

Junho de 2019

Relatório Final

Projeto Temático em Aplicações SIG

LICENCIATURA EM
TECNOLOGIAS DA INFORMAÇÃO

RELATÓRIO FINAL NO ÂMBITO DA UNIDADE CURRICULAR
PROJETO TEMÁTICO EM APLICAÇÕES SIG

Autores:

Daniel Martins – nº 83645
Jorge Anjos – nº 81048
Pedro Ferreira – nº 81861
Ricardo Balreira – nº 88078
Rui Duarte – nº 22979

Professores:

Ana Calvão
António Barbeito
Jorge Gonçalves

Junho de 2019

Relatório Final

Projeto Temático em Aplicações SIG

LICENCIATURA EM
TECNOLOGIAS DA INFORMAÇÃO

RELATÓRIO FINAL NO ÂMBITO DA UNIDADE CURRICULAR
PROJETO TEMÁTICO EM APLICAÇÕES SIG

Autores:

Daniel Martins – nº 83645
Jorge Anjos – nº 81048
Pedro Ferreira – nº 81861
Ricardo Balreira – nº 88078
Rui Duarte – nº 22979

Professores:

Ana Calvão
António Barbeito
Jorge Gonçalves

Agradecimentos

Este trabalho é o culminar de um longo processo e resulta do nosso esforço pessoal e do empenho de todos os elementos do grupo, mas o mesmo nunca teria sido realizado com sucesso sem o contributo e apoio de diversas pessoas que, de forma direta ou indireta, nele colaboraram.

Queremos agradecer, de forma muito especial, ao Professor Doutor Luís Jorge dos Santos Gouveia Marques Gonçalves, pela orientação e disponibilidade ao longo deste trabalho. A sabedoria e qualidade dos orientadores na supervisão foram cruciais para a realização deste projeto.

A toda a equipa de docentes deste curso, pelo apoio, por tudo o que nos ensinaram e sobretudo pela metodologia de trabalho que nos incutiram.

A todos eles o nosso muito obrigado.

Resumo

Este projeto visa desenvolver uma aplicação envolvendo a gestão de informação geograficamente referenciada, utilizando dados espaciais e ferramentas que permitem manipular e analisar esses mesmos dados para serem vistos na página Web.

O trabalho tem como principal objetivo desenvolver uma aplicação WebSig cujo foco é visualização, por meio geográfico, da oferta desportiva da região de Aveiro, complementado pela componente desportiva da UA e seus politécnicos. Num ponto de vista mais detalhado, a plataforma permite disponibilizar a informação das infraestruturas desportivas, clubes e equipas, e além disso, a respetiva localização por meio de um mapa. Tendo acesso à componente desportiva, o utilizador vê também pontos de interesse (restaurantes, meios de transporte públicos, cafés, centros comerciais...) que lhe podem ser úteis (por exemplo: um pai leva a filha ao treino e pretende aguardar num café mais próximo enquanto o treino não termina). Com efeito, existe a possibilidade de filtrar a informação consoante a intenção do utilizador, isto é, caso este pretenda ver a componente desportiva apenas numa determinada zona (município e/ou freguesia) ou por determinada condição (por exemplo: modalidade) há a hipótese de restringir a informação que é fornecida no mapa. No contexto de rotas ou trajetos, o utilizador consegue saber o percurso, duração e distância entre dois pontos (por exemplo: entre um pavilhão e um restaurante). No ponto de vista da UA, pela plataforma, o utilizador consegue saber se esta detém de meios de transporte, por exemplo, antes de tomar uma decisão para a facultade a ingressar e a praticar desporto.

Índice geral

1. INTRODUÇÃO	1
2. PLANEAMENTO	3
2.1. PLANO DO PROJETO	3
2.1.1. ESCALONAMENTO DAS TAREFAS	4
2.1.2. GRÁFICO DE GANTT	5
2.2. EXECUTADO	14
2.2.1. O QUE DIFERIU	14
2.2.2. EXECUÇÃO DAS TAREFAS	14
2.2.3. GRÁFICO DE GANTT	16
3. REQUISITOS DO SISTEMA.....	25
3.1. REQUISITOS FUNCIONAIS	25
3.2. REQUISITOS NÃO FUNCIONAIS	26
4. DIAGRAMA DE CASOS DE UTILIZAÇÃO	27
4.1. ATORES	27
4.2. DESCRIÇÃO DOS CASOS DE UTILIZAÇÃO	28
4.2.1. <i>Visualizar mapa</i>	28
4.2.2. <i>Pesquisar informação</i>	31
4.3. DIAGRAMA INTEGRAL	33
5. DIAGRAMA DE CLASSES	35
6. MODELO DE DADOS PERSISTENTES	37
6.1. BASE DE DADOS	37
6.2. SGBD ALVO.....	38
6.3. RESTRIÇÕES (CONSTRAINTS).....	38
6.4. DEFINIÇÃO DE VISTAS (VIEWS)	39
6.5. PROCEDIMENTOS DE EXECUÇÃO AUTOMÁTICA (<i>TRIGGERS</i>)	40
6.6. EXTENSÕES	41
6.7. PROCEDIMENTOS ARMAZENADOS (STORED PROCEDURES)	42
6.7.1. <i>Nó mais próximo</i>	42
6.7.2. <i>Algoritmo Driving Distance</i>	43
6.7.3. <i>Pontos de interesse contidos no polígono</i>	44
7. TECNOLOGIAS USADAS	45

8. CAMADA DE APRESENTAÇÃO	47
8.1. PÁGINA INICIAL.....	48
8.2. PESQUISA MANUAL	49
8.3. FILTRAGEM	51
8.4. CARREGAR NUM PONTO OU RESULTADO DE PESQUISA.....	54
8.5. PONTOS DE INTERESSE.....	57
8.6. CALCULAR ROTAS.....	59
8.7. ALTERNAR A CARTOGRAFIA DE CONTEXTO	60
8.8. TRATAMENTO DE ERROS	61
9. ANÁLISE DE RESULTADOS E REFLEXÃO CRÍTICA.....	63
9.1. REQUISITOS FUNCIONAIS CUMPRIDOS E NÃO CUMPRIDOS.....	64
9.1.1. REQUISITOS NÃO FUNCIONAIS CUMPRIDOS E NÃO CUMPRIDOS.....	64
10. CONCLUSÃO	65
11. BIBLIOGRAFIA	67

Índice de tabelas

TABELA 1 - LISTA DE ATIVIDADES DO GRÁFICO DE GANTT DO PLANEAMENTO (PARTE 1).....	8
TABELA 2 - LISTA DE ATIVIDADES DO GRÁFICO DE GANTT DO PLANEAMENTO (PARTE 2).....	9
TABELA 3 - LISTA DE ATIVIDADES DO GRÁFICO DE GANTT DO PLANEAMENTO (PARTE 3)....	10
TABELA 4 - LISTA DE ATIVIDADES DO GRÁFICO DE GANTT DO PLANEAMENTO (PARTE 4)....	11
TABELA 5 - LISTA DE ATIVIDADES DO GRÁFICO DE GANTT DO PLANEAMENTO (PARTE 5)....	12
TABELA 6 - LISTA DE ATIVIDADES DO GRÁFICO DE GANTT DO PLANEAMENTO (PARTE 6)....	13
TABELA 7 - LISTA DE ATIVIDADES DO GRÁFICO DE GANTT DA EXECUÇÃO (PARTE 1).....	19
TABELA 8 - LISTA DE ATIVIDADES DO GRÁFICO DE GANTT DA EXECUÇÃO (PARTE 2).....	20
TABELA 9 - LISTA DE ATIVIDADES DO GRÁFICO DE GANTT DA EXECUÇÃO (PARTE 3).....	21
TABELA 10 - LISTA DE ATIVIDADES DO GRÁFICO DE GANTT DA EXECUÇÃO (PARTE 4).....	22
TABELA 11 - LISTA DE ATIVIDADES DO GRÁFICO DE GANTT DA EXECUÇÃO (PARTE 5).....	23
TABELA 12 - REQUISITOS FUNCIONAIS (RF)	25
TABELA 13 - REQUISITOS NÃO FUNCIONAIS (RNF)	26
TABELA 14 - ATORES E DESCRIÇÃO DO SEU PAPEL NO SISTEMA	27
TABELA 15 - DESCRIÇÃO DO CASO DE USO "VISUALIZAR MAPA".....	28
TABELA 16 - DESCRIÇÃO DO CASO DE USO " PESQUISAR INFORMAÇÃO"	33
TABELA 17 - REQUISITOS FUNCIONAIS (RF) CUMPRIDOS E NÃO CUMPRIDOS	64
TABELA 18 - REQUISITOS NÃO FUNCIONAIS (RNF) CUMPRIDOS E NÃO CUMPRIDOS.....	64

Índice de figuras

FIGURA 1 - PLANEAMENTO DAS TAREFAS (PARTE 1).....	4
FIGURA 2 - PLANEAMENTO DAS TAREFAS (PARTE 2).....	5
FIGURA 3 - GRÁFICO DE GANTT DO PLANEAMENTO (PARTE 1)	6
FIGURA 4 - GRÁFICO DE GANTT DO PLANEAMENTO (PARTE 2)	7
FIGURA 5 - EXECUÇÃO DAS TAREFAS (PARTE 1)	15
FIGURA 6 - EXECUÇÃO DAS TAREFAS (PARTE 2)	16
FIGURA 7 - GRÁFICO DE GANTT DA EXECUÇÃO (PARTE 1)	17
FIGURA 8 - GRÁFICO DE GANTT DA EXECUÇÃO (PARTE 2)	18
FIGURA 9 - MOCKUP DE VISUALIZAR O MAPA COM DESCRIÇÃO E FILTRAGEM DA COMPONENTE DESPORTIVA.....	29
FIGURA 10 - MOCKUP COM UMA DESCRIÇÃO DETALHADA DE UM PONTO SELECIONADO.....	30
FIGURA 11 - MOCKUP COM PONTOS DE INTERESSE AO CLICAR NUM PONTO DO MAPA.....	30
FIGURA 12 - MOCKUP COM FILTRAGEM DOS PONTOS DO MAPA	31
FIGURA 13 - FILTRAGEM DOS PONTOS VISTO COM MAIOR DETALHE	32
FIGURA 14 - MOCKUP COM ITINERÁRIO AO SELECIONAR O BOTÃO DAS DIREÇÕES	32
FIGURA 15 - DIAGRAMA DE CASOS DE UTILIZAÇÃO	34
FIGURA 16 - DIAGRAMA DE CLASSES	35
FIGURA 17 - DIAGRAMA LÓGICO DA BASE DE DADOS	37
FIGURA 18 - VISTAS.....	39
FIGURA 19 - RESULTADO DA FUNÇÃO SELECT_ALL_TRIGGER()	41
FIGURA 20 - RESULTADO DA EXECUÇÃO DA FUNÇÃO ST_UNION.....	43
FIGURA 21 - RESULTADO UTILIZANDO A FUNÇÃO ST_CONCAVEHULL.....	43
FIGURA 22 - RESULTADO UTILIZANDO A FUNÇÃO ST_CONVEXHULL.....	43
FIGURA 23 - APRESENTAÇÃO DA PÁGINA INICIAL	48
FIGURA 24 - APRESENTAÇÃO DA PESQUISA DE UMA INFRAESTRUTURA.....	49
FIGURA 25 - APRESENTAÇÃO DA PESQUISA DE UM CLUBE	50
FIGURA 26 - FILTRAGEM DAS INFRAESTRUTURAS.....	52
FIGURA 27 - FILTRAGEM DOS CLUBES/UNIVERSIDADE	53
FIGURA 28 - POPUP COM INFORMAÇÃO RESUMIDA	54
FIGURA 29 - DETALHES DA INFRAESTRUTURA SELECIONADA	55
FIGURA 30 - DETALHES DO CLUBE SELECIONADO	56
FIGURA 31 - PONTOS DE INTERESSE QUE SURGEM AO SELECIONAR A INFRAESTRUTURA ...	57
FIGURA 32 - POPUP QUANDO SE SELECCIONA UM PONTO DE INTERESSE	58
FIGURA 33 - CÁLCULO DA ROTA AO SELECIONAR DOIS PONTOS NO MAPA	59

FIGURA 34 - VISTA SATÉLITE 60

FIGURA 35 - MENSAGEM DE ERRO QUANDO OCORRE UM ERRO DE COMUNICAÇÃO 61

Lista de siglas e abreviaturas

SIG	Sistemas de Informação Geográfica
HTML	HyperLink Text Markup Language
CSS	Cascading Style Sheets
PHP	HyperText Preprocessor
SQL	Structured Query Language
PL/pgSQL	Procedural Language/PostgreSQL
SGBD	Sistema de Gestão de Base de Dados
AJAX	Asynchronous JavaScript and XML
JSON	JavaScript Object Notation
WKT	Well-Known text
SRID	Spatial Reference System Identifier
Ref ^a	Referência
GeoJSON	Geographic JavaScript Object Notation
ol-ext	OpenLayers Extensions
GML	Geographic Markup Language
POI	Pontos de Interesse

1. Introdução

No âmbito do módulo temático em Aplicações SIG, o qual engloba, para além do Projeto Temático em Aplicações SIG, a Unidade Curricular associada à disciplina Sistemas de Informação Geográfica, foi proposto elaborar uma aplicação que permitisse guardar, analisar, manipular, capturar e visualizar informação geográfica com o fundamento que seja apresentado numa plataforma Web.

Desta forma, o tema escolhido englobava a implementação de uma aplicação WebSIG que representasse geograficamente a componente desportiva da região de Aveiro e da Universidade de Aveiro e seus politécnicos. Possibilita, assim, que o utilizador encontre toda a informação pretendida de cada clube, equipa e das infraestruturas desportivas que existem na região de Aveiro. Restringindo a informação que é vista no mapa, este pode filtrar por várias condições: município (Ílhavo, Ovar, Estarreja...), freguesia (Borralha, Angeja, Recardães...), modalidade (futebol, basquetebol, andebol...), escalão (juniors, seniors), género (masculino e/ou feminino), se o campo é ou não coberto, se se pratica ou não desporto adaptado e preço.

Como foi possível ver, há um conjunto de critérios importantes que enriquecem a aplicação e que tornam uma simples ideia numa plataforma muito útil e única, principalmente para os atletas apaixonados pelo desporto e que procuram um clube próximo da sua residência. Além destes aspetos, o utilizador tem ao seu dispor um itinerário que indica o trajeto, a duração e a distância entre dois pontos inseridos pelo mesmo, consoante o meio de transporte (a pé, de carro ou de bicicleta).

Neste relatório serão debatidos e abordados os seguintes aspectos: descrição do planeamento das tarefas pelos elementos do grupo e respetiva execução; consolidação dos requisitos funcionais e não funcionais que descrevem o sistema; modelação do sistema por meio de um diagrama de casos de utilização e de classes; desenho e conceção do protótipo de baixa fidelidade; definição do modelo de dados persistentes; realização da camada de apresentação com a descrição das atividades desenvolvidas; por fim, uma conclusão onde haverá a oportunidade de referir a opinião pessoal sobre o trabalho, adversidades surgidas e aspetos mais conseguidos.

2. Planeamento

Contemplando a atividade de gestão do projeto onde é uma fase contínua e importante em todo o percurso do trabalho e influencia a forma como o projeto termina, se da melhor ou da pior maneira, fez-se assim um escalonamento das tarefas e divisão de trabalhos entre os elementos do grupo. Como se sabe, nesta etapa é de enorme importância estabelecer um planeamento ou, por outras palavras, uma planificação das tarefas que se vão realizar futuramente e, no final, conceber a execução real (como realmente o projeto foi feito).

2.1. Plano do projeto

Antes de partir para o escalonamento das tarefas é necessário identificar e descrever as tarefas a realizar. Com isto, tendo em conta o contexto do trabalho, há que considerar o seguinte: divisão das tarefas, documentação do trabalho por meio de um relatório, levantamento dos requisitos funcionais e não funcionais, modelação do sistema por meio de um diagrama de casos de utilização, diagrama de classes e modelo lógico, protótipos de alta e baixa fidelidade e, ainda, a concretização da base de dados.

Uma vez que consiste num trabalho de grupo e na utilização de tecnologias Web, ferramentas que manipulam dados de informação geográfica e também o armazenamento e gestão dos dados persistentes, é importante assim definir primeiro as tarefas, e, em seguida, coloca-las em prática. Dentro da base de dados existem as tabelas, *views*, *indexes*, *triggers*, funções em SQL ou PL/pgSQL e *check constraints* que é preciso implementar. Relativamente à camada de apresentação, utilizar-se-ão as tecnologias Web (HTML, CSS, JavaScript e PHP). No ponto de vista da funcionalidade da aplicação necessitou-se de dividir tarefas para o seguinte: apresentar e filtrar os pontos no mapa, *popup* que surge com informação do ponto selecionado, calcular rota entre dois pontos, pesquisa avançada, mudar o estilo do mapa (mapa ou satélite) e localização atual. De realçar também que o comparecer às reuniões, redação das atas, criação do grupo na plataforma codeUA e relatório modelo também estão incluídas como tarefas.

2.1.1. Escalonamento das tarefas

As figuras 1 e 2 dizem respeito ao escalonamento das tarefas, isto é, a execução ideal do trabalho. Estão constadas as tarefas, bem como a respetiva duração e a data de início e conclusão. As tarefas foram divididas em duas imagens (figuras 1 e 2) para ser possível ver na íntegra.

