



# Guia Autónomo e Personalizado para Apoio ao Turismo

João Ricardo Cristóvão Cardoso

## Orientadores

Professor Doutor Fernando Reinaldo Silva Garcia Ribeiro

Dissertação apresentada à Escola Superior de Tecnologia do Instituto Politécnico de Castelo Branco para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Desenvolvimento de Software e Sistemas Interativos, realizada sob a orientação científica do Doutor Fernando Reinaldo Silva Garcia Ribeiro, do Instituto Politécnico de Castelo Branco.

Dezembro de 2018



## Composição do júri

Presidente do júri

Grau académico, nome do presidente do júri”

Vogais

Grau académico, nome do presidente do júri”

Categoria profissional e o nome da Instituição

Grau académico, nome do presidente do júri”

Categoria profissional e o nome da Instituição

Grau académico, nome do presidente do júri”

Categoria profissional e o nome da Instituição



## Agradecimentos

Ainda que esta dissertação seja um trabalho académico realizado a título individual, toda a minha experiência académica, pessoal e profissional foi muito importante para conseguir chegar a esta fase, bem como o contributo daqueles que me rodeiam, que se revelaram peças importantes no resultado final deste projeto. Assim, pretendo agradecer a todos os que contribuíram para esta etapa, em especial:

Agradeço aos meus pais por todo o carinho, valores e educação que me inculcaram, não esquecendo o esforço que fizeram para me proporcionar sempre o melhor futuro possível. Em especial nesta fase, agradeço as palavras de motivação para chegar ao final desta jornada.

À minha irmã pela paciência e compreensão nas fases mais difíceis do meu percurso académico.

À Laura, por toda a paciência, apoio e trabalho. Mesmo não sendo da área, gastou do seu tempo pessoal e contribuiu para a realização desta dissertação e, conseqüentemente, para o meu sucesso. Sem o apoio dela este trabalho não seria possível.

Aos professores que fizeram parte do meu percurso académico, tanto no curso de Eletrotécnica e Telecomunicações, como no mestrado de Desenvolvimento de Software e Sistemas Interativos.

Aos meus amigos e colegas de trabalho que se mostraram prestáveis para me ajudar no que fosse necessário, o que me permitiu concluir a dissertação.

Aos colegas e amigos com experiência em OutSystems que sempre me ajudaram, em especial ao Ricardo Araújo de Proença-a-Nova que sempre esteve disponível para responder às minhas questões.

Por último, ao meu orientador, uma enorme gratidão: o acompanhamento, tempo pessoal despendido, sugestões e correções, palavras motivacionais e de incentivo nos momentos em que tudo parecia mais complicado devido à dificuldade de conciliar a escrita da dissertação com o meu trabalho. Por toda a preocupação, muito obrigado!

A todos, o meu humilde obrigado!



## Resumo

O crescimento do turismo em Portugal e, conseqüentemente, da oferta de atividades e diversidade de eventos, levou a um crescimento tecnológico que trouxe aos utilizadores a possibilidade de utilizar plataformas que agrupam estas ofertas.

Com base na sua experiência pessoal no planeamento de viagens e atividades, o autor constatou que, apesar da grande oferta de aplicações móveis para este fim, ainda é necessário tempo e esforço para fazer o planeamento das mesmas. Tal deve-se ao facto de, na maior parte das aplicações, a informação sobre eventos não estar completa, as sugestões não serem direcionadas aos interesses do utilizador e não haver uma única aplicação que contemple a maior parte das necessidades do utilizador, obrigando-o a consultar várias aplicações e páginas web.

Nesta dissertação, com o objetivo de agilizar e rentabilizar o processo de gestão de tempo livre e escolha de eventos numa cidade ou região, o autor propõe-se a desenvolver uma aplicação móvel que apresente ao utilizador informação mais completa e personalizada com base nos seus interesses sobre as diversas atividades; que permita gerar planos e percursos automáticos personalizados ao utilizador; encontrar e seguir pessoas com os mesmos interesses; gerir viagens e guardar os planos, eventos e informação adicional; fomentar a comunicação entre utilizadores através de mensagens e notificações internas e possibilidade de consultar conteúdos em modo *offline*.

Para o desenvolvimento da aplicação foi adotada a metodologia ICONIX, por ser uma metodologia ágil e aquela que mais se enquadra no pretendido para este projeto. Seguindo esta metodologia, a aplicação passou pela fase de análise de requisitos, análise e desenho preliminar, projeto e, por fim, fase de implementação. A tecnologia escolhida para a implementação da aplicação foi OutSystems, por ter um leque de ofertas abrangente em conta com o pretendido, permitindo construir desde o *front-end* e o *back-end*, passando por toda a integração com APIs externas. As APIs utilizadas neste projeto foram APIs da *Google*, componentes essenciais no enriquecimento e manipulação de mapas, lugares e eventos.

Em síntese, esta dissertação propõe o desenvolvimento de uma aplicação móvel que contenha toda a informação que o utilizador necessita para planear as suas férias e gerir o seu tempo livre.

