear Viagem

O planeador automático de viagens é a ferramenta diferenciadora relativamente aos sistemas existentes no mercado, uma vez que é neste componente que o utilizador tem a possibilidade de efetuar uma rota de acordo com as suas preferências.

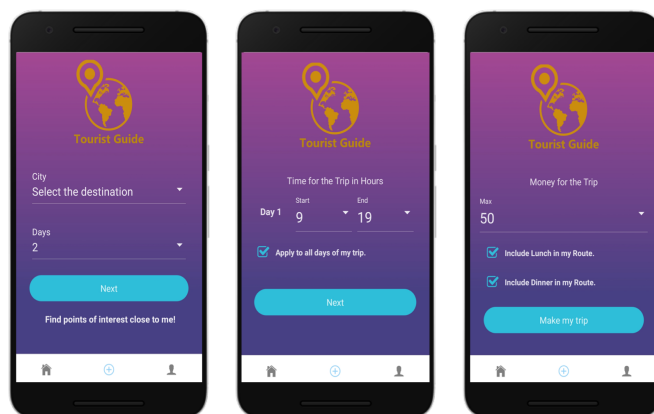


Figura 5: Planear uma viagem.

• Realizar Viagem

Sempre que pretender, o utilizador pode visualizar as informações sobre uma viagem guardada, mostrando os respetivos pontos de interesse e sendo redirecionado para os serviços de mapas do seu telemóvel para ser guiado pela rota correspondente.

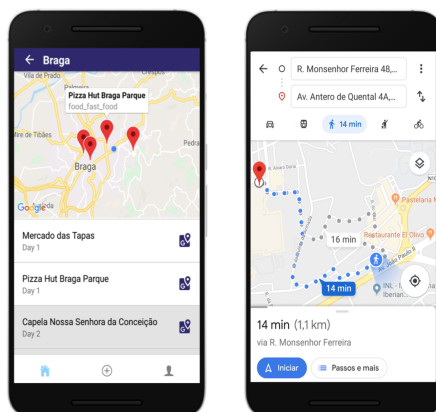


Figura 6: Realizar viagem.

- **Terminar Viagem**

Quando o utilizador termina a sua rota é apresentado um questionário, permitindo-lhe avaliar cada ponto onde passou e dar informações sobre o dinheiro e tempo que cada um requerem. Estas informações serão utilizadas como forma de melhorar o funcionamento futuro da aplicação e apenas são apresentadas numa fase inicial, até existirem dados suficientes. Nesta fase este questionário será retirado.

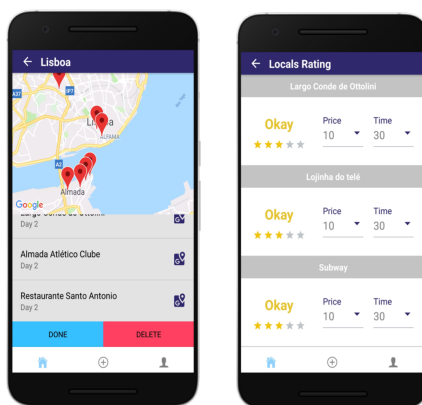


Figura 7: Terminar viagem.

- **Apresentar Pontos de Interesse**

De acordo com a cidade seleccionada, são apresentados todos os pontos de interesse existentes na base de dados para essa cidade, filtrados pela categoria desejada.

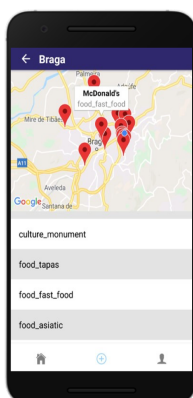


Figura 8: Apresentar Pontos de Interesse.

7 Conclusão

A utilização de algoritmos genéticos para a resolução do problema proposto demonstrou-se satisfatória, tendo em conta os resultados que esta abordagem é capaz de apresentar. O problema em questão exibe fortes semelhanças com o problema da mochila (*knapsack problem* [5]), no entanto a sua complexidade é muito superior devido à existência de 2 medidas distintas de custo/peso (tempo e dinheiro) e de 2 medidas distintas de ganho (preferências e classificação). Deste modo, o algoritmo genético representa uma forma de contornar a complexidade exigida e apresentar bons resultados.

Para a inserção dos restaurantes na rota de acordo com o número de refeições, foram pensadas várias soluções entre as quais a resolução de um problema de roteamento de veículos com janelas temporais [2]. Pela complexidade do algoritmo apresentado, foi utilizada uma heurística simplificada que coloca os restaurantes no tempo correto.

O desempenho destas duas técnicas são no entanto muito difíceis de medir com os dados disponíveis. De facto, os dados adquiridos são em pouca quantidade, incertos e o facto de muitos deles serem forjados dificulta ainda mais a obtenção de métricas de qualidade.

Como trabalho futuro perspectiva-se o melhoramento do algoritmo genético, a inserção de novas funcionalidades na aplicação, como por exemplo obter os pontos de interesse próximos da localização de um utilizador que mais se aproximem do perfil do mesmo, e a procura de novas formas de obtenção de dados.

Referências

- [1] Turismo em Portugal: http://www.turismodeportugal.pt/pt/Turismo_Portugal/visao_geral/Paginas/default.aspx
- [2] Brian Kallehauge, Jesper Larsen, Oli B.G. Madsen and Marius M. Solomon VEHICLE ROUTING PROBLEM WITH TIME WINDOWS
- [3] Dados do OSM: <http://download.geofabrik.de/europe.html>
- [4] Easy Access to All Points of Interest Data: <https://medium.com/codait/easy-access-to-all-points-of-interest-data-acc6569e45b2>
- [5] Silvano Martello and Paolo Toth, DEIS, University of Bologna, KNAPSACK PROBLEMS Algorithms and Computer Implementations