ID	Nome da Tarefa	Duração	Inicio	Conclusão
1	Reuniões	75,13 dias	Sex 22/02/19	Sex 07/06/19
2	Reuniões 1	1 hr	Sex 22/02/19	Sex 22/02/19
3	Reuniões 2	1 hr	Sex 01/03/19	Sex 01/03/19
4	Reuniões 3	1 hr	Sex 08/03/19	Sex 08/03/19
5	Reuniões 4	1 hr	Sex 15/03/19	Sex 15/03/19
6	Reuniões 5	1 hr	Sex 22/03/19	Sex 22/03/19
7	Reuniões 6	1 hr	Sex 29/03/19	Sex 29/03/19
8	Reuniões 7	1 hr	Sex 05/04/19	Sex 05/04/19
9	Reuniões 8	1 hr	Sex 12/04/19	Sex 12/04/19
10	Reuniões 9	1 hr	Sex 19/04/19	Sex 19/04/19
11	Reuniões 10	1 hr	Sex 26/04/19	Sex 26/04/19
12	Reuniões 11	1 hr	Sex 03/05/19	Sex 03/05/19
13	Reuniões 12	1 hr	Sex 10/05/19	Sex 10/05/19
14	Reuniões 13	1 hr	Sex 17/05/19	Sex 17/05/19
15	Reuniões 14	1 hr	Sex 24/05/19	Sex 24/05/19
16	Reuniões 15	1 hr	Sex 31/05/19	Sex 31/05/19
17	Reuniões 16	1 hr	Sex 07/06/19	Sex 07/06/19
18	Atas	75,02 dias	Sex 22/02/19	Sex 07/06/19
19	Atas 1	10 mins	Sex 22/02/19	Sex 22/02/19
20	Atas 2	10 mins	Sex 01/03/19	Sex 01/03/19
21	Atas 3	10 mins	Sex 08/03/19	Sex 08/03/19
22	Atas 4	10 mins	Sex 15/03/19	Sex 15/03/19
23	Atas 5	10 mins	Sex 22/03/19	Sex 22/03/19
24	Atas 6	10 mins	Sex 29/03/19	Sex 29/03/19
25	Atas 7	10 mins	Sex 05/04/19	Sex 05/04/19
26	Atas 8	10 mins	Sex 12/04/19	Sex 12/04/19
27	Atas 9	10 mins	Sex 19/04/19	Sex 19/04/19
28	Atas 10	10 mins	Sex 26/04/19	Sex 26/04/19
29	Atas 11	10 mins	Sex 03/05/19	Sex 03/05/19
30	Atas 12	10 mins	Sex 10/05/19	Sex 10/05/19
31	Atas 13	10 mins	Sex 17/05/19	Sex 17/05/19
32	Atas 14	10 mins	Sex 24/05/19	Sex 24/05/19
33	Atas 15	10 mins	Sex 31/05/19	Sex 31/05/19
34	Atas 16	10 mins	Sex 07/06/19	Sex 07/06/19

Figura 1 - Planeamento das tarefas (parte 1)

ID	Nome da Tarefa	Duração	Inicio	Conclusão
35	Gestão de Projeto	5,15 dias	Qua 20/02/19	Qui 28/02/19
36	Criar grupo no codeUA	10 mins	Qua 20/02/19	Qua 20/02/19
37	Relatório Modelo	1 hr	Qui 21/02/19	Qui 21/02/19
38	Divisão de tarefas	5 dias	Qui 21/02/19	Qui 28/02/19
39	Relatório	71 dias	Sáb 02/03/19	Seg 10/06/19
40	Relatório Final	72 dias	Sáb 02/03/19	Seg 10/06/19
41	Planeamento	20 dias	Qua 06/02/19	Ter 05/03/19
42	Requisitos funcionais e não funcionais	3 dias	Sex 01/03/19	Ter 05/03/19
43	Definir utilizadores do sistema	2 dias	Qua 06/02/19	Qui 07/02/19
44	Modelação	8 dias	Qui 14/03/19	Seg 25/03/19
45	Diagrama de casos de utilização	1 dia	Sex 08/03/19	Dom 10/03/19
46	Diagrama de classes	2 dias	Qui 14/03/19	Sex 15/03/19
47	Modelo de dados persistentes	2 dias	Qui 14/03/19	Sex 15/03/19
48	Prototipagem	10 dias	Seg 18/03/19	Sex 29/03/19
49	Protótipo baixa fidelidade	10 dias	Seg 18/03/19	Sex 29/03/19
50	Base de dados	2 dias	Seg 01/04/19	Ter 02/04/19
51	Criar tabelas	10 mins	Seg 01/04/19	Seg 01/04/19
52	Criar views	1 dia	Ter 02/04/19	Ter 02/04/19
53	Criar triggers	10 mins	Ter 02/04/19	Ter 02/04/19
54	Camada de apresentação	32 dias	Sex 05/04/19	Seg 20/05/19
55	Estrutura da página	5 dias	Sex 05/04/19	Qui 11/04/19
56	Mostrar pontos no mapa	3 dias	Sex 12/04/19	Ter 16/04/19
57	Popup com informação	2 dias	Qui 18/04/19	Sex 19/04/19
58	Detalhes dos pontos	1 dia	Seg 22/04/19	Seg 22/04/19
59	Filtrar dados	4 dias	Ter 23/04/19	Sex 26/04/19
60	Calcular rotas	8 dias	Seg 29/04/19	Qua 08/05/19
61	Obter geolocalização	1 dia	Qui 09/05/19	Qui 09/05/19
62	Mudar estilo do mapa	1 dia	Sex 10/05/19	Sex 10/05/19
63	Mostrar pontos de interesse	2 dias	Seg 13/05/19	Ter 14/05/19
64	Motor de busca	4 dias	Qua 15/05/19	Seg 20/05/19
65	Defesa do trabalho	8 dias	Seg 17/06/19	Qua 26/06/19
66	Preparar apresentação	5 dias	Seg 17/06/19	Sex 21/06/19
67	Dia da apresentação	1 dia	Qua 26/06/19	Qua 26/06/19

Figura 2 - Planeamento das tarefas (parte 2)

2.1.2. Gráfico de Gantt

Uma vez que já se sabem as tarefas, as figuras 3 e 4 complementam as tarefas acima indicadas com a inclusão do gráfico de Gantt. Separou-se por duas figuras o gráfico de Gantt pelo mesmo motivo de ser mais visível, logo as figuras 3 e 4 contemplam a primeira e segunda partes do gráfico, respetivamente. Pelas figuras é possível ver a designação para cada tarefa, duração, data de início e conclusão, tal e qual como está retratado nas figuras 1 e 2, mas desta vez, observa-se o gráfico de Gantt com a indicação da percentagem de participação de cada elemento do grupo.

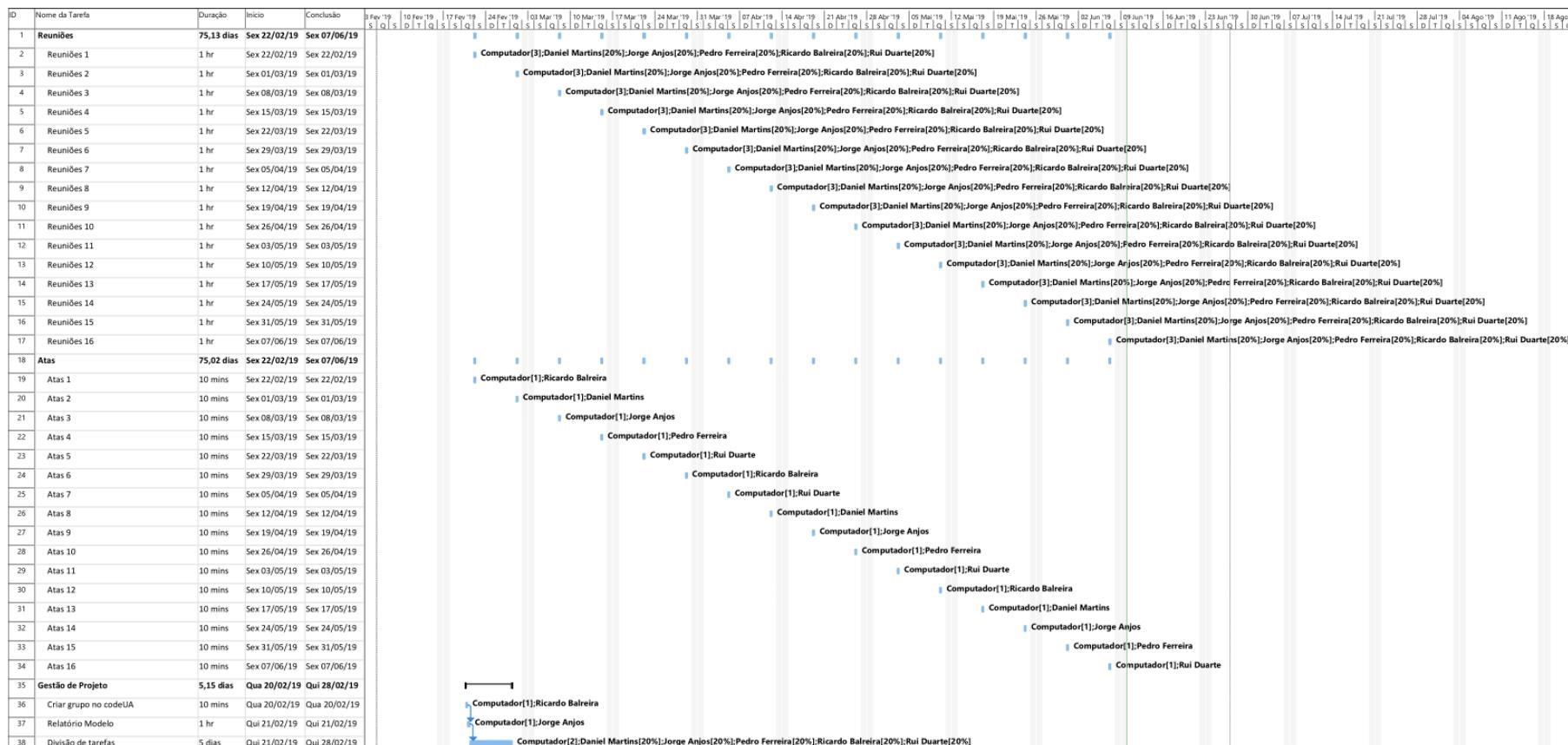


Figura 3 - Gráfico de Gantt do Planeamento (parte 1)



Figura 4 - Gráfico de Gantt do Planeamento (parte 2)

Visto que as figuras acima, que ilustram o gráfico de Gantt, não estão inteiramente legíveis (figuras 3 e 4), as tabelas 1-6 contextualizam a designação de cada tarefa, duração, data de início e conclusão e ainda com a respetiva percentagem de participação de cada elemento do grupo:

Tabela 1 - Lista de atividades do gráfico de Gantt do Planeamento (parte 1)

Nº	Tarefa	Duração	Início	Fim	Percentagem (%)					
1	Reuniões	75,13 dias	22/02/19	07/06/19						
2	Reuniões 1	1 hr	22/02/19	22/02/19	20	20	20	20	20	3
3	Reuniões 2	1 hr	01/03/19	01/03/19	20	20	20	20	20	3
4	Reuniões 3	1 hr	08/03/19	08/03/19	20	20	20	20	20	3
5	Reuniões 4	1 hr	15/03/19	15/03/19	20	20	20	20	20	3
6	Reuniões 5	1 hr	22/03/19	22/03/19	20	20	20	20	20	3
7	Reuniões 6	1 hr	29/04/19	29/04/19	20	20	20	20	20	3
8	Reuniões 7	1 hr	05/04/19	05/04/19	20	20	20	20	20	3
					Daniel Martins	Jorge Anjos	Pedro Ferreira	Ricardo Balreira	Rui Duarte	Computador

Tabela 2 - Lista de atividades do gráfico de Gantt do Planeamento (parte 2)

Nº	Tarefa	Duração	Início	Fim	Percentagem (%)					
9	Reuniões 8	1 hr	12/04/19	12/04/19	20	20	20	20	20	3
10	Reuniões 9	1 hr	19/04/19	19/04/19	20	20	20	20	20	3
11	Reuniões 10	1 hr	26/04/19	26/04/19	20	20	20	20	20	3
12	Reuniões 11	1 hr	03/05/19	03/05/19	20	20	20	20	20	3
13	Reuniões 12	1 hr	10/05/19	10/05/19	20	20	20	20	20	3
14	Reuniões 13	1 hr	17/05/19	17/05/19	20	20	20	20	20	3
15	Reuniões 14	1 hr	24/05/19	24/05/19	20	20	20	20	20	3
16	Reuniões 15	1 hr	31/05/19	31/05/19	20	20	20	20	20	3
17	Reuniões 16	1 hr	07/06/19	07/06/19	20	20	20	20	20	3
18	Atas	75,02 dias	22/02/19	07/06/19						
19	Atas 1	10 mins	22/02/19	22/02/19	0	0	0	100	0	1
20	Atas 2	10 mins	01/03/19	01/03/19	100	0	0	0	0	1
21	Atas 3	10 mins	08/03/19	08/03/19	0	100	0	0	0	1

Daniel Martins	Jorge Anjos	Pedro Ferreira	Ricardo Balreira	Rui Duarte	Computador
----------------	-------------	----------------	------------------	------------	------------

Tabela 3 - Lista de atividades do gráfico de Gantt do Planeamento (parte 3)

Nº	Tarefa	Duração	Início	Fim	Percentagem (%)					
22	Atas 4	10 mins	15/03/19	15/03/19	0	0	100	0	0	1
23	Atas 5	10 mins	22/03/19	22/03/19	0	0	0	0	100	1
24	Atas 6	10 mins	29/03/19	29/03/19	0	0	0	100	0	1
25	Atas 7	10 mins	05/04/19	05/04/19	0	0	0	0	100	1
26	Atas 8	10 mins	12/04/19	12/04/19	100	0	0	0	0	1
27	Atas 9	10 mins	19/04/19	19/04/19	0	100	0	0	0	1
28	Atas 10	10 mins	26/04/19	26/04/19	0	0	100	0	0	1
29	Atas 11	10 mins	03/05/19	03/05/19	0	0	0	0	100	1
30	Atas 12	10 mins	10/05/19	10/05/19	0	0	0	100	0	1
31	Atas 13	10 mins	17/05/19	17/05/19	100	0	0	0	0	1
32	Atas 14	10 mins	24/05/19	24/05/19	0	100	0	0	0	1
33	Atas 15	10 mins	31/05/19	31/05/19	0	0	100	0	0	1
34	Atas 16	10 mins	07/06/19	07/06/19	0	0	0	0	100	1

Daniel Martins

Jorge Anjos

Pedro Ferreira

Ricardo Balreira

Rui Duarte

Computador

Tabela 4 - Lista de atividades do gráfico de Gantt do Planeamento (parte 4)

Nº	Tarefa	Duração	Início	Fim	Percentagem (%)					
35	Gestão de Projeto	5,15 dias	20/02/19	28/02/19						
36	Criar grupo no codeUA	10 mins	20/02/19	20/02/19	0	0	0	100	0	1
37	Relatório Modelo	1 hr	21/02/19	21/02/19	0	100	0	0	0	1
38	Divisão de tarefas	5 dias	21/02/19	28/02/19	20	20	20	20	20	2
39	Relatório	71 dias	02/03/19	10/06/19						
40	Relatório Final	72 dias	02/03/19	10/06/19	20	20	20	20	20	5
41	Planeamento	20 dias	06/02/19	05/03/19						
42	Requisitos funcionais e não funcionais	3 dias	01/03/19	05/03/19	20	20	20	20	20	3
43	Definir utilizadores do sistema	2 dias	06/02/19	07/02/19	20	20	20	20	20	3
44	Modelação	8 dias	14/03/19	25/03/19						
45	Diagrama de casos de uso	1 dia	08/03/19	10/03/19	0	0	100	0	0	1
46	Diagrama de classes	2 dias	14/03/19	15/03/19	0	0	30	30	40	1
					Daniel Martins	Jorge Anjos	Pedro Ferreira	Ricardo Balreira	Rui Duarte	Computador

Tabela 5 - Lista de atividades do gráfico de Gantt do Planeamento (parte 5)

Nº	Tarefa	Duração	Início	Fim	Percentagem (%)					
47	Modelo de dados persistentes	2 dias	14/03/19	15/03/19	0	100	0	0	0	1
48	Prototipagem	10 dias	18/03/19	29/03/19						
49	Protótipo baixa fidelidade	10 dias	18/03/19	29/03/19	50	50	0	0	0	2.5
50	Base de dados	2 dias	01/04/19	02/04/19						
51	Criar tabelas	10 mins	05/04/19	05/04/19	0	0	0	0	100	1
52	Criar views	1 dia	02/04/19	02/04/19	0	0	0	100	0	1
53	Criar triggers	10 mins	02/04/19	02/04/19	0	0	0	0	100	1
54	Camada de apresentação	32 dias	05/04/19	20/05/19						
55	Estrutura da página	5 dias	26/04/19	26/04/19	20	20	20	20	20	3
56	Mostrar pontos no mapa	3 dias	03/05/19	03/05/19	0	0	0	0	100	1
57	<i>Popup</i> com informação	2 dias	10/05/19	10/05/19	0	0	100	0	0	1
58	Detalhes dos pontos	1 dia	17/05/19	17/05/19	0	100	0	0	0	1
59	Filtrar dados	4 dias	24/05/19	24/05/19	0	0	0	100	0	1
					Daniel Martins	Jorge Anjos	Pedro Ferreira	Ricardo Balreira	Rui Duarte	Computador

Tabela 6 - Lista de atividades do gráfico de Gantt do Planeamento (parte 6)

Nº	Tarefa	Duração	Início	Fim	Percentagem (%)					
60	Calcular rotas	8 dias	29/04/19	08/05/19	0	0	50	0	50	2
61	Obter geolocalização	1 dia	09/05/19	09/05/19	100	0	0	0	0	1
62	Mudar estilo do mapa	1 dia	10/05/19	10/05/19	0	0	100	0	0	1
63	Mostrar pontos de interesse	2 dias	13/05/19	14/05/19	0	100	0	0	0	1
64	Motor de busca	4 dias	15/05/19	20/05/19	0	100	0	0	0	1
65	Defesa do trabalho	8 dias	17/06/19	26/06/19						
66	Preparar apresentação	5 dias	17/06/19	21/06/19	20	20	20	20	20	3
67	Dia da apresentação	1 dia	26/06/19	26/06/19	20	20	20	20	20	1
					Daniel Martins	Jorge Anjos	Pedro Ferreira	Ricardo Balreira	Rui Duarte	Computador

2.2. Executado

Apresenta-se, em seguida, a execução real do trabalho, isto é, como realmente as tarefas se sucederam ao longo do trabalho. O procedimento é idêntico ao do planeamento: nas figuras abaixo encontram-se a descrição das tarefas (figuras 5 e 6), gráfico de Gantt (figuras 7 e 8) e a lista de atividades por meio de várias tabelas (tabelas 7-11) para o caso da figura do gráfico de Gantt não estiver visível.

2.2.1. O que diferiu

Estabelecendo a diferença entre o planeamento e a execução real do trabalho existem alguns aspetos que não se mantiveram iguais entre as duas fases devido à falta de empenho nalguns elementos do grupo e de coordenação resultando assim num trabalho que, apesar de ter ficado razoavelmente aceitável, tinha potencial para ser melhor, como se poderá ver ao longo do relatório. A distribuição das tarefas na execução não foi idêntica ao do planeamento (houve participação de elementos em tarefas que não estava estipulado participarem) e a funcionalidade da localização atual não foi a avante por ser uma baixa prioridade.

2.2.2. Execução das tarefas

Neste sentido, as figuras 5 e 6 aplicam-se à execução das tarefas onde estão identificadas as tarefas, duração, data de início e conclusão.

ID	Nome da Tarefa	Duração	Início	Conclusão
1	Reuniões	76,13 dias	Sex 22/02/19	Sex 07/06/19
2	Reuniões 1	1 hr	Sex 22/02/19	Sex 22/02/19
3	Reuniões 2	1 hr	Qui 28/02/19	Qui 28/02/19
4	Reuniões 3	1 hr	Sáb 09/03/19	Sáb 09/03/19
5	Reuniões 4	1 hr	Sex 15/03/19	Sex 15/03/19
6	Reuniões 5	1 hr	Seg 01/04/19	Seg 01/04/19
7	Reuniões 6	1 hr	Sex 05/04/19	Sex 05/04/19
8	Reuniões 7	1 hr	Sex 12/04/19	Sex 12/04/19
9	Reuniões 8	1 hr	Sex 19/04/19	Sex 19/04/19
10	Reuniões 9	1 hr	Sex 26/04/19	Sex 26/04/19
11	Reuniões 10	1 hr	Sex 03/05/19	Sex 03/05/19
12	Reuniões 11	1 hr	Sex 10/05/19	Sex 10/05/19
13	Reuniões 12	1 hr	Sex 17/05/19	Sex 17/05/19
14	Reuniões 13	1 hr	Sex 24/05/19	Sex 24/05/19
15	Reuniões 14	1 hr	Sex 31/05/19	Sex 31/05/19
16	Reuniões 15	1 hr	Sex 07/06/19	Sex 07/06/19
17	Atas	56,02 dias	Sex 22/02/19	Sex 10/05/19
18	Atas 1	10 mins	Sex 22/02/19	Sex 22/02/19
19	Atas 2	10 mins	Qui 28/02/19	Qui 28/02/19
20	Atas 3	10 mins	Sáb 09/03/19	Sáb 09/03/19
21	Atas 4	10 mins	Sex 15/03/19	Sex 15/03/19
22	Atas 5	10 mins	Seg 01/04/19	Seg 01/04/19
23	Atas 11	10 mins	Sex 10/05/19	Sex 10/05/19
24	Gestão de Projeto	55,02 dias	Qua 20/02/19	Ter 07/05/19
25	Criar grupo no codeUA	10 mins	Qua 20/02/19	Qua 20/02/19
26	Relatório Modelo	1 hr	Qui 21/02/19	Qui 21/02/19
27	Divisão de tarefas	54,88 dias	Qui 21/02/19	Ter 07/05/19

Figura 5 - Execução das tarefas (parte 1)

ID	Nome da Tarefa	Duração	Inicio	Conclusão
28	Relatório	75 dias	Sáb 02/03/19	Qui 13/06/19
29	Relatório Final	76 dias	Sáb 02/03/19	Qui 13/06/19
30	Planeamento	3 dias	Sex 01/03/19	Ter 05/03/19
31	Requisitos funcionais e não funcionais	2,6 dias	Sex 01/03/19	Ter 05/03/19
32	Definir utilizadores do sistema	0,8 dias	Sex 01/03/19	Sex 01/03/19
33	Modelação	23 dias	Qui 14/03/19	Seg 15/04/19
34	Diagrama de casos de utilização	1 dia	Sáb 09/03/19	Dom 10/03/19
35	Diagrama de classes	23 dias	Qui 14/03/19	Seg 15/04/19
36	Modelo de dados persistentes	115 dias	Qua 07/11/18	Seg 15/04/19
37	Prototipagem	16,67 dias	Sex 08/03/19	Sex 29/03/19
38	Protótipo baixa fidelidade	16,67 dias	Sex 08/03/19	Sex 29/03/19
39	Base de dados	36 dias	Ter 16/04/19	Ter 04/06/19
40	Criar tabelas	480 mins	Ter 16/04/19	Ter 16/04/19
41	Criar views	15 dias	Qua 15/05/19	Ter 04/06/19
42	Criar triggers	10 mins	Qui 18/04/19	Qui 18/04/19
43	Camada de apresentação	50 dias	Sex 05/04/19	Qui 13/06/19
44	Estrutura da página	2 dias	Sex 05/04/19	Seg 08/04/19
45	Mostrar pontos no mapa	1,43 dias	Qui 18/04/19	Sex 19/04/19
46	Popup com informação	2,5 dias	Ter 23/04/19	Qui 25/04/19
47	Detalhes dos pontos	29 dias	Sex 03/05/19	Qua 12/06/19
48	Filtrar dados	25 dias	Sex 03/05/19	Qui 06/06/19
49	Calcular rotas	29 dias	Sex 03/05/19	Qua 12/06/19
50	Mudar estilo do mapa	18 dias	Sex 03/05/19	Ter 28/05/19
51	Mostrar pontos de interesse	11 dias	Qui 30/05/19	Qui 13/06/19
52	Motor de busca	17 dias	Seg 06/05/19	Ter 28/05/19
53	Defesa do trabalho	8 dias	Seg 17/06/19	Qua 26/06/19
54	Preparar apresentação	5 dias	Seg 17/06/19	Sex 21/06/19
55	Dia da apresentação	1 dia	Qua 26/06/19	Qua 26/06/19

Figura 6 - Execução das tarefas (parte 2)

2.2.3. Gráfico de Gantt

As figuras 7 e 8, também representadas pelas primeira e segunda partes, respetivamente, dizem respeito ao gráfico de Gantt ligado às tarefas das figuras 5 e 6. As figuras encontram-se na página seguinte.