## Palavras-chave

Turismo, aplicações móveis, eventos, tempo, smartphone, ICONIX



## **Abstract**

The growth of tourism in Portugal and, consequently, the increase of number of activities and events, accompanied by a technological growth, has brought to customers the possibility of using platforms that combine travel offers and events.

Based on his own experience in traveling and activity planning, the author realized that despite the large number of mobile applications with a similar purpose, planning activities or trips was still difficult and time consuming. This was due to the fact that, in most mobile applications, event information is not complete and recommendations are not directed to the user's interests. Furthermore, there are no applications that aggregate most of the user's needs, having the traveler to consult various applications and web pages.

In this dissertation, with the aim of streamlining and improving the time management and choice of events in a city or region, the author proposes to develop an application that: 1) provides the user with more complete and customized information based on personal interests in various activities; 2) allows the generation of plans and automatic personal routes; 3) allows the user to find and follow people with the same interests; 4) manages trips and saves plans, events and additional information; 5) fosters interactions and communication between users through messages and internal notifications and 6) provides contents in offline mode.

In the development of the application, ICONIX methodology was adopted, due to being an agile methodology and one that most fits in the intended for this project. Following this methodology, the application went through the requirements analysis phase, analysis and preliminary design, project and, finally, implementation phase. The technology chosen for the implementation of the application was OutSystems, that allows the building of the front end and the back end, with a wide range of features, as well as the integration with external APIs. The APIs used in this project were Google APIs, essential components in enriching and manipulating maps, places, and events.

In summary, the aim of this dissertation is the development of a mobile application that aggregates all the information that the user needs to plan his vacations and manage his free time.

## **Keywords**

Turism, mobile apps, events, time, smartphone, ICONIX



# Índice Geral

COMPOSIÇÃO DO JÚRI.....	III
AGRADECIMENTOS .....	V
RESUMO .....	VII
ABSTRACT .....	IX
ÍNDICE GERAL .....	XI
ÍNDICE DE FIGURAS .....	XIII
ÍNDICE DE TABELAS .....	XVII
LISTA DE ABREVIATURAS .....	XIX
<b>1. INTRODUÇÃO .....</b>	<b>1</b>
1.1. CONTEXTUALIZAÇÃO E MOTIVAÇÃO .....	1
1.2. OBJETIVO .....	2
1.3. CRONOGRAMA .....	2
1.4. ESTRUTURA DO RELATÓRIO .....	3
<b>2. ESTADO DA ARTE .....</b>	<b>5</b>
2.1. APLICAÇÕES EM ESTUDO.....	5
2.1.1. <i>Google Trips – Travel</i> .....	6
2.1.2. <i>GuiaDeRodas</i> .....	8
2.1.3. <i>JiTT.travel</i> .....	11
2.1.4. <i>Minube</i> .....	15
2.1.5. <i>TripCase</i> .....	17
2.2. ANÁLISE COMPARATIVA.....	19
<b>3. METODOLOGIA ADOTADA - ICONIX.....</b>	<b>21</b>
3.1. ENQUADRAMENTO .....	21
3.2. ICONIX.....	22
3.2.1. <i>Análise de Requisitos</i> .....	23
3.2.2. <i>Análise e Desenho Preliminar</i> .....	24
3.2.3. <i>Projeto</i> .....	24
3.2.4. <i>Implementação</i> .....	25
<b>4. MODELAÇÃO DO SISTEMA.....</b>	<b>26</b>
4.1. ANÁLISE DE REQUISITOS.....	26
4.1.1. <i>Requisitos</i> .....	26
4.1.2. <i>Modelo de Domínio</i> .....	28
4.1.3. <i>Interface Gráfica</i> .....	28
4.1.4. <i>Diagramas de Navegação</i> .....	37
4.1.5. <i>Casos de Uso</i> .....	40
4.2. ANÁLISE E DESENHO PRELIMINAR.....	44
4.2.1. <i>Descrição dos Atores</i> .....	45
4.2.2. <i>Descrição dos Casos de Uso</i> .....	47
4.2.3. <i>Diagramas de Robustez</i> .....	70
4.2.4. <i>Atualização do Modelo de Domínio</i> .....	73
4.3. PROJETO.....	74
4.3.1. <i>Modelação de Base de Dados</i> .....	74
4.3.2. <i>Tabelas de Dados</i> .....	79