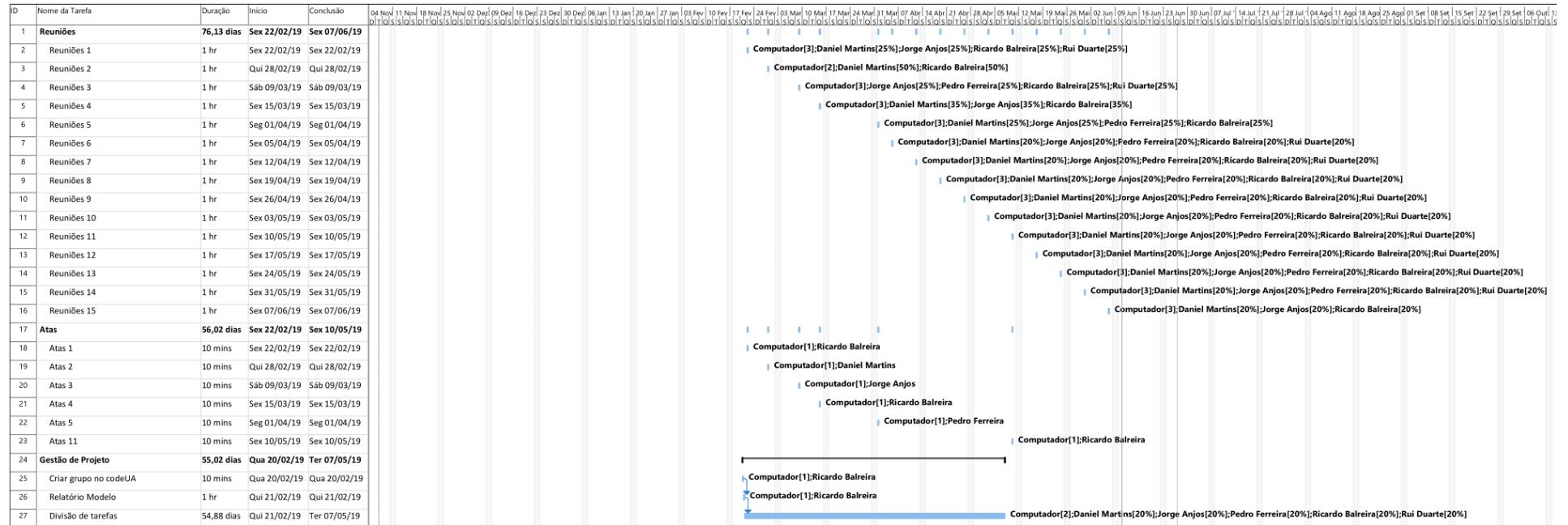


Figura 7 - Gráfico de Gantt da Execução (parte 1)

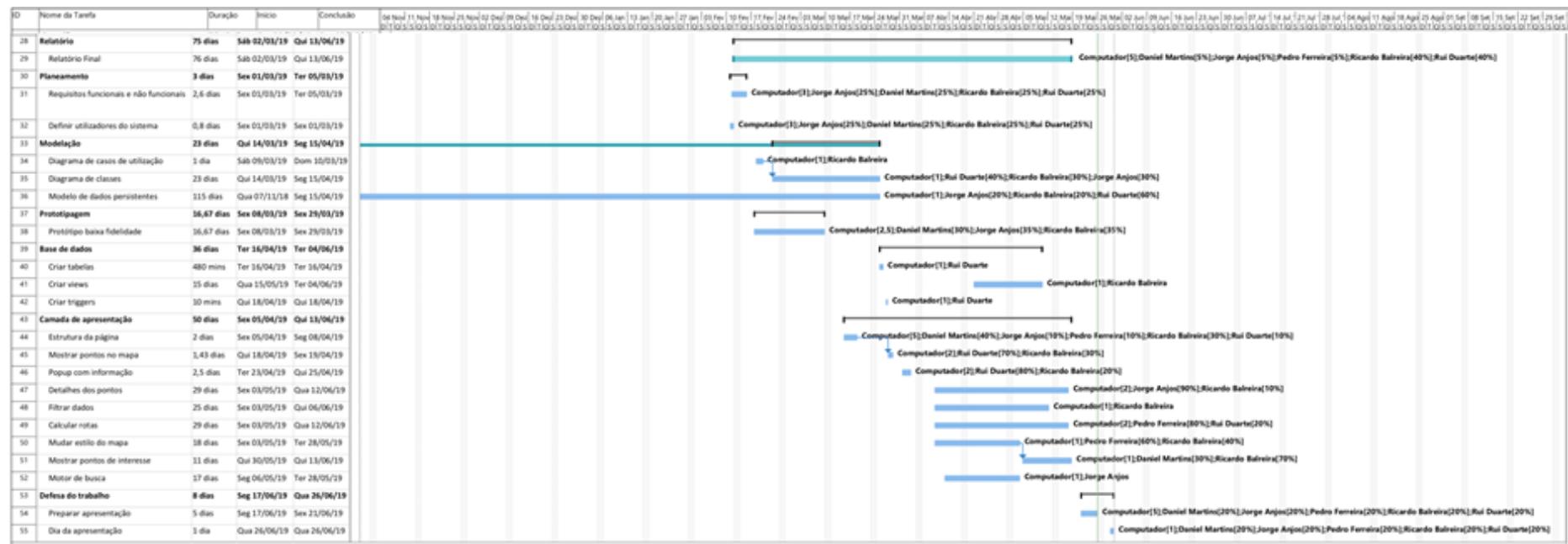


Figura 8 - Gráfico de Gantt da Execução (parte 2)

Tal como fora referido anteriormente, as tabelas 7-11 contêm a execução real das tarefas, considerando a designação de cada tarefa, duração, data de início e conclusão e ainda com a respetiva percentagem de participação de cada elemento do grupo.

Tabela 7 - Lista de atividades do gráfico de Gantt da Execução (parte 1)

Nº	Tarefa	Duração	Início	Fim	Percentagem (%)						
1	Reuniões	76,13 dias	22/02/19	07/06/19							
2	Reuniões 1	1 hr	22/02/19	22/02/19	25	25	0	25	25	25	3
3	Reuniões 2	1 hr	28/03/19	28/03/19	50	0	0	50	0	0	2
4	Reuniões 3	1 hr	09/03/19	09/03/19	0	25	25	25	25	25	3
5	Reuniões 4	1 hr	15/03/19	15/03/19	35	35	0	35	0	0	3
6	Reuniões 5	1 hr	01/04/19	01/04/19	25	25	25	25	0	0	3
7	Reuniões 6	1 hr	05/04/19	05/04/19	20	20	20	20	20	20	3
8	Reuniões 7	1 hr	12/04/19	12/04/19	20	20	20	20	20	20	3
					Daniel Martins	Jorge Anjos	Pedro Ferreira	Ricardo Balreira	Rui Duarte	Computador	

Tabela 8 - Lista de atividades do gráfico de Gantt da Execução (parte 2)

Nº	Tarefa	Duração	Início	Fim	Percentagem (%)					
9	Reuniões 8	1 hr	19/04/19	19/04/19	20	20	20	20	20	3
10	Reuniões 9	1 hr	26/04/19	26/04/19	20	20	20	20	20	3
11	Reuniões 10	1 hr	03/05/19	03/05/19	20	20	20	20	20	3
12	Reuniões 11	1 hr	10/05/19	10/05/19	20	20	20	20	20	3
13	Reuniões 12	1 hr	17/05/19	17/05/19	20	20	20	20	20	3
14	Reuniões 13	1 hr	24/05/19	24/05/19	20	20	20	20	20	3
15	Reuniões 14	1 hr	31/05/19	31/05/19	20	20	20	20	20	3
16	Reuniões 15	1 hr	07/06/19	07/06/19	20	20	0	20	0	3
17	Atas	56,02 dias	22/02/19	10/05/19						
18	Atas 1	10 mins	22/02/19	22/02/19	0	0	0	100	0	1
19	Atas 2	10 mins	28/02/19	28/02/19	100	0	0	0	0	1
20	Atas 3	10 mins	09/03/19	09/03/19	0	100	0	0	0	1

Daniel Martins	Jorge Anjos	Pedro Ferreira	Ricardo Balreira	Rui Duarte	Computador
-----------------------	--------------------	-----------------------	-------------------------	-------------------	-------------------

Tabela 9 - Lista de atividades do gráfico de Gantt da Execução (parte 3)

Nº	Tarefa	Duração	Início	Fim	Percentagem (%)					
21	Atas 4	10 mins	15/03/19	15/03/19	0	0	0	100	0	1
22	Atas 5	10 mins	01/04/19	01/04/19	0	0	100	0	0	1
23	Atas 11	10 mins	10/05/19	10/05/19	0	0	0	100	0	1
24	Gestão de Projeto	55,02 dias	20/02/19	07/05/19						
25	Criar grupo no codeUA	10 mins	20/02/19	20/02/19	0	0	0	100	0	1
26	Relatório Modelo	1 hr	21/02/19	21/02/19	0	0	0	100	0	1
27	Divisão de tarefas	54,88 dias	21/02/19	07/05/19	20	20	20	20	20	2
28	Relatório	75 dias	02/03/19	13/06/19						
29	Relatório Final	76 dias	02/03/19	13/06/19	10	15	10	32	32	5
30	Planeamento	3 dias	01/03/19	05/03/19						
31	Requisitos funcionais e não funcionais	2,6 dias	01/03/19	05/03/19	25	25	0	25	25	3
32	Definir utilizadores do sistema	0,8 dias	01/03/19	01/03/19	25	25	0	25	25	3
					Daniel Martins	Jorge Anjos	Pedro Ferreira	Ricardo Balreira	Rui Duarte	Computador

Tabela 10 - Lista de atividades do gráfico de Gantt da Execução (parte 4)

Nº	Tarefa	Duração	Início	Fim	Percentagem (%)					
33	Modelação	23 dias	14/03/19	15/04/19						
34	Diagrama de casos de uso	1 dias	09/03/19	10/03/19	0	0	0	100	0	1
35	Diagrama de classes	23 dias	14/03/19	15/04/19	0	30	0	30	40	1
36	Modelo de dados persistentes	23 dias	14/03/19	15/04/19	0	20	0	20	60	1
37	Prototipagem	16,67 dias	08/03/19	29/03/19						
38	Protótipo baixa fidelidade	16,67 dias	08/03/19	29/03/19	35	35	0	35	0	2.5
39	Base de dados	36 dias	16/04/19	04/06/19						
40	Criar tabelas	480 mins	16/04/19	16/04/19	0	0	0	0	100	1
41	Criar views	15 dias	15/05/19	04/06/19	0	0	0	100	0	1
42	Criar triggers	10 mins	18/04/19	18/04/19	0	0	0	0	100	1
43	Camada de apresentação	73,5 dias	05/03/19	13/06/19						
44	Estrutura da página	2 dias	05/04/19	08/04/19	40	10	10	30	10	5
45	Mostrar pontos no mapa	1,43 dias	18/04/19	19/04/19	0	0	0	30	70	2
					Daniel Martins	Jorge Anjos	Pedro Ferreira	Ricardo Balreira	Rui Duarte	Computador

Tabela 11 - Lista de atividades do gráfico de Gantt da Execução (parte 5)

Nº	Tarefa	Duração	Início	Fim	Percentagem (%)					
46	Popup com informação	2,5 dias	23/04/19	25/04/19	0	0	0	20	80	2
47	Detalhes dos pontos	33,33 dias	26/04/19	12/06/19	0	90	0	10	0	2
48	Filtrar dados	25 dias	03/05/19	06/06/19	0	0	0	100	0	1
49	Calcular rotas	25 dias	03/05/19	12/06/19	0	0	95	0	5	2
50	Mudar estilo do mapa	18 dias	03/05/19	28/05/19	0	0	60	40	0	2
51	Mostrar pontos de interesse	11 dias	30/05/19	13/06/19	30	0	0	70	0	2
52	Motor de busca	17 dias	06/05/19	28/05/19	0	100	0	0	0	1
53	Defesa do trabalho	8 dias	17/06/19	26/06/19						
54	Preparar apresentação	5 dias	17/06/19	21/06/19	20	20	20	20	20	5
55	Dia da apresentação	1 dia	26/06/19	26/06/19	20	20	20	20	20	1

Daniel Martins	Jorge Anjos	Pedro Ferreira	Ricardo Balreira	Rui Duarte	Computador
-----------------------	--------------------	-----------------------	-------------------------	-------------------	-------------------

3. Requisitos do sistema

Neste ponto são descritas as funcionalidades do sistema, isto é, especifica-se o que o sistema deve fazer, obrigatoriamente, e como o sistema se deve comportar, tendo em conta, por exemplo, a usabilidade e desempenho.

As tabelas 12 e 13 referem-se aos requisitos funcionais e não funcionais do sistema, respetivamente.

3.1. Requisitos funcionais

A tabela seguinte é alusiva aos requisitos funcionais do sistema.

Referência - numeração do requisito

Descrição - texto descritivo do requisito

Tipo - grau de prioridade do requisito (alto, médio, baixo)

Tabela 12 - Requisitos funcionais (RF)

Tipo	Ref^a	Descrição
Alto	RF 1	Visualizar o mapa geográfico com a representação da componente desportiva (clubes, pavilhões, ginásios e campos...) da região de Aveiro, complementado com a UA.
	RF 2	Filtrar os dados no mapa consoante a pesquisa do utilizador (por exemplo: quantos clubes existem com escalão sénior ou num determinado município).
	RF 3	Ter acesso à distância e ao tempo que é necessário para chegar de um ponto ao outro.
	RF 4	Clicar num ponto de referência no mapa e ter acesso à imagem, descrição e contactos (no caso de um clube desportivo, incluem-se os horários, sexo e escalões).
Médio	RF 5	Representar os pontos de interesse a uma distância igual ou a menos de 600 metros em volta de cada zona que tenha a componente desportiva (por exemplo: imagine-se um pai que leva a filha à aula de ginástica e pretende saber o centro comercial ou snack-bar mais próximo do local que a filha pratica ginástica).

Tipo	Ref^a	Descrição
Baixo	RF 6	Ter uma tabela de <i>logs</i> para guardar as alterações e inserções feitas em outras tabelas.

3.2. Requisitos não funcionais

A tabela abaixo corresponde aos requisitos não funcionais do sistema.

Referência - numeração do requisito

Descrição - texto descritivo do requisito

Tipo - grau de prioridade do requisito (alto, médio, baixo)

Tabela 13 - Requisitos não funcionais (RNF)

Tipo	Ref^a	Descrição
Médio	RNF 1	Máximo de dois segundos no tempo de resposta numa pesquisa de uma determinada informação entre a base de dados e a aplicação.
	RNF 2	A interface deve ter uma boa usabilidade (fácil acesso às funcionalidades, pouco tempo de aprendizagem da aplicação...).
Baixo	RNF 3	Adaptar a página Web de acordo com o dispositivo a ser utilizado.
	RNF 4	A plataforma deverá ter a possibilidade de traduzir o conteúdo das páginas para a linguagem universal (inglês).

4. Diagrama de casos de utilização

Dependendo da utilidade e dos requisitos que fundamentam um determinado sistema, o diagrama de casos de utilização tem como objetivo capturar as funcionalidades da aplicação tal como é visto pelos utilizadores.

Em relação à aplicação em questão, há um conjunto de atores que interagirão com esta, que neste caso, é apenas o utilizador. Este terá ao dispor funcionalidades que a plataforma fornece, tais como: ver a componente desportiva no mapa, filtrar os pontos representados no mapa, obter detalhes de um determinado ponto, pesquisa avançada e obter direções entre dois pontos (itinerário). Cada um destes anteriormente referido domina-se, tecnicamente, por caso de utilização.

Os subtópicos seguintes descrevem os atores e os casos de utilização que integram o sistema. Dentro dos casos de utilização fez-se uma associação aos *mockups* do protótipo de baixa fidelidade que lhe correspondem, isto é, se o caso de utilização estiver relacionado com o ver a componente desportiva no mapa, por exemplo, apresenta-se o *mockup* do ver os pontos no mapa. Antes de serem apresentados os *mockups*, detalha-se o caso de utilização referindo o seguinte: atores e requisitos funcionais associados; pré-condição (o que é preciso acontecer antes do caso de uso), pós-condição (o que deve acontecer depois de concluído o caso de uso), fluxo básico (série de afirmações declarativas que listam os passos do caso de uso), fluxo alternativo (alternativas ao caminho básico) e ainda a sua finalidade (descrição breve).

4.1. Atores

A tabela 14 contém uma breve descrição dos atores que se definem como utilizadores do sistema, onde estão o nome do ator e respetiva descrição.

Tabela 14 - Atores e descrição do seu papel no sistema

Ator	Descrição
Utilizador	É quem interage com o sistema tendo possibilidade de visualizar a componente desportiva no mapa, filtrar, alterar a cartografia de contexto, pesquisar e ver detalhes de um determinado clube, universidade ou infraestrutura e, ainda o trajeto entre dois pontos.

4.2. Descrição dos casos de utilização

Segue-se então a descrição de cada caso de utilização, que integra o diagrama de casos de utilização, e a respetiva associação ao protótipo de baixa fidelidade.