4.3.3.	<i>Tabelas de Dados de Referência</i> .....	83
4.3.4.	<i>Diagramas de Sequência</i> .....	85
4.3.5.	<i>Atualização do Modelo de Domínio – Diagrama de Classes</i> .....	92
<b>5.</b>	<b>ARQUITETURA E TECNOLOGIAS</b> .....	<b>94</b>
5.1.	ARQUITETURA.....	94
5.1.1.	<i>Camada de Apresentação – GUI</i> .....	96
5.1.2.	<i>Camada de Lógica de Negócio</i> .....	96
5.1.3.	<i>Camada de Acesso a Dados</i> .....	96
5.1.4.	<i>Camada de Acesso ao Servidor e Entidades Externas</i> .....	96
5.1.5.	<i>Escalabilidade</i> .....	96
5.2.	TECNOLOGIAS USADAS.....	97
5.2.1.	<i>OutSystems</i> .....	97
5.2.2.	<i>APIs da Google</i> .....	98
<b>6.</b>	<b>PROTÓTIPO</b> .....	<b>100</b>
6.1.	PROTÓTIPO DA APLICAÇÃO MÓVEL .....	100
6.1.1.	<i>Visitante</i> .....	100
6.1.2.	<i>Utilizador</i> .....	102
6.1.3.	<i>Organizador</i> .....	115
6.1.4.	<i>Revisor</i> .....	120
6.2.	LÓGICA DE NEGÓCIO.....	122
6.2.1.	<i>Cálculo da Relevância do Evento</i> .....	122
6.2.2.	<i>Sincronização Aplicação e Servidor</i> .....	124
6.2.3.	<i>Cálculo do Plano Automático</i> .....	126
6.2.4.	<i>Processos e Timers</i> .....	127
6.3.	ROTEIROWEB – ASSISTENTE PARA ADMINISTRAÇÃO .....	128
<b>7.</b>	<b>CONCLUSÃO</b> .....	<b>132</b>
7.1.	CONSIDERAÇÕES FINAIS.....	132
7.2.	TRABALHO FUTURO .....	133
<b>8.</b>	<b>BIBLIOGRAFIA</b> .....	<b>135</b>
<b>9.</b>	<b>ANEXOS</b> .....	<b>138</b>
I.	TABELAS USADAS.....	138
II.	DIAGRAMAS DE SEQUÊNCIA.....	144
III.	DIAGRAMAS DE ROBUSTEZ .....	155

# Índice de Figuras

<b>FIGURA 1</b> - LISTA DE VIAGENS FUTURAS.....	6
<b>FIGURA 2</b> - LISTA DE VIAGENS PASSADAS.....	6
<b>FIGURA 3</b> - DETALHE DA VIAGEM.....	7
<b>FIGURA 4</b> - LISTAS DE CATEGORIAS.....	7
<b>FIGURA 5</b> - DETALHE DO EVENTO.....	7
<b>FIGURA 6</b> - DETALHE DO EVENTO.....	7
<b>FIGURA 7</b> - COMENTÁRIOS AO EVENTO.....	7
<b>FIGURA 8</b> - PERCURSO GERADO AUTOMATICAMENTE.....	8
<b>FIGURA 9</b> - DETALHE DE UM PERCURSO.....	8
<b>FIGURA 10</b> - INFORMAÇÕES ÚTEIS.....	8
<b>FIGURA 11</b> - LOADING DA APLICAÇÃO.....	9
<b>FIGURA 12</b> - ECRÃ DE LOGIN/REGISTO.....	9
<b>FIGURA 13</b> - LISTA DE PONTOS DE INTERESSE.....	10
<b>FIGURA 14</b> - RESUMO DE UM PONTO DE INTERESSE.....	10
<b>FIGURA 15</b> - LOCALIZAÇÃO DE UM PONTO DE INTERESSE COM ACESSIBILIDADE.....	10
<b>FIGURA 16</b> - LOCALIZAÇÃO DE UM PONTO DE INTERESSE COM ACESSIBILIDADE PARCIAL.....	11
<b>FIGURA 17</b> - DETALHE DO PONTO DE INTERESSE.....	11
<b>FIGURA 18</b> - ECRÃ DE LOGIN OU REGISTO.....	12
<b>FIGURA 19</b> - ÍDIOMAS DE PREFERÊNCIA.....	12
<b>FIGURA 20</b> - LISTA DE GUIAS DE LISBOA.....	12
<b>FIGURA 21</b> - DETALHE DE UM GUIA.....	13
<b>FIGURA 22</b> - PONTOS DO GUIA.....	13
<b>FIGURA 23</b> - OPÇÕES DO GUIA.....	13
<b>FIGURA 24</b> - RESUMO DO PONTO DE INTERESSE DO GUIA.....	14
<b>FIGURA 25</b> - DETALHE DO PONTO DE INTERESSE DO GUIA.....	14
<b>FIGURA 26</b> - MENU DA JITT.TRAVEL.....	14
<b>FIGURA 27</b> - ECRÃ PRINCIPAL DA APLICAÇÃO MINUBE.....	15
<b>FIGURA 28</b> - SELECIONAR LOCAL.....	15
<b>FIGURA 29</b> - PONTOS DE INTERESSE PARA LISBOA.....	15
<b>FIGURA 30</b> - PONTO DE INTERESSE.....	16
<b>FIGURA 31</b> - DETALHE DE UM PONTO DE INTERESSE.....	16
<b>FIGURA 32</b> - COMENTÁRIO E AVALIAÇÃO DO LOCAL.....	16
<b>FIGURA 33</b> - PÁGINA PRINCIPAL DA APLICAÇÃO TRIP CASE.....	17
<b>FIGURA 34</b> - LISTA DE VIAGENS.....	17
<b>FIGURA 35</b> - LISTA DE ITENS PARA UMA VIAGEM.....	17
<b>FIGURA 36</b> - ADICIONAR ITEM À VIAGEM.....	18
<b>FIGURA 37</b> - TIMELINE DA VIAGEM.....	18
<b>FIGURA 38</b> - DETALHE DE UM ITEM DA VIAGEM.....	18
<b>FIGURA 39</b> - ICONIX (ROSENBERG, STEPHENS AND COLLINS-COPE, 2005).....	23
<b>FIGURA 40</b> - MODELO DE DOMÍNIO.....	28
<b>FIGURA 41</b> - ECRÃ ILUSTRATIVO DE LOGIN OU REGISTO.....	29
<b>FIGURA 42</b> - ECRÃ ILUSTRATIVO DO MENU.....	30
<b>FIGURA 43</b> - ECRÃ ILUSTRATIVO DO DASHBOARD.....	30
<b>FIGURA 44</b> - ECRÃ ILUSTRATIVO DA LISTA DE EVENTOS.....	32
<b>FIGURA 45</b> - ECRÃ DE CRIAÇÃO DO EVENTO (3 FASES).....	33
<b>FIGURA 46</b> - ECRÃ DE APROVAÇÃO OU REJEIÇÃO DE EVENTOS.....	34
<b>FIGURA 47</b> - ECRÃ DO DETALHE DE UM EVENTO.....	35
<b>FIGURA 48</b> - ECRÃ DE PLANEAMENTO DE ATIVIDADES – VISÃO DO MAPA.....	36
<b>FIGURA 49</b> - ECRÃ DE PLANEAMENTO DE ATIVIDADES - VISÃO DO ITINERÁRIO.....	36

<b>FIGURA 50</b> - DIAGRAMA DE NAVEGAÇÃO DO UTILIZADOR.....	38
<b>FIGURA 51</b> - DIAGRAMA DE NAVEGAÇÃO DO ORGANIZADOR .....	39
<b>FIGURA 52</b> - DIAGRAMA DE NAVEGAÇÃO DO REVISOR.....	40
<b>FIGURA 53</b> - INTERAÇÃO DOS ATORES COM O SISTEMA - VISÃO GLOBAL .....	41
<b>FIGURA 54</b> - DIAGRAMA DE CASOS DE USO - VISITANTE .....	41
<b>FIGURA 55</b> - DIAGRAMA DE CASOS DE USO - UTILIZADOR .....	42
<b>FIGURA 56</b> - DIAGRAMA DE CASOS DE USO - ORGANIZADOR .....	43
<b>FIGURA 57</b> - DIAGRAMA DE CASOS DE USO - REVISOR.....	44
<b>FIGURA 58</b> - DIAGRAMA DE ROBUSTEZ - VER SUGESTÕES.....	71
<b>FIGURA 59</b> - DIAGRAMA DE ROBUSTEZ – FILTRAR.....	71
<b>FIGURA 60</b> - DIAGRAMA DE ROBUSTEZ - VER VIAGENS.....	72
<b>FIGURA 61</b> - DIAGRAMA DE ROBUSTEZ - VER DETALHE DO EVENTO .....	72
<b>FIGURA 62</b> - DIAGRAMA DE ROBUSTEZ – REJEITAR.....	73
<b>FIGURA 63</b> - MODELO DE DOMÍNIO ATUALIZADO .....	73
<b>FIGURA 64</b> - MODELO E/R.....	75
<b>FIGURA 65</b> - MODELO RELACIONAL.....	78
<b>FIGURA 66</b> - DIAGRAMA DE SEQUÊNCIA - VER SUGESTÕES.....	87
<b>FIGURA 67</b> - DIAGRAMA DE SEQUÊNCIA – FILTRAR.....	88
<b>FIGURA 68</b> - DIAGRAMA DE SEQUÊNCIA - A PARTICIPAR NO EVENTO .....	89
<b>FIGURA 69</b> - DIAGRAMA DE SEQUÊNCIA - VER VIAGENS.....	90
<b>FIGURA 70</b> - DIAGRAMA DE SEQUÊNCIA - PUBLICAR OU EDITAR EVENTO.....	91
<b>FIGURA 71</b> - DIAGRAMA DE SEQUÊNCIA – REJEITAR .....	92
<b>FIGURA 72</b> - MODELO DE DOMÍNIO FINAL – DIAGRAMA DE CLASSES.....	93
<b>FIGURA 73</b> - EXEMPLO DE UM MODELO DE 4 CAMADAS (MENEZES AND OUTSYSTEMS, 2012) .....	95
<b>FIGURA 74</b> - ARQUITETURA DA APLICAÇÃO .....	95
<b>FIGURA 75</b> - PÁGINA DE REGISTO .....	101
<b>FIGURA 76</b> - ECRÃ DE LOGIN .....	102
<b>FIGURA 77</b> - MENU.....	102
<b>FIGURA 78</b> - ECRÃ LISTA DE EVENTOS SUGERIDOS.....	103
<b>FIGURA 79</b> - FILTRAR EVENTOS .....	103
<b>FIGURA 80</b> - ECRÃ LISTA DE EVENTOS SUGERIDOS (FILTRO) .....	104
<b>FIGURA 81</b> - PROCURAR EVENTO .....	104
<b>FIGURA 82</b> - MAIS OPÇÕES NA LISTA DE EVENTOS .....	105
<b>FIGURA 83</b> - VISTA MAPA DE TODOS OS EVENTOS .....	105
<b>FIGURA 84</b> - VISTA MAPA SATÉLITE .....	106
<b>FIGURA 85</b> - TRAJETO PARA UM EVENTO.....	106
<b>FIGURA 86</b> - DETALHE DO EVENTO.....	107
<b>FIGURA 87</b> - DETALHE DO EVENTO.....	107
<b>FIGURA 88</b> - POPUP NOVA IMAGEM .....	108
<b>FIGURA 89</b> - OPÇÕES DE CARREGAMENTO DE NOVA IMAGEM .....	108
<b>FIGURA 90</b> - VISUALIZAÇÃO DA IMAGEM.....	109
<b>FIGURA 91</b> - COMENTÁRIOS DO EVENTO.....	109
<b>FIGURA 92</b> - MAPA EVENTO ÚNICO.....	110
<b>FIGURA 93</b> - DASHBOARD UTILIZADOR.....	110
<b>FIGURA 94</b> - UTILIZADORES SEGUIDOS .....	111
<b>FIGURA 95</b> - UTILIZADORES COM OS MESMOS INTERESSES .....	111
<b>FIGURA 96</b> - VIAGENS DO UTILIZADOR.....	112
<b>FIGURA 97</b> - DETALHE DE VIAGEM.....	112
<b>FIGURA 98</b> - CONVERSAS DO UTILIZADOR .....	113
<b>FIGURA 99</b> - DETALHE DE UMA CONVERSA .....	113
<b>FIGURA 100</b> - PLANOS DO UTILIZADOR .....	114
<b>FIGURA 101</b> - PLANOS DE UTILIZADOR (MAPA).....	114