4.2.1. Visualizar mapa

As figuras 9 e 10 dizem respeito ao visualizar o mapa. Sendo esta a página inicial, o utilizador tem acesso à componente desportiva da região de Aveiro e às interfaces para filtragem e visualização dos detalhes de cada respetivo ponto no mapa (figura 9). Ao selecionar um ponto no mapa surge um *popup* referindo a principal descrição (figura 9) e uma informação mais detalhada como se observa na figura 10. Ainda é possível observar as seguintes funcionalidades: alterar a cartografia de contexto, isto é, mudar o estilo do mapa de fundo; proceder a uma pesquisa avançada no motor de busca; e, ver infraestruturas ou clubes/Universidade consoante o separador selecionado. A figura 11 demonstra o que acontece quando se seleciona um ponto qualquer num mapa – surgem os pontos de interesse (restaurante, café, centro comercial...) mais próximos.

Estes *mockups* estão associados ao caso de utilização “Visualizar mapa”, tal como se pode comprovar pela tabela 15, que pormenoriza o caso de uso em questão.

Tabela 15 - Descrição do caso de uso "Visualizar mapa"

Nome:	Visualizar mapa
Atores:	Utilizador
Finalidade:	Visualizar o mapa permitindo ver a descrição da componente desportiva
Prioridade:	Alto
Pré-Condições:	O sistema retorna devidamente a página
Pós-Condições:	O utilizador recebeu a página O utilizador conseguiu ver a componente desportiva no mapa e ainda a respetiva descrição
Fluxo Básico:	<ol style="list-style-type: none"> 1. O sistema apresenta a página inicial 2. O sistema apresenta o mapa com a componente desportiva e seus detalhes

Nome:	Visualizar mapa
Fluxo Alternativo:	1a. e 2a. A página não é carregada
Requisitos Funcionais:	RF 1, RF 4, RF 5

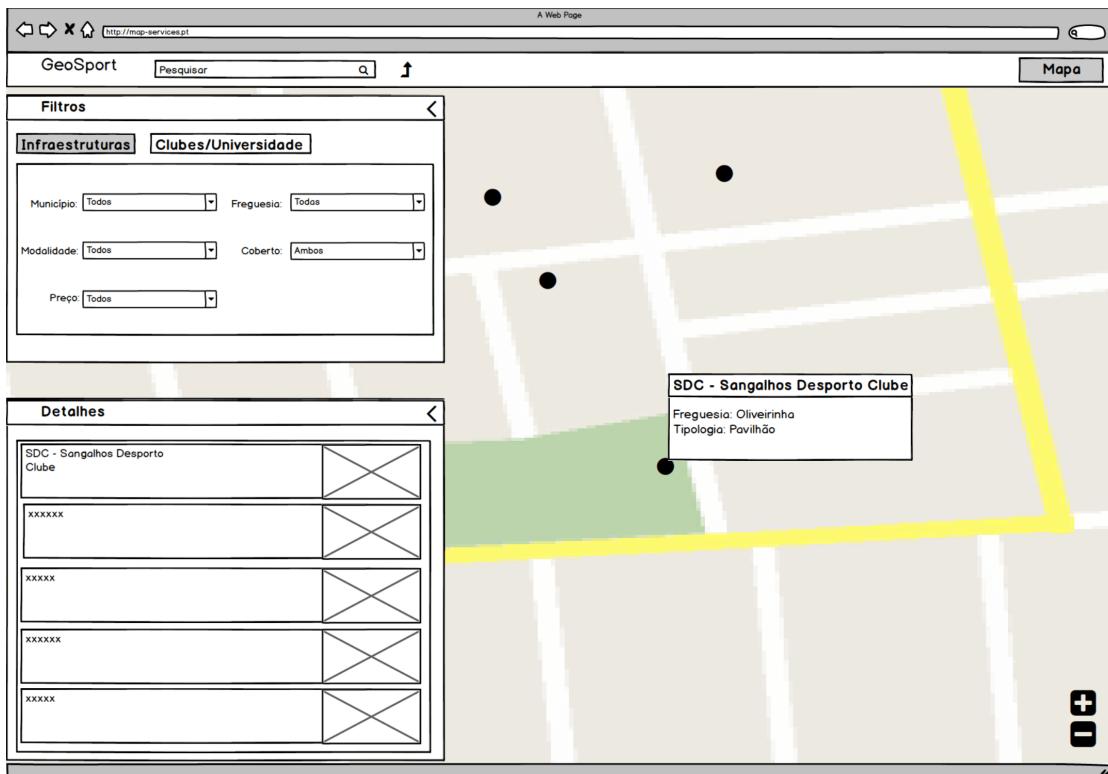


Figura 9 - *Mockup* de visualizar o mapa com descrição e filtragem da componente desportiva

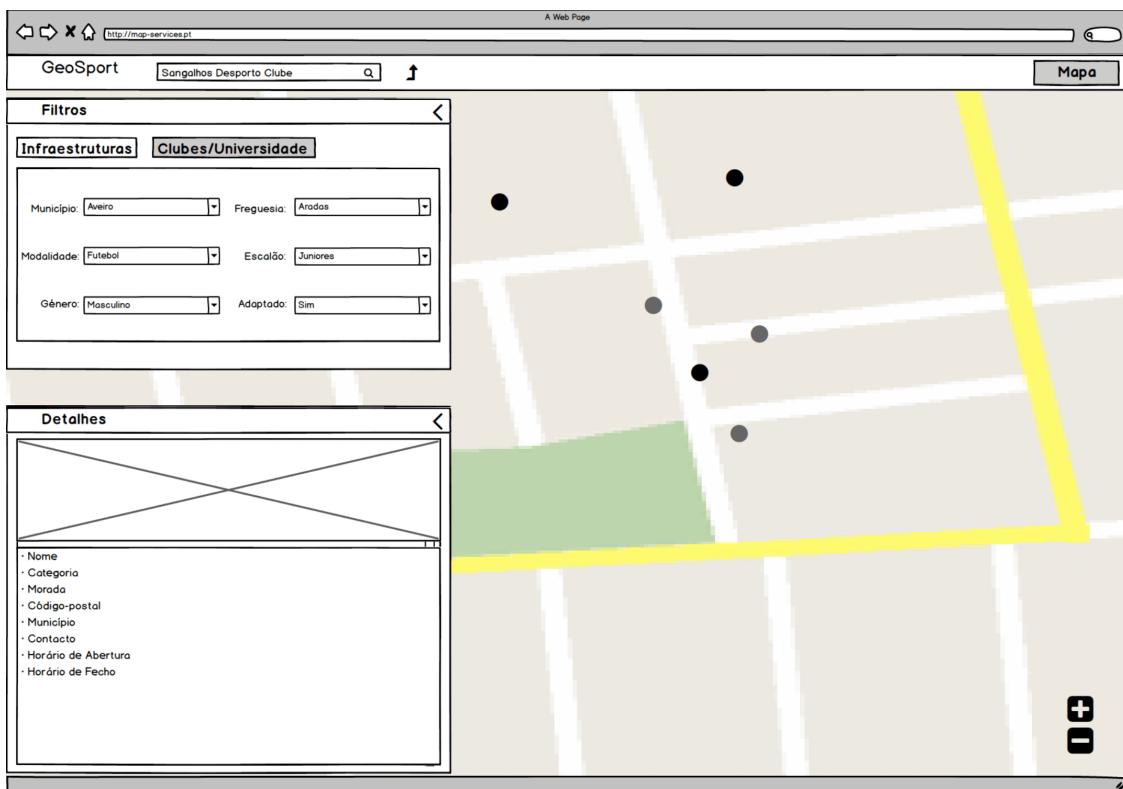


Figura 10 - Mockup com uma descrição detalhada de um ponto selecionado

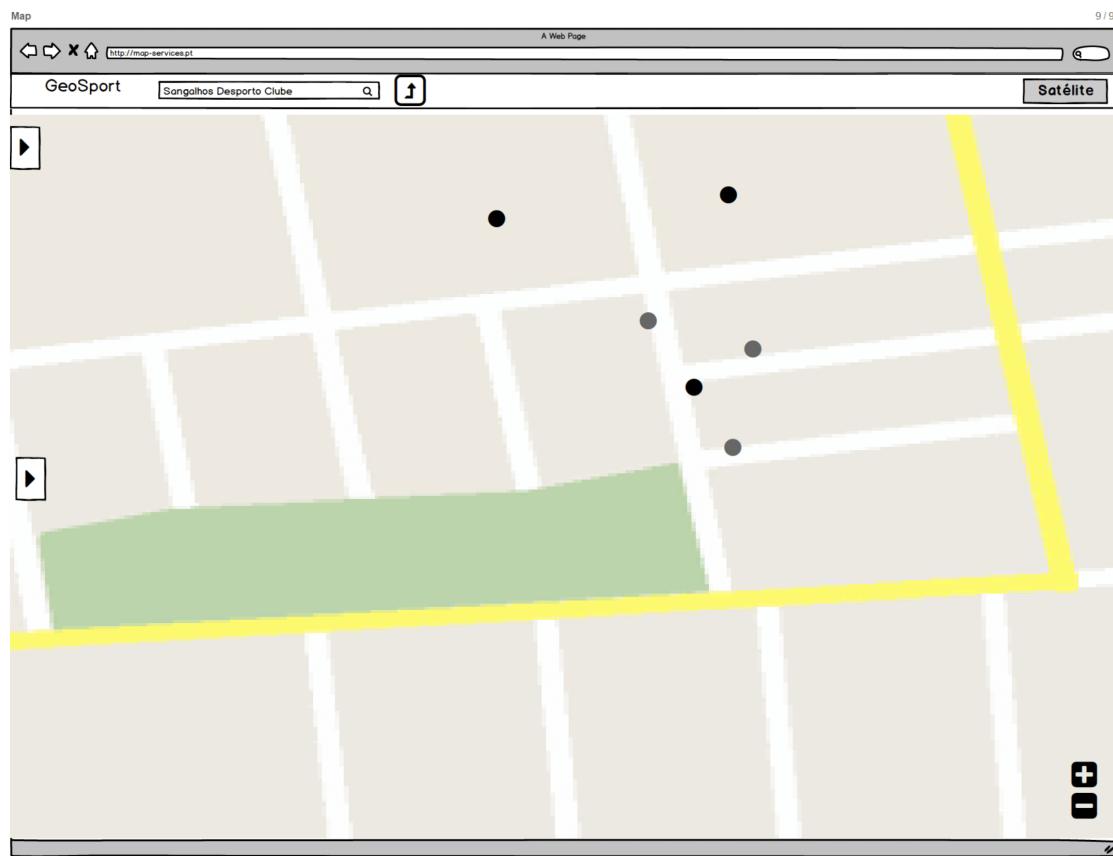


Figura 11 - Mockup com pontos de interesse ao clicar num ponto do mapa

4.2.2. Pesquisar informação

As figuras 12 e 14 consideram a pesquisa de informação utilizando o itinerário ou a filtragem da componente desportiva. Desta forma, o utilizador tem possibilidade de restringir os pontos visualizados no mapa, consoante os critérios definidos em cada separador dos filtros (Infraestruturas ou Clubes/Universidade), tal como se pode observar pela figura 12, e ainda obter o trajeto com tempo e duração entre um ponto de partida e um ponto de chegada (figura 14). Só é possível ter acesso ao itinerário selecionado o botão das direções situado do lado direito do motor de busca. A figura 13 serve apenas para ver de forma mais visível o comportamento da filtragem dos pontos.

Estes *mockups* estão associados ao caso de utilização “Pesquisar informação”, tal como se pode comprovar pela tabela 16, que pormenoriza o caso de uso em questão.

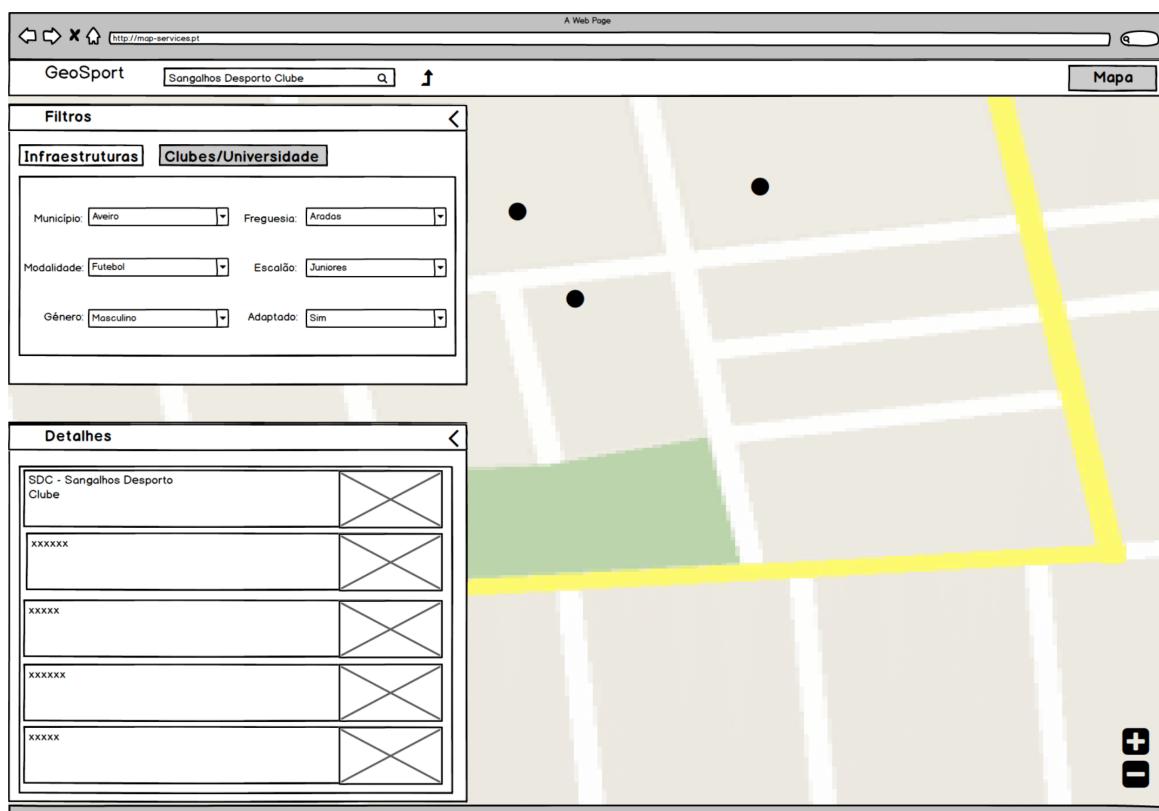


Figura 12 - Mockup com filtragem dos pontos do mapa

Filtros

Infraestruturas **Clubes/Universidade**

Município: Aveiro	Freguesia: Aradas
Modalidade: Futebol	Escalão: Juniores
Género: Masculino	Adaptado: Sim

Figura 13 - Filtragem dos pontos visto com maior detalhe

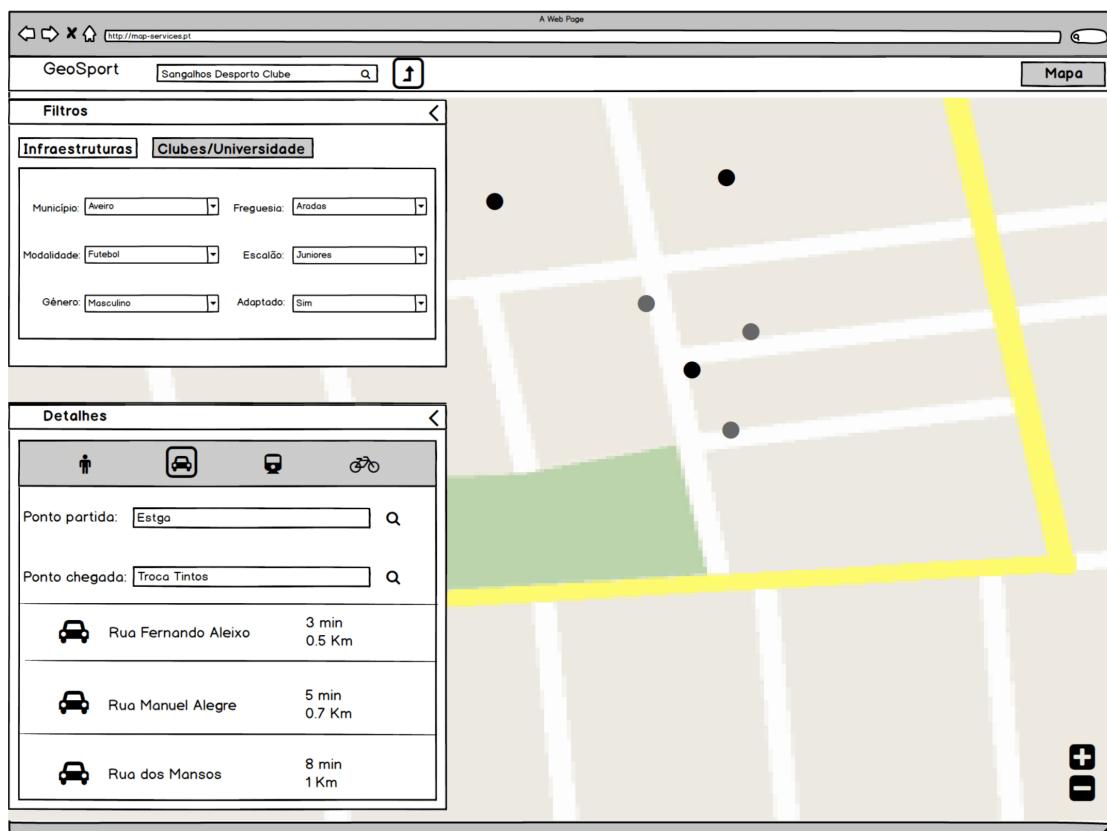


Figura 14 - Mockup com itinerário ao selecionar o botão das direções

Tabela 16 - Descrição do caso de uso " Pesquisar informação"

Nome:	Pesquisar informação
Atores:	Utilizador
Finalidade:	Filtrar os pontos no mapa e obter o trajeto entre dois pontos
Prioridade:	Alto
Pré-Condições:	O sistema retorna devidamente a página
Pós-Condições:	O utilizador recebeu a página O utilizador conseguiu filtrar e obter as direções entre dois pontos
Fluxo Básico:	<ol style="list-style-type: none"> 1. O sistema apresenta a página inicial 2. O sistema apresenta o mapa com a componente desportiva e seus detalhes 3. O utilizador é capaz de filtrar os pontos 4. O utilizador consegue obter as direções
Fluxo Alternativo:	<ol style="list-style-type: none"> 1a. e 2a. A página não é carregada 3a. O sistema não consegue retornar os pontos filtrados 4a. O sistema não consegue retornar as direções
Requisitos Funcionais:	RF 2, RF 3

4.3. Diagrama integral

Depois de descritos os casos de uso complementando com o protótipo de baixa fidelidade, apresenta-se agora o diagrama de casos de utilização que reúne todos os casos de utilização abordados nas secções anterior (figura 15). O diagrama encontra-se na página seguinte.

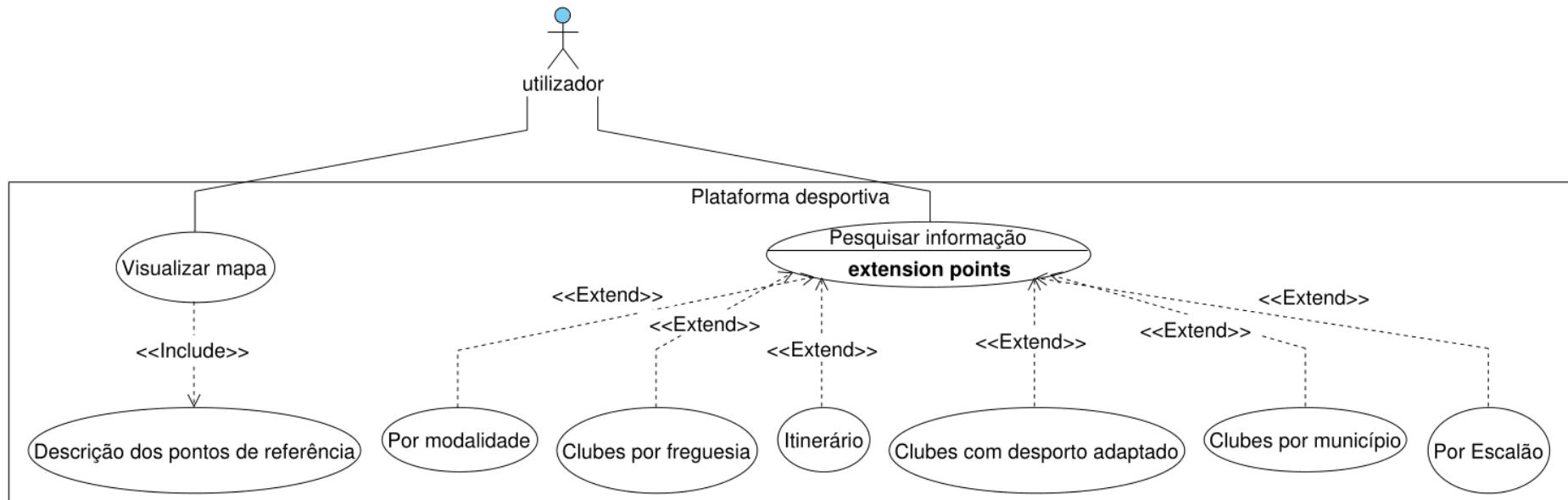


Figura 15 - Diagrama de casos de utilização

5. Diagrama de classes

A figura 16 ilustra o diagrama de classes, mostrando os atributos e a sua relação.

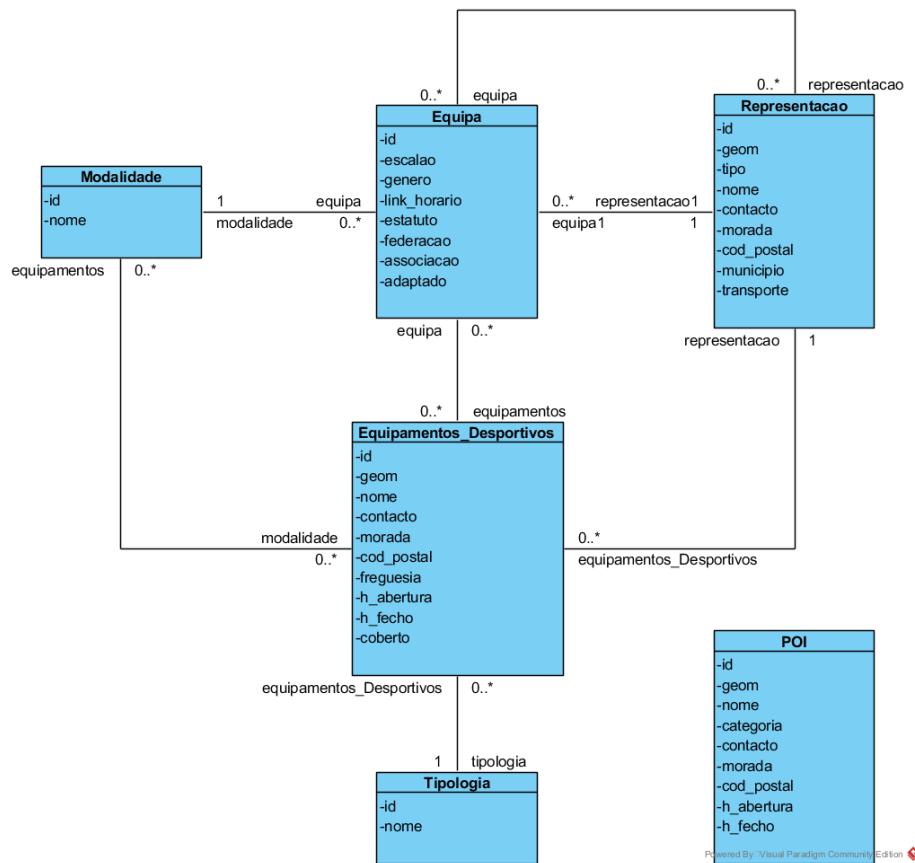


Figura 16 - Diagrama de Classes

6. Modelo de dados persistentes

6.1. Base de dados

Pretendeu-se construir um modelo capaz de identificar todas as características do projeto, as atividades mais importantes, e priorizar as diferentes etapas do trabalho a desenvolver.

Para a modelação da base de dados, tomou-se em atenção que esta deveria:

- Assegurar o cruzamento eficiente de informação, para evitar redundância de dados;
- Permitir obter ganhos importantes na análise e consulta da informação gerada;
- Reduzir o tempo na tomada de decisão, tornando a aplicação a desenvolver uma ferramenta relevante para a gestão.

Após a análise de requisitos e da criação do diagrama de casos de uso e de classes, procedeu-se à normalização dos conceitos. Seguiu-se a criação do diagrama lógico da base de dados, como demonstrado na figura 17.

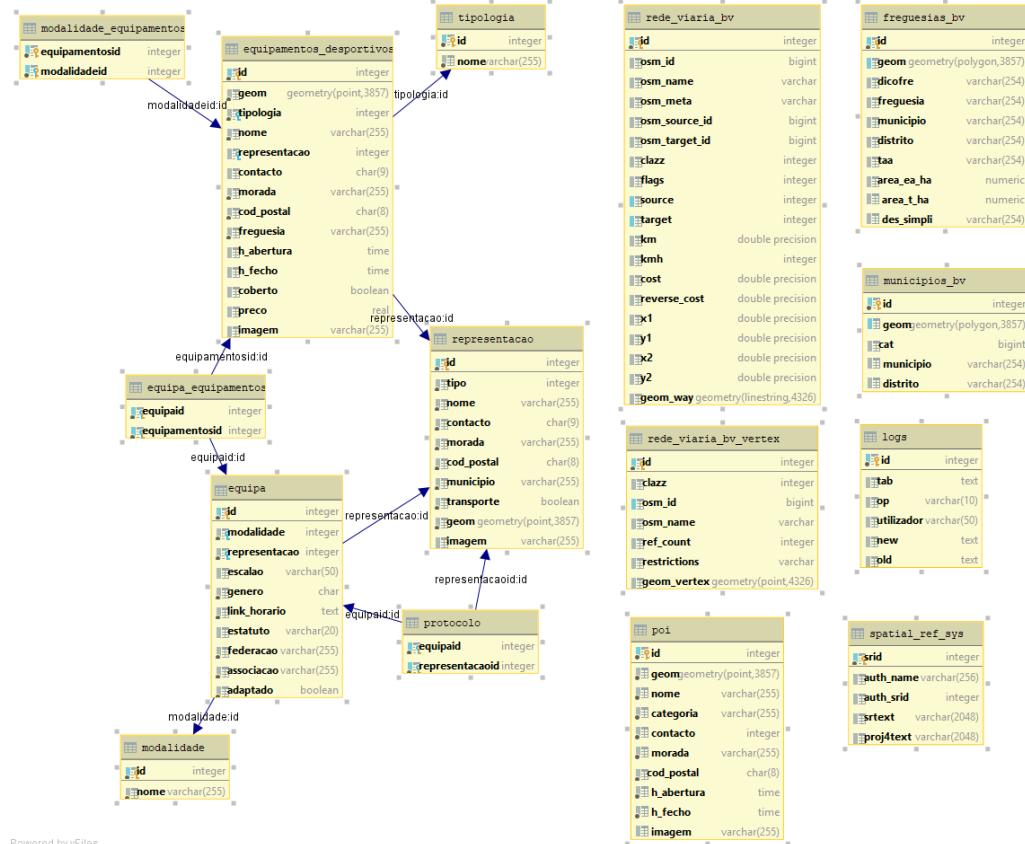


Figura 17 - Diagrama lógico da Base de Dados

6.2. SGBD alvo

O SGBD escolhido para desenvolver este projeto é o PostgreSQL. O PostgreSQL é um sistema de gestão de bases de dados objeto-relacional baseado no POSTGRES desenvolvido pelo Departamento de Ciência da Computação da Universidade da Califórnia em Berkeley.

Esta base de dados foi pensada tendo em vista o modelo funcional e de integridade dos seus dados. Recorreram-se a múltiplas tabelas por se considerar que algumas destas apenas fariam sentido se fossem únicas, evitando dados repetidos em várias tabelas. Com esta metodologia obtêm-se dados mais fidedignos no manuseamento das tabelas, como se pode verificar no modelo relacional. Adicionalmente, é possível ter um maior controlo sobre os dados inseridos pelos utilizadores do sistema aquando da inserção de dados errados.

6.3. Restrições (*Constraints*)

Constraints NOT NULL

Os constraints NOT NULL permitem tornar um campo de preenchimento obrigatório. Foram usados em todos os atributos com a exceção dos considerados opcionais.

Constraints UNIQUE

Os constraints UNIQUE permitem somente a introdução de valores únicos. Além das chaves primárias e estrangeiras que são sempre únicas, não houve a necessidade de aplicar este conceito.

6.4. Definição de vistas (Views)

Vistas (views) são representações virtuais de tabelas criadas a partir de comandos SELECT. Uma VIEW funciona como uma janela, dando diferentes perspetivas da BD para diferentes utilizadores. Geralmente utilizadas para devolver dados desnortinalizados. Em vez de obrigar o utilizador a proceder constantemente a um conjunto de JOINS, as VIEWS permitem gerar facilmente um “ResultSet” com base numa consulta previamente gravada.

De forma a agilizar as queries efetuadas à base de dados, evitando a criação de múltiplos joins, foram criadas vistas, como apresentado na figura 18.



Figura 18 - Vistas

6.5. Procedimentos de execução automática (*Triggers*)

Um trigger é um tipo especial de Stored Procedure que é invocado automaticamente sempre que uma query de ação (INSERT, UPDATE ou DELETE) é executada sobre uma tabela ou view à qual o trigger esteja associado. Permitem implementar as regras definidas no desenho da BD, implementar regras subjacentes à lógica da aplicação e garantir a integridade e consistência dos dados.

Em termos de auditoria, foi criado uma relação logs (registos) com o objetivo de registar todas as operações do tipo INSERT, UPDATE e DELETE efetuadas em todas as tabelas da BD, com a óbvia exceção da própria relação de logs. A função “f_logs()” foi a responsável por tal operação, gravando os dados alterados em formato JSON para a relação NEW ou OLD (dependendo do tipo de operação) como abaixo apresentado.

```
CREATE OR REPLACE FUNCTION f_logs() RETURNS trigger AS $$  
BEGIN  
    IF TG_OP = 'INSERT' THEN  
        INSERT INTO logs (tab, op, utilizador, new)  
        VALUES (TG_RELNAME, TG_OP, user, row_to_json(NEW));  
        RETURN NEW;  
    ELSIF TG_OP = 'UPDATE' THEN  
        INSERT INTO logs (tab, op, utilizador, new, old)  
        VALUES (TG_RELNAME, TG_OP, user, row_to_json(NEW),  
        row_to_json(OLD));  
        RETURN NEW;  
    ELSIF TG_OP = 'DELETE' THEN  
        INSERT INTO logs (tab, op, utilizador, old)  
        VALUES (TG_RELNAME, TG_OP, user, row_to_json(OLD));  
        RETURN OLD;  
    END IF;  
END;  
$$ LANGUAGE plpgsql SECURITY DEFINER;
```

De seguida é necessário criar um trigger para cada uma das relações a monitorizar.

Para automatizar tal operação é criada a função “select_all_triggers()” que imprime um linha de texto por relação com o código para a criação do trigger. A figura 19 mostra a função em funcionamento.

```

1 CREATE TRIGGER logs_tipologia AFTER INSERT OR UPDATE OR DELETE ON tipologia FOR EACH ROW EXECUTE PROCEDURE f_logs();
2 CREATE TRIGGER logs_spatial_ref_sys AFTER INSERT OR UPDATE OR DELETE ON spatial_ref_sys FOR EACH ROW EXECUTE PROCEDURE f_logs();
3 CREATE TRIGGER logs_representacao AFTER INSERT OR UPDATE OR DELETE ON representacao FOR EACH ROW EXECUTE PROCEDURE f_logs();
4 CREATE TRIGGER logs_rede_viaria_bv_vertex AFTER INSERT OR UPDATE OR DELETE ON rede_viaria_bv_vertex FOR EACH ROW EXECUTE PROCEDURE f_logs();
5 CREATE TRIGGER logs_rede_viaria_bv AFTER INSERT OR UPDATE OR DELETE ON rede_viaria_bv FOR EACH ROW EXECUTE PROCEDURE f_logs();
6 CREATE TRIGGER logs_raster_overviews AFTER INSERT OR UPDATE OR DELETE ON raster_overviews FOR EACH ROW EXECUTE PROCEDURE f_logs();
7 CREATE TRIGGER logs_raster_columna AFTER INSERT OR UPDATE OR DELETE ON raster_columna FOR EACH ROW EXECUTE PROCEDURE f_logs();
8 CREATE TRIGGER logs_protocolo AFTER INSERT OR UPDATE OR DELETE ON protocolo FOR EACH ROW EXECUTE PROCEDURE f_logs();
9 CREATE TRIGGER logs_poi AFTER INSERT OR UPDATE OR DELETE ON poi FOR EACH ROW EXECUTE PROCEDURE f_logs();
10 CREATE TRIGGER logs_municipios_bv AFTER INSERT OR UPDATE OR DELETE ON municipios_bv FOR EACH ROW EXECUTE PROCEDURE f_logs();
11 CREATE TRIGGER logs_modalidades AFTER INSERT OR UPDATE OR DELETE ON modalidades FOR EACH ROW EXECUTE PROCEDURE f_logs();
12 CREATE TRIGGER logs_modalidade_equipamentos AFTER INSERT OR UPDATE OR DELETE ON modalidade_equipamentos FOR EACH ROW EXECUTE PROCEDURE f_logs();
13 CREATE TRIGGER logs_modalidade AFTER INSERT OR UPDATE OR DELETE ON modalidade FOR EACH ROW EXECUTE PROCEDURE f_logs();
14 CREATE TRIGGER logs_geometry_columns AFTER INSERT OR UPDATE OR DELETE ON geometry_columns FOR EACH ROW EXECUTE PROCEDURE f_logs();
15 CREATE TRIGGER logs_geography_columns AFTER INSERT OR UPDATE OR DELETE ON geography_columns FOR EACH ROW EXECUTE PROCEDURE f_logs();
16 CREATE TRIGGER logs_equipamentos_desportivos AFTER INSERT OR UPDATE OR DELETE ON equipamentos_desportivos FOR EACH ROW EXECUTE PROCEDURE f_logs();
17 CREATE TRIGGER logs_equipa_equipamentos AFTER INSERT OR UPDATE OR DELETE ON equipa_equipamentos FOR EACH ROW EXECUTE PROCEDURE f_logs();
18 CREATE TRIGGER logs_equipa AFTER INSERT OR UPDATE OR DELETE ON equipa FOR EACH ROW EXECUTE PROCEDURE f_logs();
```

Figura 19 - Resultado da função select_all_triggers()

6.6. Extensões

O PostgreSQL dá a possibilidade de estender as suas funcionalidades através da instalação de módulos adicionais, chamados extensões.

As extensões utilizadas neste projeto foram as seguintes:

- PL/pgSQL – linguagem estrutural estendida do SQL que tem por objetivo auxiliar as tarefas de programação no PostgreSQL. Adiciona funcionalidades tais como loops estruturados (for, while) e controlo de decisão (if then else).
- PostGIS – extensão espacial que permite o armazenamento de objetos GIS;
- PgRouting – extensão que permite calcular rotas através de informação geoespacial guardada na base de dados.

6.7. Procedimentos armazenados (Stored Procedures)

As funções, também conhecidas por Stored Procedures, permitem executar operações que levariam normalmente a diversas *queries* caso fosse realizado numa simples função dentro da base de dados. Com isto, seguindo esta abordagem é possível declarar variáveis e efetuar imensas operações numa só função com estas características, sendo a lógica idêntica às funções que existem em outras linguagens de programação.

No que respeita às funcionalidades da aplicação era necessário mostrar, após selecionar uma componente desportiva no mapa, todos os pontos de interesse a uma distância igual ou a menos de 600 metros, tendo sido este o valor acordado pelos elementos do grupo. Para prosseguir à implementação desta *feature* teve-se em conta o seguinte: uma vez que o propósito é mostrar os pontos de interesse em volta de um determinado ponto de referência no mapa, necessita-se assim de utilizar um dos algoritmos de *routing* da extensão pgRouting e a tabela referente aos arcos (linhas) e outra referente aos nós (extremidades das linhas) da rede viária do Baixo Vouga (atualmente designada de Região de Aveiro). O procedimento consistia, então, em selecionar o nó mais próximo da componente desportiva selecionada pela tabela dos nós (operador geométrico “ $<->$ ”) e, a partir desse mesmo nó, calcular os limites com distância máxima de 600 metros utilizando a tabela dos arcos (algoritmo *pgr_drivingDistance*). Tendo agora os limites em volta do ponto selecionado calcula-se então os pontos de interesse que estão dentro desse limite (função *ST_Contains*). É importante referir que o sistema de referência de coordenadas (SRID) usado nos campos com tipo de dados *geometry* foi o 3857 pelas razões que vão ser descritas mais à frente. Apenas a rede viária está no sistema de coordenadas 4326. Não se converteu para 3857 por serem tabelas com bastantes registos pelo que se optou por deixar no SRID padrão.

6.7.1. Nó mais próximo

Frisando ainda algumas particularidades não referidas anteriormente, o operador geométrico “ $<->$ ” permite selecionar a distância entre dois campos que, neste caso vão ser os valores do campo *geom_vertex* e o resultado da construção em *geometry* das coordenadas (longitude e latitude) do ponto selecionado. Visto que a *query* retornará o valor do campo *geom_vertex* dos registos por ordem crescente, utilizou-se então a operação *LIMIT 1* para obter apenas o nó mais próximo. A ordenação fez-se graças ao *ORDER BY*, que por defeito, retorna por ordem crescente num certo atributo.

6.7.2. Algoritmo Driving Distance

Em seguida, na segunda operação SELECT estão incluídas as funções espaciais ST_ConcaveHull e ST_Union para além do algoritmo pgr_drivingDistance. A função ST_ConcaveHull representa uma possível geometria côncava que cerca todas as geometrias dentro de um conjunto. Esta função junta então o valor geométrico resultante da função ST_Union, que por sua vez, provém do campo geom_vertex da tabela dos nós formando, portanto, um polígono. A função ST_Union é uma função de agregação que recebe um conjunto de geometrias e une-os num único valor geométrico (figura 20). Havia duas possibilidades de funções para cercar o valor geométrico: ST_ConcaveHull ou ST_ConvexHull. Como é possível observar pelas figuras 21 e 22 existe uma ligeira diferença nas duas abordagens, que ilustram as funções ST_ConcaveHull e ST_ConvexHull, respetivamente. O teste foi realizado executando apenas a query que está a ser descrita com a análise do resultado pela interface do pgAdmin 4. Nota-se a ausência significativa no detalhe em circundar cada nó do Convex Hull em relação ao Concave Hull. Para algumas aplicações, o método Convex Hull não representa da melhor forma os limites de um dado conjunto de pontos, o que leva a inferir que o algoritmo Concave Hull é o que melhor se adapta para o trabalho.

Relativamente ao ST_Union, ainda havia outra alternativa com as mesmas funções – ST_Collect – porém esta última é apenas utilizada para pontos, linhas ou *geometry collections*. Para polígonos, deve-se usar a função ST_Union.

```
funcao_union
geometry
0104000020E6100000640000000101000000666F84A0FEF620C013DD0E6844584440010100000A6214F37F5F620C07DFD...
```

Figura 20 - Resultado da execução da função ST_Union



Figura 21 - Resultado utilizando a função
ST_ConcaveHull



Figura 22 - Resultado utilizando a função
ST_ConvexHull

O que o algoritmo pgr_drivingDistance vai fazer é extrair todos os nós da rede viária que tenham um custo (de atravessar um arco) igual ou menos em relação à distância (600 metros), iniciando no nó mais próximo ao ponto selecionado no mapa. Por que motivo se usou o algoritmo pgr_drivingDistance? Porque é o único algoritmo que retorna a distância percorrida a partir de um determinado nó. Calcula o caminho mais curto (com o algoritmo Dijkstra) e, depois, extrai o custo para chegar a cada nó a partir do nó inicial. Utilizando estes nós e custos, é possível calcular o polígono a uma distância menor ou igual a 600 metros em volta do ponto selecionado.