<b>FIGURA 102</b> - PERFIL.....	115
<b>FIGURA 103</b> - MENU DO ORGANIZADOR.....	116
<b>FIGURA 104</b> - ECRÃ DESTAQUE DO EVENTO .....	118
<b>FIGURA 105</b> - ECRÃ DETALHES DO EVENTO.....	118
<b>FIGURA 106</b> - ECRÃ LOCALIZAÇÃO DO EVENTO.....	118
<b>FIGURA 107</b> - ECRÃ LOCALIZAÇÃO DO EVENTO.....	118
<b>FIGURA 108</b> - RELATÓRIO DO ORGANIZADOR.....	119
<b>FIGURA 109</b> - DASHBOARD DO REVISOR.....	120
<b>FIGURA 110</b> - ECRÃ EVENTOS PENDENTES (REJECT MESSAGE) .....	121
<b>FIGURA 111</b> - ECRÃ APPROVAL .....	121
<b>FIGURA 112</b> - HOMEPAGE ROTEIROWEB .....	129
<b>FIGURA 113</b> - LISTA DE ATORES .....	130
<b>FIGURA 114</b> - ECRÃ DE DETALHE E EDIÇÃO DO ATOR .....	130
<b>FIGURA 115</b> - LISTA DE EVENTOS.....	131



# Índice de Tabelas

<b>TABELA 1</b> - CRONOGRAMA DO PLANO DE TRABALHO.....	3
<b>TABELA 2</b> - ANÁLISE COMPARATIVA DAS APLICAÇÕES.....	19
<b>TABELA 3</b> - METODOLOGIAS ÁGEIS VS METODOLOGIAS TRADICIONAIS.....	21
<b>TABELA 4</b> - CASO DE USO REGISTRAR.....	47
<b>TABELA 5</b> - CASO DE USO LOGIN .....	48
<b>TABELA 6</b> - CASO DE USO RECUPERAR PALAVRA-PASSE .....	48
<b>TABELA 7</b> - CASO DE USO VER SUGESTÕES .....	49
<b>TABELA 8</b> - CASO DE USO PROCURAR EVENTO.....	50
<b>TABELA 9</b> - CASO DE USO FILTRAR .....	50
<b>TABELA 10</b> - CASO DE USO VER DETALHE DE EVENTO.....	51
<b>TABELA 11</b> - CASO DE USO MOSTRAR INTERESSE.....	51
<b>TABELA 12</b> - CASO DE USO COMENTAR / VER COMENTÁRIOS .....	52
<b>TABELA 13</b> - CASO DE USO AVALIAR / VER AVALIAÇÕES.....	52
<b>TABELA 14</b> - CASO DE USO ADICIONAR IMAGENS / VER IMAGENS .....	53
<b>TABELA 15</b> - CASO DE USO VER LOCALIZAÇÃO .....	54
<b>TABELA 16</b> - CASO DE USO PLANEAR.....	54
<b>TABELA 17</b> - CASO DE USO EDITAR PLANO .....	55
<b>TABELA 18</b> - CASO DE USO PARTICIPAR NO EVENTO.....	56
<b>TABELA 19</b> - CASO DE USO "EU ESTIVE" NO EVENTO .....	56
<b>TABELA 20</b> - CASO DE USO VER VIAGENS.....	57
<b>TABELA 21</b> - CASO DE USO CONSULTAR VIAGEM.....	58
<b>TABELA 22</b> - CASO DE USO VER MENSAGENS .....	58
<b>TABELA 23</b> - CASO DE USO CONVERSAR .....	59
<b>TABELA 24</b> - CASO DE USO PROCURAR UTILIZADORES SEGUIDOS .....	60
<b>TABELA 25</b> - CASO DE USO PROCURAR UTILIZADORES COM OS MESMOS INTERESSES .....	60
<b>TABELA 26</b> - CASO DE USO CONSULTAR PERFIL .....	61
<b>TABELA 27</b> - CASO DE USO VER PERFIL DE OUTRO UTILIZADOR.....	61
<b>TABELA 28</b> - CASO DE USO SEGUIR UTILIZADOR .....	62
<b>TABELA 29</b> - CASO DE USO ALTERAR FOTO.....	62
<b>TABELA 30</b> - CASO DE USO ALTERAR DADOS DO PERFIL.....	63
<b>TABELA 31</b> - CASO DE USO CRIAR EVENTO/PUBLICAR.....	64
<b>TABELA 32</b> - CASO DE USO EVENTOS REJEITADOS .....	65
<b>TABELA 33</b> - CASO DE USO EVENTOS FAVORITO (POPULARES) .....	65
<b>TABELA 34</b> - CASO DE USO PUBLICITAR EVENTO .....	66
<b>TABELA 35</b> - CASO DE USO EDITAR EVENTO .....	67
<b>TABELA 36</b> - CASO DE USO VER RELATÓRIO .....	67
<b>TABELA 37</b> - CASO DE USO VER EVENTOS PENDENTES DE APROVAÇÃO.....	68
<b>TABELA 38</b> - CASO DE USO ACEITAR EVENTO.....	69
<b>TABELA 39</b> - CASO DE USO REJEITAR EVENTO.....	69
<b>TABELA 40</b> - TABELA DE EVENTOS .....	79
<b>TABELA 41</b> - TABELA DE IMAGENS .....	81
<b>TABELA 42</b> - TABELA DE RELAÇÃO UTILIZADOR VS PONTO DE INTERESSE .....	82
<b>TABELA 43</b> - TABELA DE PREFERÊNCIAS .....	82
<b>TABELA 44</b> - ESTRUTURA DAS TABELAS DE DADOS DE REFERÊNCIA .....	83
<b>TABELA 45</b> - MENSAGENS DE ERRO.....	84
<b>TABELA 46</b> - DADOS DE REFERENCIA, EVENTO UTILIZADOR .....	84
<b>TABELA 47</b> - DADOS DE REFERENCIA, UTILIZADOR FUNÇÃO.....	84
<b>TABELA 48</b> - DADOS DE REFERENCIA, ESTADO.....	85
<b>TABELA 49</b> - SOLICITAÇÕES, ERROS E LATÊNCIA MÉDIA PARA CADA API NOS ÚLTIMOS 30 DIAS.....	99

<b>TABELA 50</b> - PONTUAÇÃO ATRIBUÍDA A CATEGORIAS.....	123
<b>TABELA 51</b> - RELEVÂNCIA DA DISTÂNCIA AO PONTO DE INTERESSE.....	124
<b>TABELA 52</b> - CONFIGURAÇÕES PARA SINCRONISMO .....	126

## Lista de Abreviaturas

**API** *Application Programming Interface*

**GPS** *Global Positioning System*

**GUI** *Graphical User Interface*

**IDE** *Integrated Development Environment*

**REST** *Representational State Transfer*

**RUP** *Rational Unified Process*

**SDK** *Software Development Kit*

**UML** *Unified Modeling Language*

**XP** *Extreme Programming*



# 1. Introdução

Atualmente, o Turismo em Portugal é uma atividade em crescimento e um dos principais setores de proveito para a economia nacional. No último ano, registou-se um aumento de emprego no turismo (+ 44 mil), com um peso de 7% na economia nacional (Turismo de Portugal, 2018).