6.7.3. Pontos de interesse contidos no polígono

Para finalizar, tendo agora o polígono, após ter sido usado o algoritmo pgr_drivingDistance, é necessário executar outra *query*. Com a função ST_Contains é possível, com a seleção de todos os pontos de interesse da tabela “poi”, encontrar quais tuplos desta tabela se encontram dentro do polígono calculado na *query* anterior. Existe ainda outra função – ST_Within – que é o inverso da função ST_Contains. Isto resulta então em todos os pontos de interesse que estão situados a menos ou igual a 600 metros do ponto selecionado no mapa. Para ambas as funções, os parâmetros de entrada devem estar no mesmo sistema de referência de coordenadas.

Depois de perceber o fundamento do processo induz-se assim que a melhor forma seria tornar isto dinâmico, ou seja, qualquer que seja a componente desportiva selecionada no mapa, é sempre possível obter os pontos de interesse em volta, sendo que o método mais plausível seria criar uma Stored Procedure. Outra alternativa seria ter uma tabela temporária que armazena os nós extraídos pelo algoritmo do pgRouting; contudo, não seria muito prático e tornava-se desnecessário, tendo em conta a diversidade de opções que os SGBD fornecem, tais como as Stored Procedures.

Usaram-se ainda o ST_Transform, ST_GeometryFromText e ST_AsText para converter uma determinada geometria num diferente sistema de referência de coordenadas, construir uma geometria a partir de uma representação Well-Known text (WKT) e retornar uma geometria em representação WKT, respetivamente. Uma vez que a rede viária e as coordenadas do ponto selecionado estão no SRID 4326 e no SRID padrão (3857), respetivamente, teve-se de transformar a geometria das coordenadas para o mesmo SRID que a rede viária tem. No retorno da geometria dos pontos de interesse enviou-se, assim, no formato GeoJSON pela função ST_AsGeoJSON na qual é uma das notações utilizadas para a troca de dados entre o servidor e o cliente (outro exemplo: GML).

7. Tecnologias usadas

Para o desenvolvimento do código deste projeto foram utilizadas as seguintes bibliotecas e extensões que considerámos essenciais:

- **jQuery** – Biblioteca JavaScript que faz com que o código seja mais curto e simples de escrever/interpretar;
- **OpenLayers** – Biblioteca JavaScript utilizada para trabalhar com mapas na maioria dos browsers sem dependências do lado do servidor. Permite carregar dados geo-espaciais do lado do cliente de forma simples, como modelos de dados raster e vetorial;
- **ol-ext** – Extensões OpenLayers que adicionam funcionalidades ao OpenLayers;
- **Bootstrap** – Framework que permite o desenvolvimento de interfaces responsivas para a web;
- **FontAwesome** – Biblioteca de ícones e logos vetoriais;
- **SweetAlert2** – Permite a criações de caixas de diálogo personalizadas.

O cálculo das rotas foi feito recorrendo à API das direções do MapBox.

8. Camada de apresentação

Antes de se dar seguimento à descrição das etapas desenvolvidas, há que considerar o seguinte:

- Utilizou-se o sistema de referência de coordenadas 3857 em todas as tabelas espaciais, à exceção da rede viária, por ser o sistema de projeção utilizado para apresentar mapas da Google Maps, OpenStreetMap, etc. e por ser o Web Mercator projection padrão do OpenLayers. Deste modo, permite a simplificação da gestão da informação geográfica, não havendo necessidade de realizar transformações constantes ao projetar os dados espaciais no mapa. Apenas é necessária transformação (3857 para 4326) quando é para executar a função de obtenção dos pontos de interesse a uma distância a menos ou igual a 600 metros de uma componente desportiva selecionada no mapa e, ainda, no cálculo das rotas, devido à rede viária e à API do MapBox terem o SRID 4326;
- A lógica da aplicação consistiu em pedidos AJAX assíncronos (e alguns síncronos como será visto no seguimento deste capítulo) para o servidor de forma a receber dados espaciais para serem projetados no mapa e/ou a informação sobre cada componente desportiva.

Segue-se então a descrição das funcionalidades do protótipo de alta fidelidade com ilustração de figuras referentes à plataforma.

8.1. Página Inicial

Página de início do site baseado nos protótipos de baixa fidelidade (figura 9), tal como se pode comprovar pelas *sidebars* e *navbar* (figura 23). De referir também que as API's Here Map e Bing Maps Tile foram utilizadas para a cartografia de contexto (estilo do mapa).

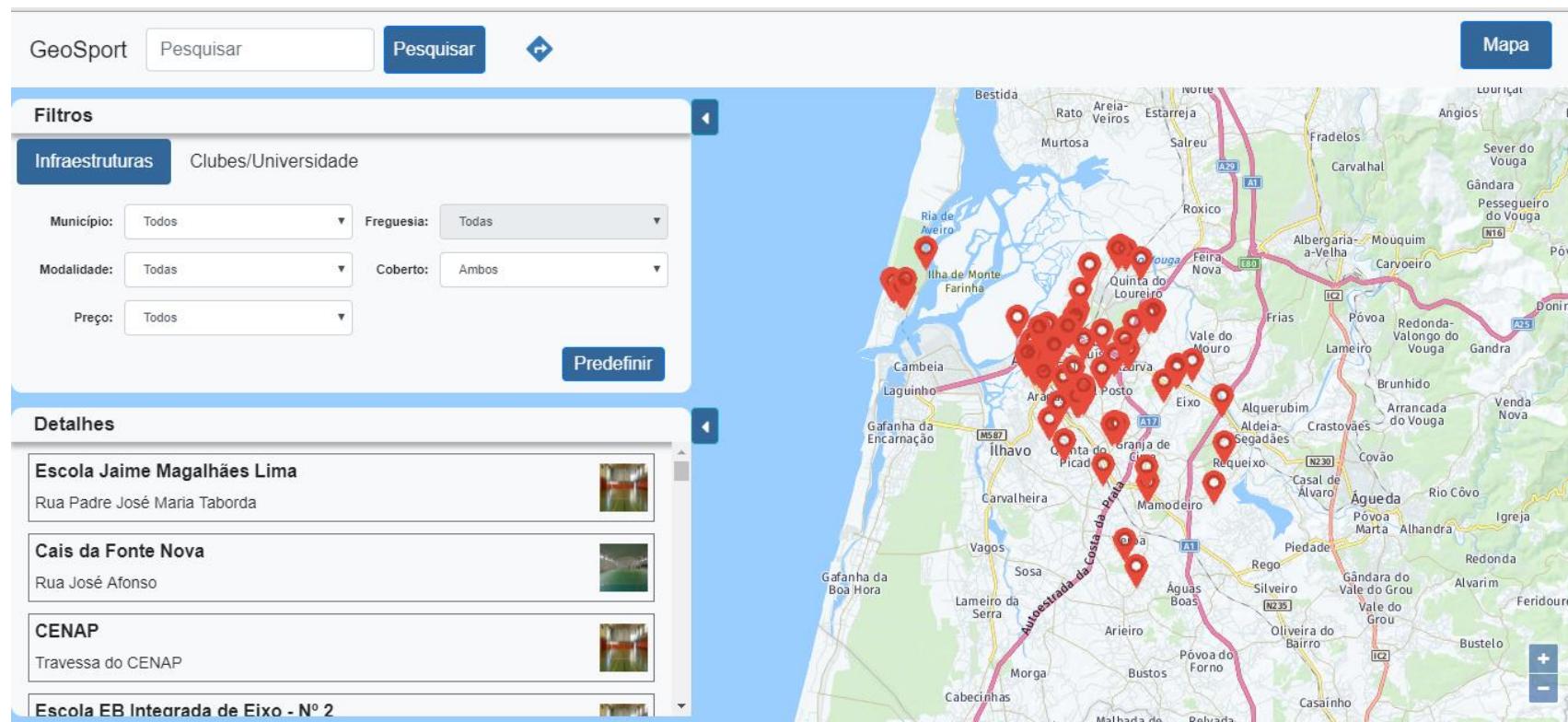


Figura 23 - Apresentação da Página Inicial

8.2. Pesquisa manual

O site apresenta uma pesquisa eficiente pois os resultados apresentados podem apenas conter uma parte do que foi procurado. Para pesquisar por pavilhões é preciso ter em conta que a coluna “Infraestruturas” nos filtros esteja selecionada para que todos os pavilhões estejam no mapa. Caso se queira pesquisar por clubes/universidades deve-se selecionar a coluna “Clubes/Universidade” (figuras 24 e 25).

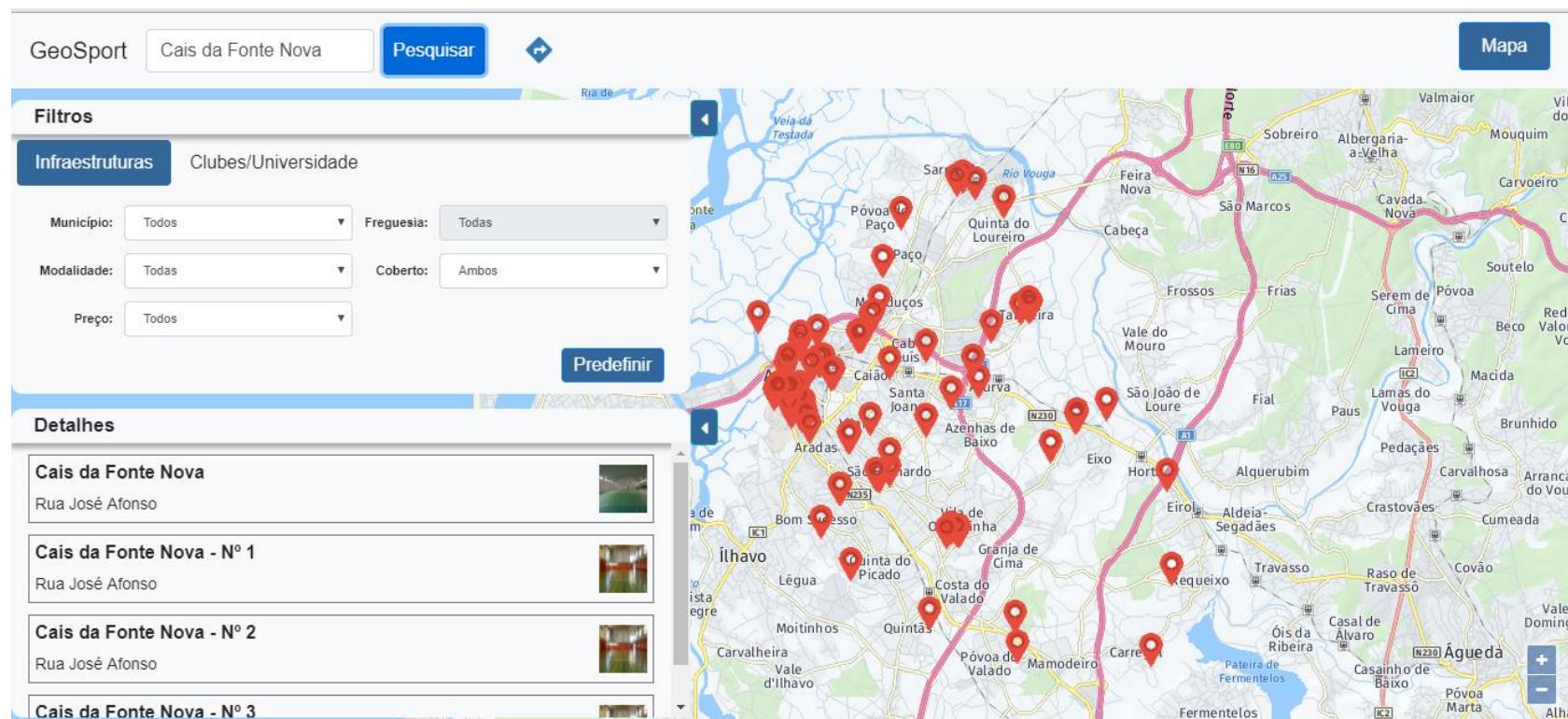


Figura 24 - Apresentação da pesquisa de uma infraestrutura

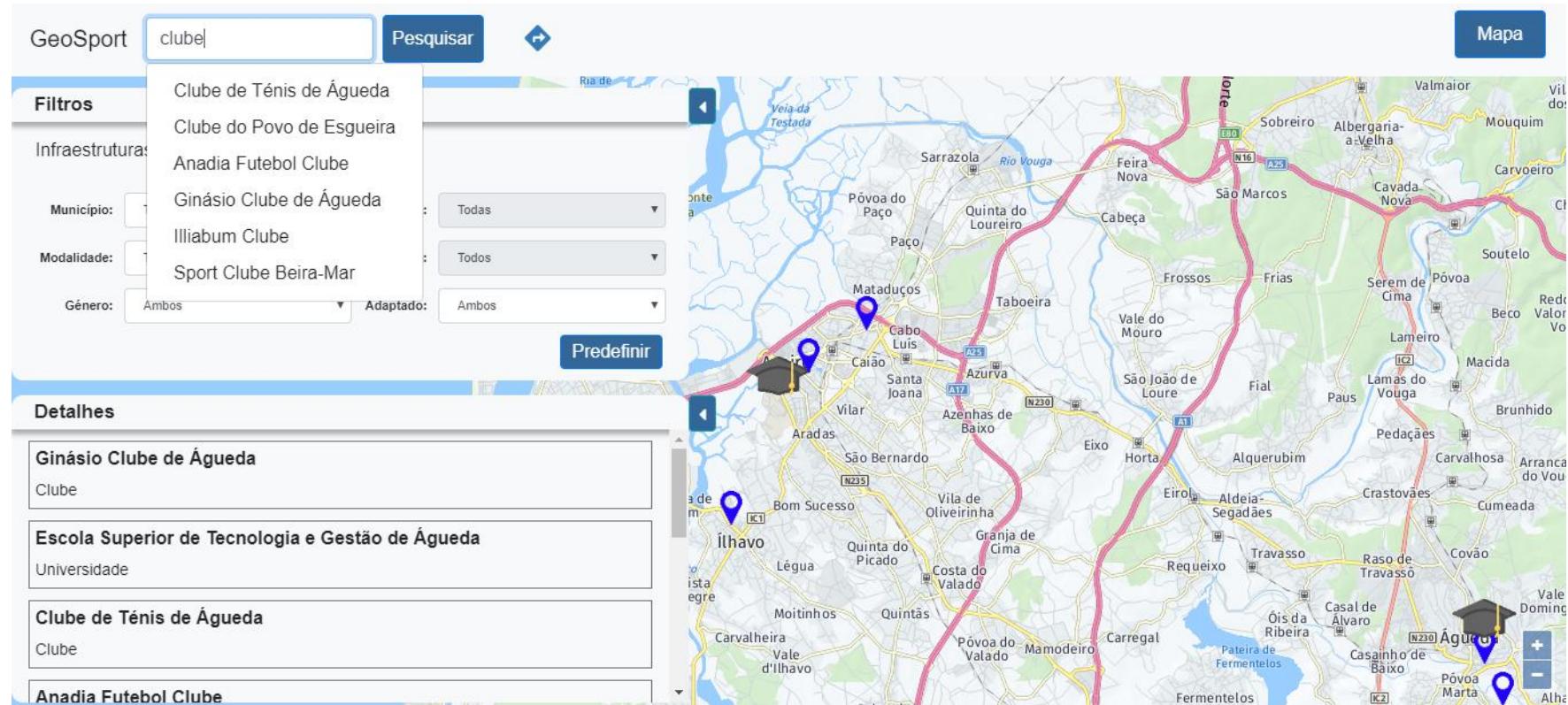


Figura 25 - Apresentação da pesquisa de um clube

8.3. Filtragem

O utilizador tem possibilidade de restringir o que aparece no mapa, isto é, pode filtrar a componente desportiva por vários fatores, quer por município ou freguesia, quer pelos critérios definidos em ambos (infraestruturas e clubes/Universidade). Quanto aos pedidos AJAX, o preenchimento das opções das freguesias e escalões fez-se de forma síncrona para o utilizador aguardar que termine antes de executar outra tarefa. O que estava a acontecer nestes dois casos é que o pedido AJAX de mostrar a componente desportiva filtrada estava a obter o valor anterior antes das opções serem geradas novamente. Imagine-se que a opção selecionada do escalão é “sub-14” e, quando o utilizador seleciona uma nova modalidade são-lhe gerados os escalões associados a essa modalidade. Posteriormente, são-lhe mostrados os clubes/Universidade com qualquer escalão (porque agora a opção é “Todos”), mas na verdade, o que acontecia realmente é que o evento no Javascript estava a obter o valor anterior “sub-14” e arruinava o resultado no mapa devido ao assincronismo. Uma vez que, por vezes, o processo de obtenção dos escalões ou freguesias é demorado, o evento “onchange” ainda obtinha o valor antigo da opção porque o objetivo consistia em mostrar os pontos filtrados sempre que houvesse uma mudança em qualquer uma das opções. Em termos de usabilidade é mais satisfatório não ter um botão para submeter constantemente as opções. Os registos foram devolvidos pelas duas views criadas: “filtros_representacao” e “filtros_eq_desportivos”.

As figuras 26 e 27 consideram a filtragem das infraestruturas e clubes/Universidade, respetivamente. De notar que, para este caso, também se utilizou a função ST_Contains para determinar os pontos contidos num município ou freguesia.