Em 2017, o país foi eleito "Melhor Destino Turístico do Mundo" nos World Travel Awards. Lisboa foi considerada como o "Melhor Destino para *City Break*" e a ilha da Madeira "o Melhor Destino Insular". Ao todo, Portugal alcançou seis prémios de enorme prestígio internacional no setor do turismo (Observador, 2017).

Com o dinamismo causado pelo aumento do turismo em Portugal, aumenta também a oferta de atividades. Todas as oportunidades de negócio são exploradas, aumentando a oferta e diversidade de eventos, tal como de plataformas e meios de divulgação dos mesmos.

“A revolução tecnológica dos últimos anos fez evoluir todos os processos de gestão e transação da informação e, conseqüentemente, alterou a relação entre os vários intervenientes na atividade turística. Alguns dos protagonistas do negócio hoje são exatamente empresas com maior capacidade tecnológica para gerir esta informação no sentido de obter vantagem no negócio: chegar melhor ao mercado, fazer a melhor oferta, vender em cada momento ao melhor preço, influenciar a decisão de compra, ouvir e interpretar a voz do cliente” (Publituris, 2014).

A tecnologia de hoje em dia permite gerir muito mais facilmente as atividades turísticas e promover essas mesmas atividades junto de todos destinatários, ou seja, os turistas e os utilizadores nacionais nas suas atividades de lazer.

Uma das formas mais rápidas e eficazes de fazer chegar a informação dos eventos ao seu público alvo é através de aplicações, que poderão disponibilizar informação de atividades nas diversas categorias e locais e para diferentes faixas etárias.

## 1.1. Contextualização e Motivação

Nos tempos modernos, quando temos acesso a variadas aplicações tecnológicas para os mais diversos fins, torna-se importante perceber o que faz com que determinada aplicação se destaque em comparação às suas semelhantes, ou seja, qual a mais valia de usar determinada aplicação em detrimento de outra criada para o mesmo fim.

Devido ao estilo de vida acelerado da maioria da população nas grandes cidades e à vasta oferta de aplicações com funcionalidades semelhantes, a principal questão que se coloca aquando da escolha da aplicação a usar é “o que vou beneficiar ao usar esta aplicação?”. Se tivermos em consideração as grandes aplicações para dispositivos móveis existentes no mercado (as mais usadas), nas mais diversas áreas, chegamos à conclusão que, no que diz respeito a aplicações, os utilizadores querem algo simples e direto – algo que faça exatamente o pretendido, de forma simples e sem perder tempo pelo caminho.

Com estes dois princípios em mente, pretende-se desenvolver uma aplicação para dispositivos móveis que dê informação sobre os eventos mais apropriados ao utilizador, que lhe permita gerir o seu tempo e planear os seus eventos e tempos livres.

## 1.2. Objetivo

A aplicação a conceber destina-se a qualquer pessoa interessada em disfrutar de experiências na cidade onde se encontra, quer se trate de cidadãos locais ou turistas. Esta aplicação tem por objetivo permitir o acesso a informação dos eventos a decorrer na cidade, permitir gestão desses mesmos eventos, em função do tempo livre do utilizador, e possibilitar a interação com outras pessoas que demonstrem interesse no mesmo evento. A aplicação deverá ser intuitiva para o utilizador e constituir-se como uma ferramenta capaz de agilizar e rentabilizar todo o processo de gestão/escolha de experiências/eventos na cidade. As principais ações a disponibilizar ao utilizador são:

- Ver eventos relevantes com base nos interesses de cada um;
- Procurar pessoas que vão aos mesmos eventos;
- Organizar viagens;
- Planear automaticamente com atividades o tempo pretendido.

## 1.3. Cronograma

Tendo por base os objetivos deste projeto e de forma a poder concretizar cada um deles com sucesso, foi realizado um plano de trabalho, com duração total de um ano, e dividido em 7 fases, que se apresentam de seguida:

### **Fase 1 – Estudo do estado de arte (2 meses):**

É nesta fase que será efetuado o estudo do estado de arte do presente tema, de modo a que seja possível identificar sistemas existentes semelhantes ao sistema que se pretende implementar.

### **Fase 2 – Modelação e proposta de solução (1 mês):**

Nesta fase será feita uma análise dos requisitos funcionais do sistema. Com base nessa primeira análise, serão definidos os casos de uso e modelo de base de dados para sustentar esses mesmos casos de uso. A modelação do sistema servirá para definir a arquitetura do sistema e apresentar uma proposta de solução com o objetivo de resolver os requisitos pedidos para sistema.

### **Fase 3 – Desenvolvimento da aplicação (3 meses):**

Nesta fase será colocada em prática a implementação da aplicação, será construída a base de dados, será desenvolvido o código da componente móvel e do servidor, que permitirá colocar o sistema a funcionar.

### **Fase 4 – Integração dos diversos componentes do sistema (1 mês):**

Será realizada a integração das diversas camadas do sistema, de forma a ter uma aplicação a funcionar.