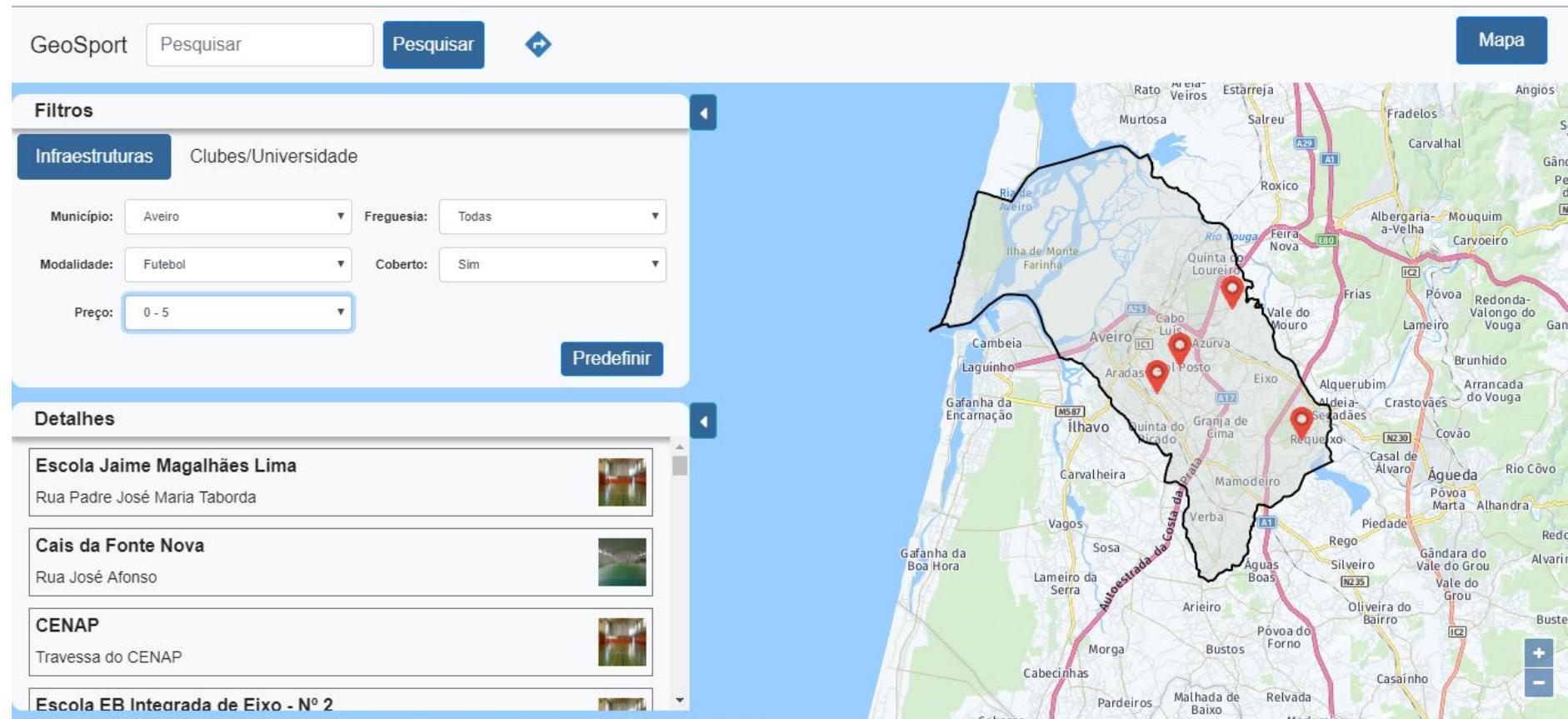


Figura 26 - Filtragem das infraestruturas

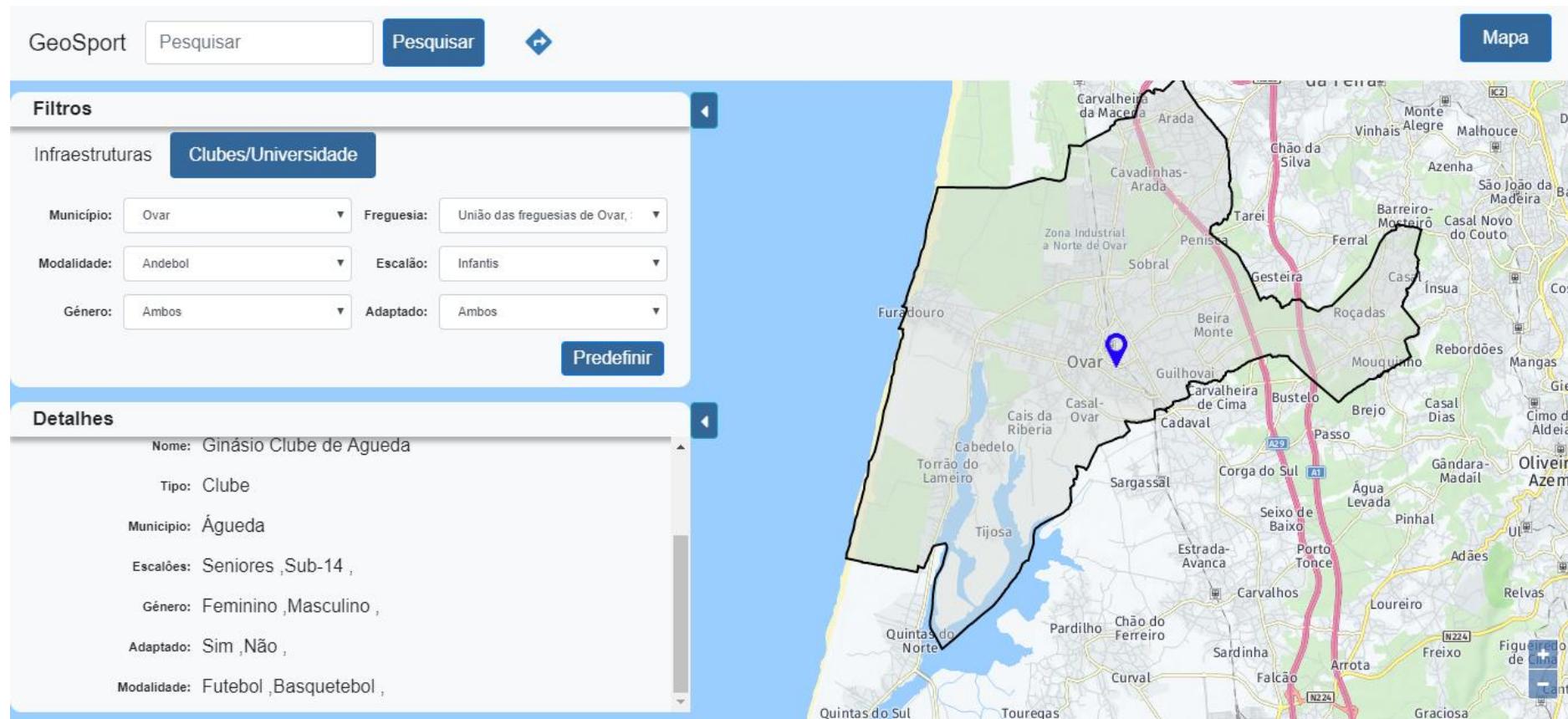


Figura 27 - Filtragem dos clubes/Universidade

8.4. Carregar num ponto ou resultado de pesquisa

Ao carregar num resultado da pesquisa aparecem os detalhes desse ponto e é feito zoom no mapa para o ponto selecionado. Se se selecionar um ponto no mapa aparece um *popup* com pequena informação do ponto (figura 28) e são apresentados mais detalhes no lado esquerdo (figuras 29 e 30).

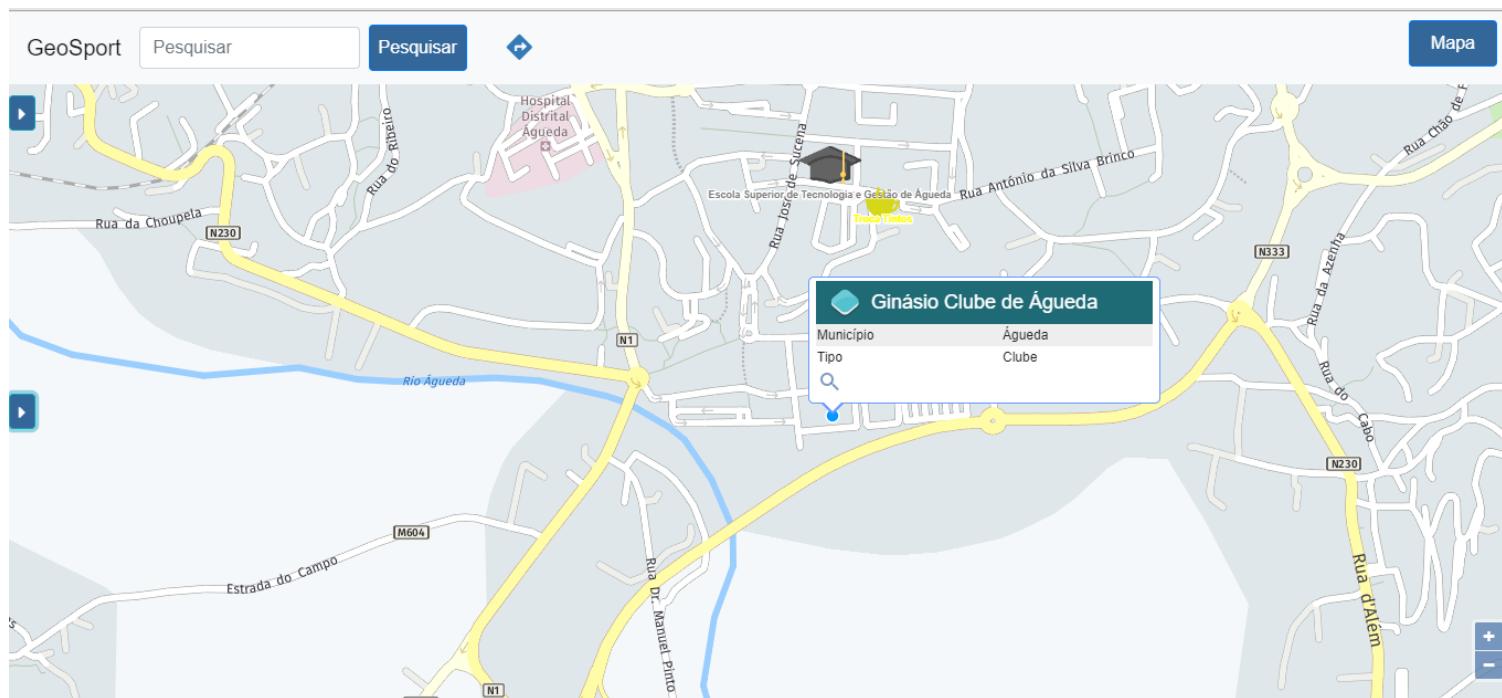


Figura 28 - Popup com informação resumida

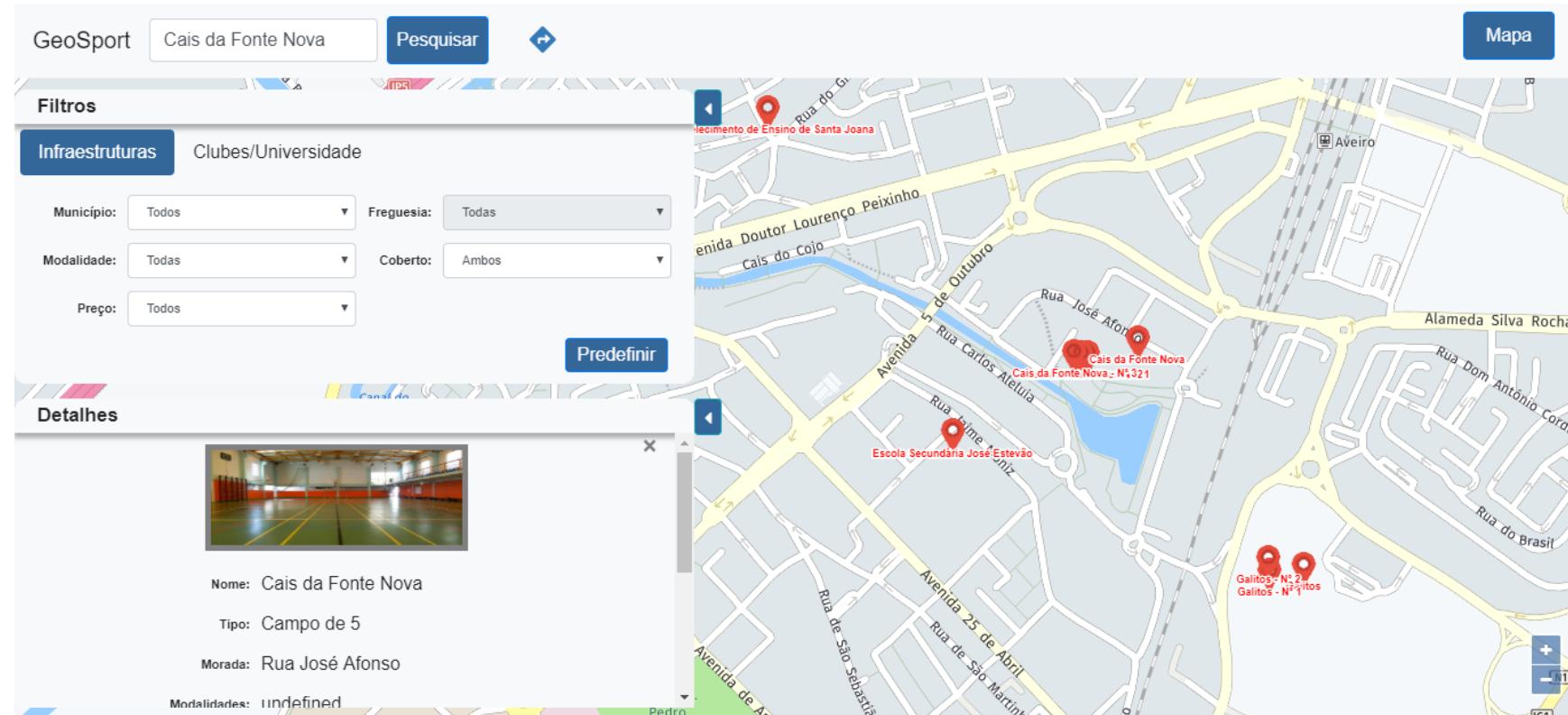


Figura 29 - Detalhes da infraestrutura selecionada

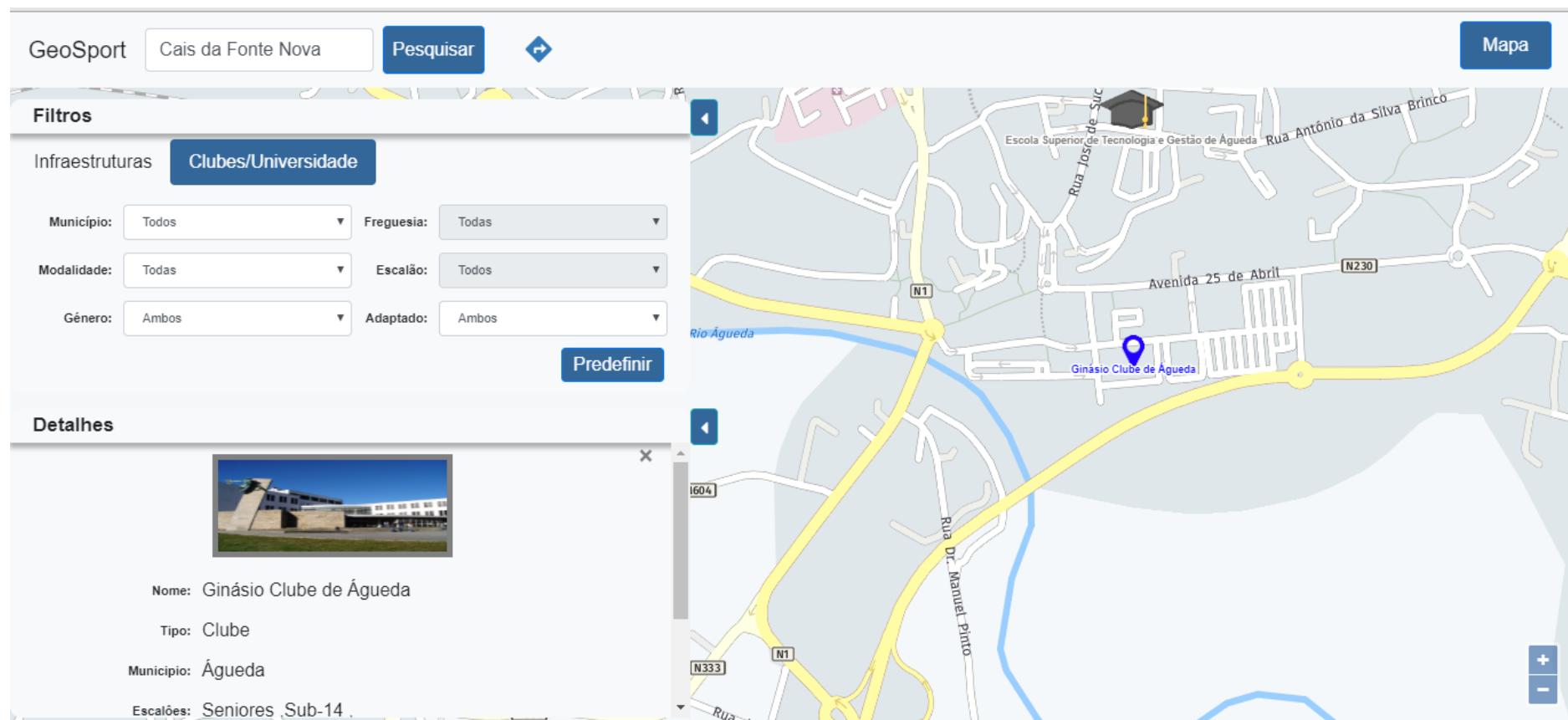


Figura 30 - Detalhes do clube selecionado

8.5. Pontos de interesse

Ao selecionar uma componente desportiva do mapa surgem, caso existam, todos os pontos de interesse (cafés, restaurantes e bares, uma vez que são os únicos tipos de estabelecimentos que estão armazenados na base de dados) a uma distância a menos ou igual a 600 metros (figura 31). Foi então nesta funcionalidade que se utilizou a função implementada que usa o algoritmo Driving Distance, tal como já fora descrito pormenorizadamente no capítulo do modelo de dados persistentes. A figura 32 ilustra o *popup* que surge com o nome e categoria, ao selecionar um ponto de interesse.

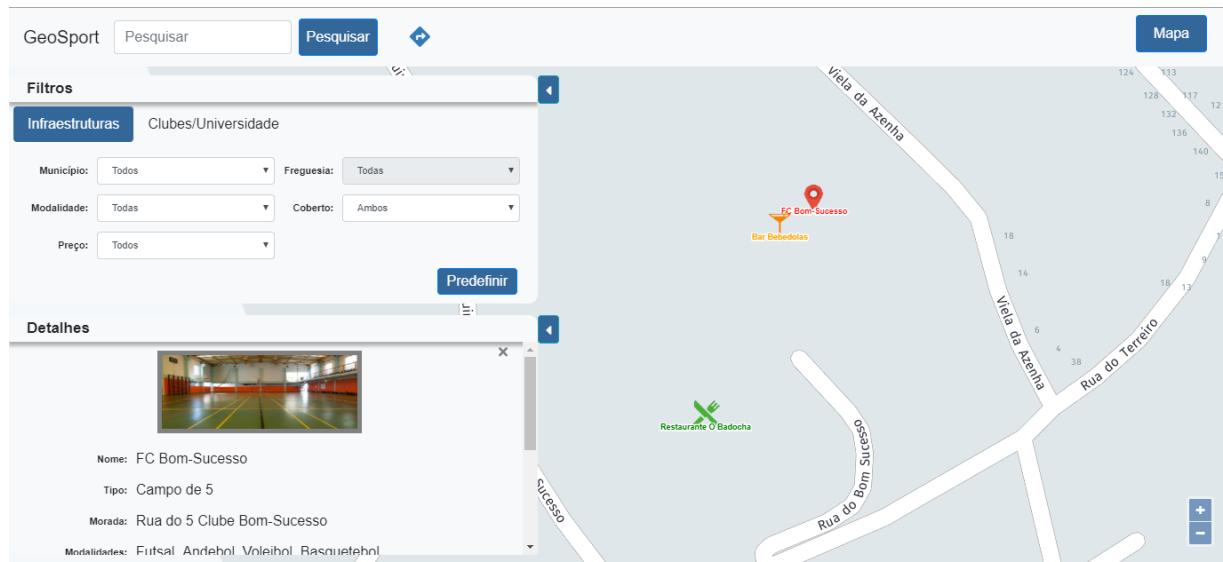


Figura 31 - Pontos de interesse que surgem ao selecionar a infraestrutura

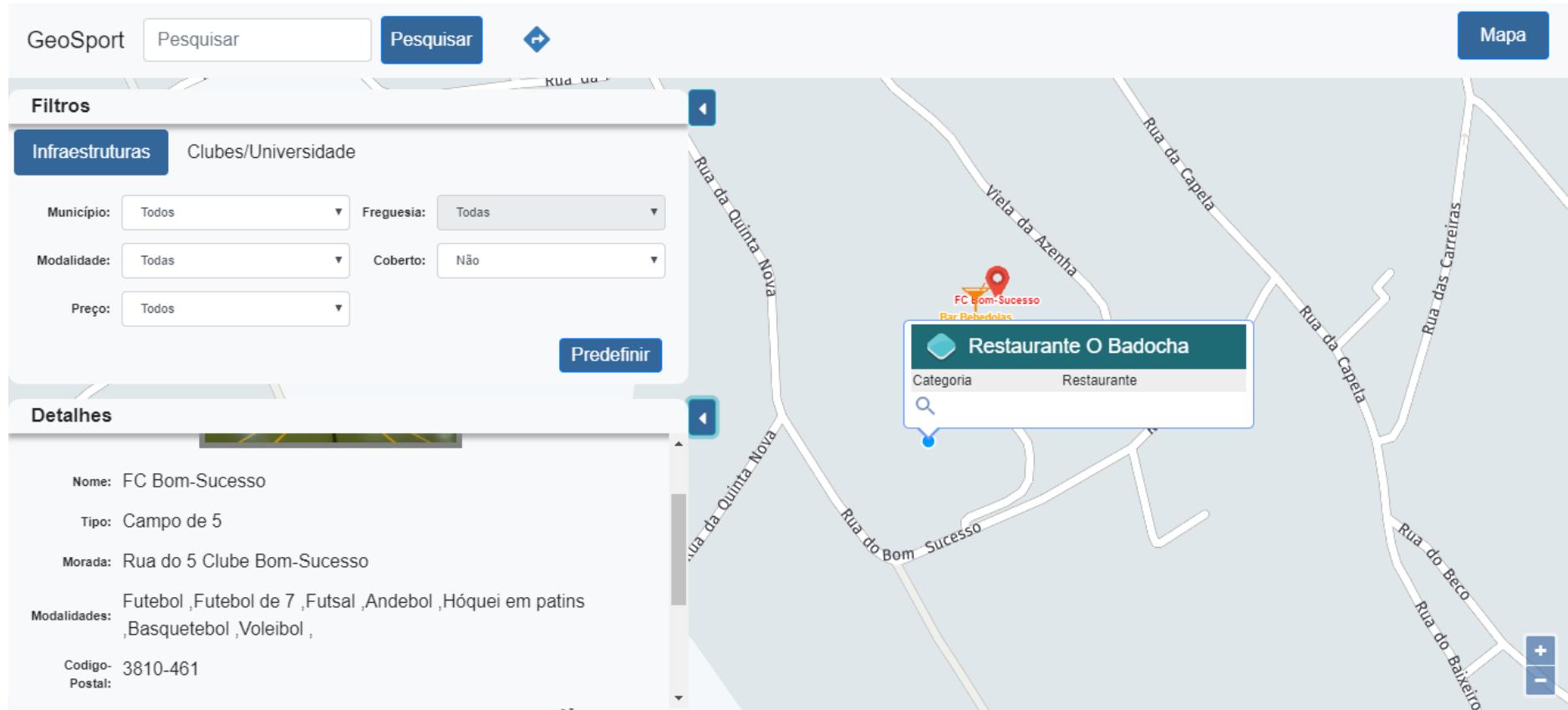


Figura 32 - Popup quando se seleciona um ponto de interesse

8.6. Calcular rotas

Ao clicar no botão direções (junto ao botão pesquisar) a caixa onde estavam a ser exibidos os detalhes dos pavilhões é substituída por uma caixa dedicada à criação das rotas. Nesta caixa temos 3 botões onde podemos selecionar o meio de transporte que vamos utilizar (a pé, de bicicleta ou de carro). Caso nenhuma destas opções seja selecionada o meio de transporte pré-definido é a pé.

Depois disso, o utilizador selecionará no mapa o ponto de partida e o ponto de chegada através de cliques, e, assim que cada um dos pontos seja selecionado será confirmado com um visto, e a rota será calculada automaticamente através da API da MapBox, e o GeoJSON devolvido vai ser utilizado para representar a rota no mapa (figura 33).

Depois de calculada a rota será apresentada dentro da caixa a distância e o tempo necessário para chegar ao destino em minutos e segundos.

Temos também um botão limpar que apaga a rota e os pontos de partida e chegada, assim como o tempo necessário para percorrer o caminho e a distância.

Ao clicar nos botões Clubes/Universidades ou Infraestruturas caso haja uma rota definida ela será automaticamente apagada.

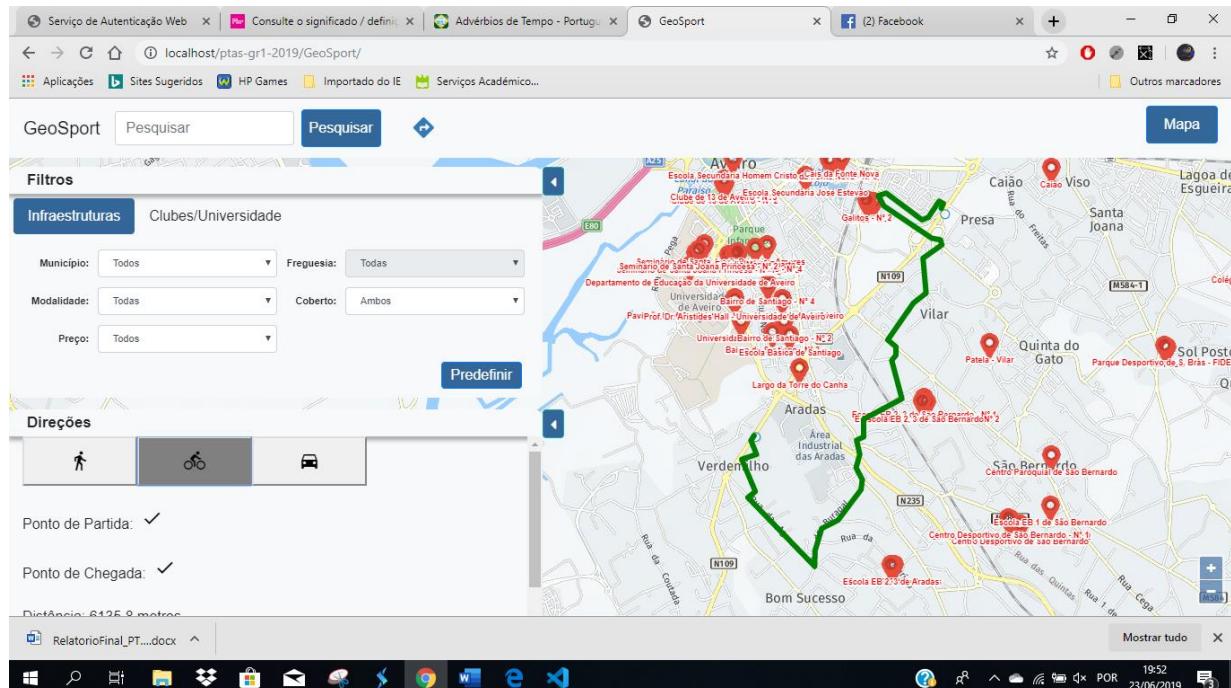


Figura 33 - Cálculo da rota ao selecionar dois pontos no mapa

8.7. Alternar a cartografia de contexto

O botão no canto superior direito permite alternar entre mapa ou satélite (figura 34).

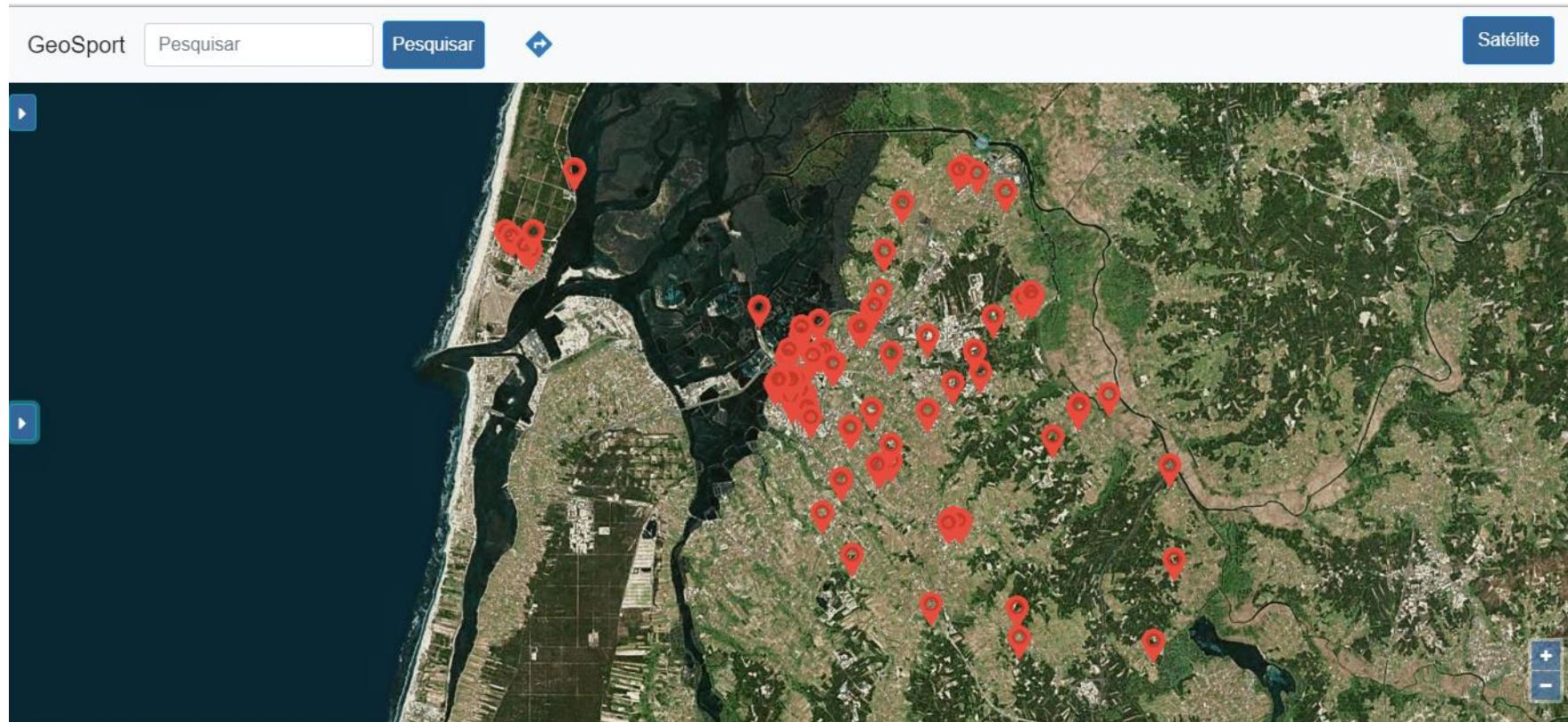


Figura 34 - Vista satélite

8.8. Tratamento de erros

Relativamente ao aparecimento de falhas no ciclo de funcionamento da aplicação, caso ocorra um erro de comunicação entre o cliente e o servidor ou uma falha na ligação do servidor com a base de dados, surgem erros que devem ser tratados. O tratamento desses erros foi feito utilizando-se a biblioteca SweetAlert2 que já possui popup boxes responsivas. A figura 35 ilustra a mensagem de erro que surge quando há um erro de comunicação.

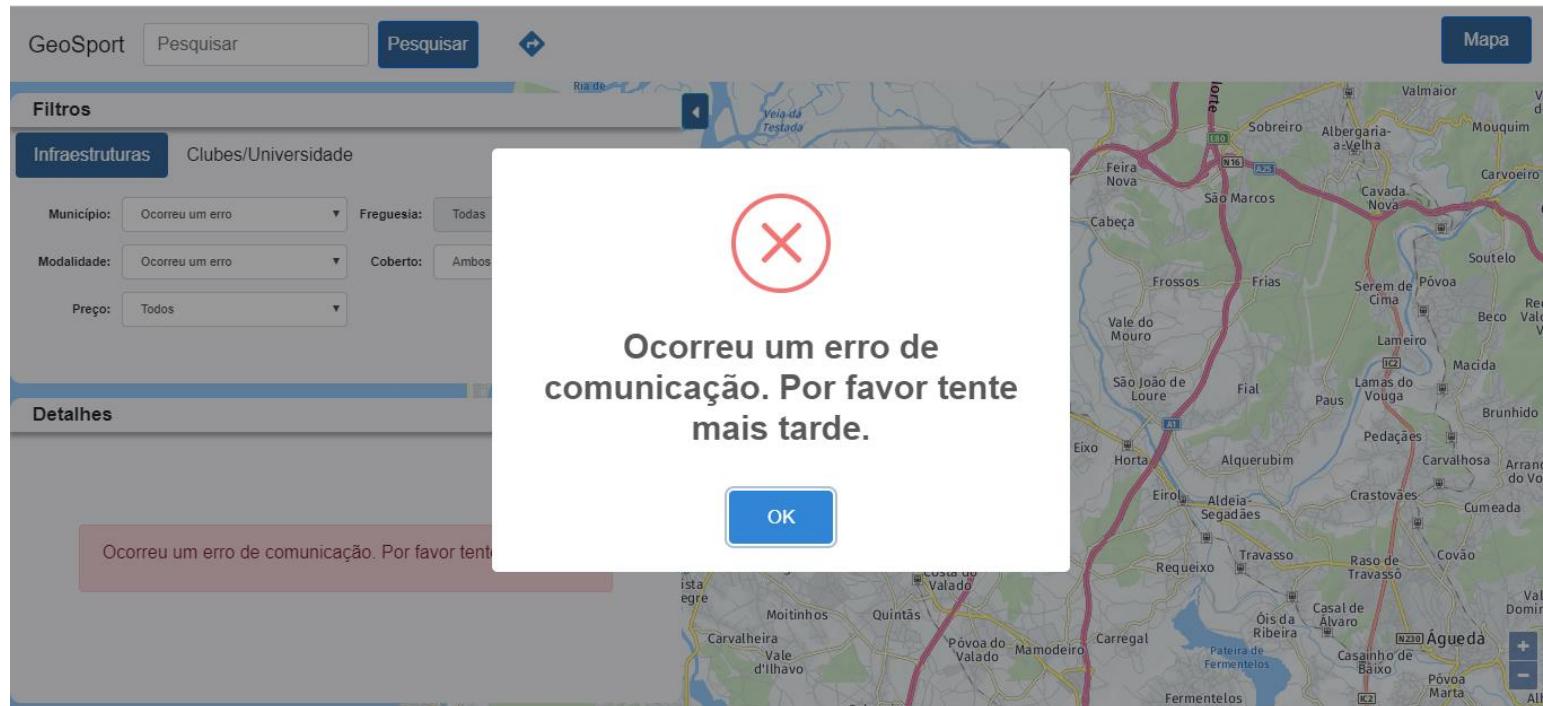


Figura 35 - Mensagem de erro quando ocorre um erro de comunicação

9. Análise de resultados e reflexão crítica

A base de dados foi também desenhada e implementada com sucesso, sendo que está preparada para suportar todas as funcionalidades implementadas.

Considerando que uma boa parte desta informação é praticamente estática, tais como limites de municípios, modalidades, e até as próprias infraestruturas desportivas, esta informação poderia ter sido colocada em ficheiros GeoJSON, o que aceleraria o processo de carregamento da dita informação. Optámos por colocar esta informação numa base de dados por uma questão de flexibilidade e numa perspetiva de ser compatível com futuras funções que possam vir a ser adicionadas no futuro, tais como a autenticação de utilizadores, painel de administração (BackOffice) para visualização de *logs*, e edição/atualização de informação tais como horários e/ou adição de novas infraestruturas, POI's, etc.

A interface tem bastante espaço para melhoria e polimento, principalmente quando vista em ecrãs de reduzidas dimensões tais como Smartphones, assim como a adição de algumas funções que melhorariam a experiência do utilizador como por exemplo o cálculo de rotas com base na localização atual ou autenticação via redes sociais. A tradução do conteúdo da página para a linguagem universal (inglês) seria outra funcionalidade interessante que promoveria a plataforma.

O grau de dificuldade na implementação da filtragem da componente desportiva foi baixo; no entanto, foi um processo demorado devido às imensas combinações para devolver a informação consoante as opções selecionadas.

Outro aspeto inserido neste projeto foi a mostragem de POI's, ao selecionar um ponto existente no mapa. O objetivo desta *feature* é fazer com que, quando um ponto é clicado, o mapa mostra os POI's mais próximos de acordo com uma distância igual ou menos de 600 metros. Graças à entreajuda dos membros do grupo a *feature* foi completada de modo a que o seu objetivo fosse cumprido.

Os requisitos funcionais e não funcionais cumpridos e não cumpridos encontram-se, esquematicamente, nas tabelas 17 e 18, respetivamente. Todos os requisitos foram cumpridos, à exceção dos requisitos não funcionais 3 e 4, referentes à responsividade e tradução da plataforma.

9.1. Requisitos funcionais cumpridos e não cumpridos

A tabela 17 refere os requisitos funcionais cumpridos e não cumpridos.

Tipo - grau de prioridade do requisito (alto, médio, baixo)

Referência - numeração do requisito

Tabela 17 - Requisitos funcionais (RF) cumpridos e não cumpridos

Tipo	Ref^a	Cumprido	Não cumprido
Alto	RF 1	✓	
	RF 2	✓	
	RF 3	✓	
	RF 4	✓	
Médio	RF 5	✓	
Baixo	RF 6	✓	

9.1. Requisitos não funcionais cumpridos e não cumpridos

A tabela 18 considera os requisitos não funcionais que foram cumpridos e não cumpridos.

Tipo - grau de prioridade do requisito (alto, médio, baixo)

Referência - numeração do requisito

Tabela 18 - Requisitos não funcionais (RNF) cumpridos e não cumpridos

Tipo	Ref^a	Cumprido	Não cumprido
Médio	RNF 1	✓	
	RNF 2	✓	
Baixo	RNF 3		✗
	RNF 4		✗

10. Conclusão

O trabalho realizado até à data revelou-se um bom caso de estudo e oportunidade de aprendizagem relativamente ao desenvolvimento de aplicações Web com base nos conceitos aprendidos em Sistemas de Informação Geográfica (SIG). Correspondeu relativamente às expetativas no início do projeto.

A elaboração deste projeto proporcionou uma excelente oportunidade para consolidar temas desenvolvidos ao longo das cadeiras lecionadas, bem como um desafio que maturou uma ética de trabalho já existente dos diversos elementos do grupo.

Apesar das muitas dificuldades encontradas no processo de desenvolvimento deste projeto, através de muita pesquisa e da ajuda do nosso professor e orientador, foi possível superar todos os problemas que foram aparecendo de forma a finalizarmos o trabalho com sucesso, satisfazendo todos os requisitos inicialmente definidos que se consideram de maior importância.

Em termos da estrutura da página, o aspeto mais desafiante foi em conseguir desenvolver as duas *sidebar* do lado esquerdo relativas aos filtros e detalhes. Ainda se tentou criar os elementos de forma a ficarem totalmente responsivos, mas nos ecrãs com pequeníssimas dimensões isso não acontece.

No que respeita à componente SIG da aplicação, o maior desafio foi em implementar o itinerário com o cálculo das rotas, sendo que foi das tarefas mais demoradas. Tentou-se de forma semelhante ao dos pontos de interesse, utilizando os nós e arcos da tabela da rede viária da base de dados; no entanto surgiram algumas complicações em termos de obter corretamente a rota, e optou-se por recorrer a uma API que calcula o trajeto entre dois pontos selecionados com recurso à sua rede viária.

Foi deveras enriquecedor ter oportunidade de desenvolver uma aplicação com esta envergadura, devido à utilidade que pode advir de pessoas que praticam desporto de forma recreativa e atletas. Um grupo de amigos que queira reservar por uma hora um campo a um preço acessível, consegue facilmente encontrar essa informação na plataforma.

11. Bibliografia

- NUNES, Mauro; O'NEILL, Henrique. Fundamental de UML. 6^a ed. FCA
- SILVA, Alberto; VIDEIRA, Carlos. UML, Metodologias e Ferramentas CASE. 2^a ed.
Vol. I. Centroatlantico.pt. Portugal, 2005
- Documentação PostgreSQL: <https://www.postgresql.org/docs/9.6/static/>
- Documentação PostGIS: <https://postgis.net/documentation/>
- Documentação pgRouting: <https://pgrouting.org/documentation.html>
- Documentação jQuery API: <https://api.jquery.com/>
- Documentação Bootstrap: <https://getbootstrap.com/docs/4.1/getting-started/introduction/>
- Documentação FontAwesome: <https://fontawesome.com/start>
- Documentação OpenLayers: <https://openlayers.org/en/latest/doc/>
- Documentação ol-ext: <https://viglino.github.io/ol-ext/doc/doc-pages/>
- Documentação SweetAlert2: <https://sweetalert2.github.io/>
- Documentação Mapbox: <https://docs.mapbox.com/api/navigation/#directions>