### **Fase 5 – Testes ao sistema (1 mês):**

Nesta fase irá ser realizado uma análise dos desenvolvimentos e testes com base nos requisitos inicialmente definidos, de modo a corrigir erros que possam existir ou funcionalidades não corretamente implementadas.

**Fase 6 – Análise e discussão de resultados (1 mês):**

Esta é a fase de análise dos resultados dos testes realizados e validação da solução implementada.

**Fase 7 – Conclusão e escrita da tese (3 meses):**

A escrita da dissertação será realizada ao longo de todas as fases. Nesta fase será concluída a escrita da dissertação e correção da mesma.

O cronograma do plano de trabalho está representado na **Tabela 1**.

Tabela 1 - Cronograma do plano de trabalho

	Jan 18	Fev 18	Mar 18	Abr 18	Mai 18	Jun 18	Jul 18	Ago 18	Set 18	Out 18	Nov 18	Dez 18
<b>Fase 1</b>												
<b>Fase 2</b>												
<b>Fase 3</b>												
<b>Fase 4</b>												
<b>Fase 5</b>												
<b>Fase 6</b>												
<b>Fase 7</b>												

#### 1.4. Estrutura do Relatório

A estrutura desta dissertação compreende sete capítulos.

O **primeiro capítulo** contém uma pequena introdução ao documento, com o objetivo de enquadrar este trabalho, contextualizando o tema e os objetivos propostos para o sistema a desenvolver.

O **segundo capítulo** apresenta uma breve análise e descrição de aplicações que, de alguma forma, se relacionam com aquela que se pretende desenvolver, sendo identificadas diversas funções que elas dispõem. No final deste capítulo é efetuada uma análise comparativa das diversas aplicações e são identificados os requisitos funcionais base que a aplicação a desenvolver deve satisfazer.

No **terceiro capítulo** é apresentada a metodologia de desenvolvimento adotada para este projeto, o porquê de ter sido escolhida esta metodologia e são detalhados os conceitos subjacentes à mesma.

No **quarto capítulo** são identificados os requisitos do sistema a desenvolver e é efetuada a modelação do sistema. Para tal, são criados diagramas de casos de uso, diagramas de sequência e diagramas de classes; é também apresentado o modelo de dados necessário para o sistema.

No **quinto capítulo** será apresentada a arquitetura do sistema e as tecnologias usadas.

No **sexto capítulo** é apresentado o protótipo final (principais funcionalidades e ecrãs finais) e o resultado da utilização da aplicação num ambiente controlado (um dispositivo

móvel e um servidor) – são descritos alguns casos de utilização e analisado se o resultado foi o pretendido.

Por último, no **sétimo capítulo**, são apresentadas as conclusões sobre o trabalho desenvolvido e são identificados pontos que poderão ser explorados em desenvolvimentos futuros.

## 2. Estado da Arte

Neste capítulo são descritas algumas das aplicações com características semelhantes à aplicação a desenvolver no âmbito desta dissertação; aplicações essas que foram desenvolvidas com o intuito de ajudar os utilizadores no planeamento das visitas, percursos ou eventos que pretendem conhecer. Na parte final do capítulo é apresentado um resumo de algumas das aplicações, bem como breve análise comparativa entre elas.

### 2.1. Aplicações em Estudo

Nesta secção são apresentadas **cinco** aplicações móveis que se destinam a apoiar os turistas nas suas viagens e atividades de lazer, ou um cidadão local que necessita de apoio no momento de decidir como pode aproveitar o tempo livre. Inicialmente será apresentada uma breve descrição para cada uma e depois serão descritas as várias funcionalidades das mesmas, considerando **dez** vertentes distintas:

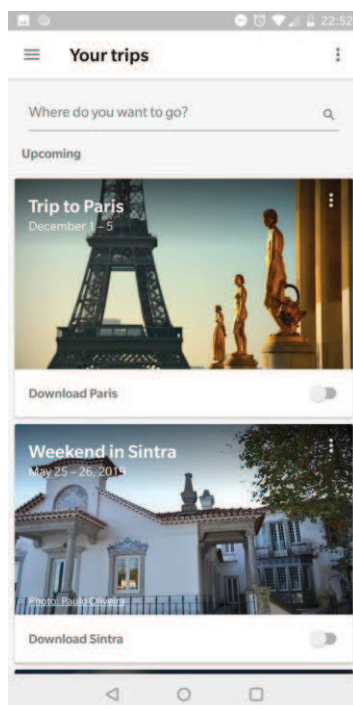
1. **Recomendação de pontos de interesse** - Sugere ao utilizador pontos de interesse com base na sua localização, nos seus interesses e na hora do dia em que está a ser feita a pesquisa (por exemplo, não sugerir um local que poderá estar fechado nesse momento), para ser mais preciso nas sugestões. Permite aplicar filtros e pesquisas sobre essas recomendações.
2. **Conteúdo sobre o local** - Qual o conteúdo e nível de detalhe apresentado para cada local/evento. Qualquer utilizador poderá adicionar informação sobre o local/evento enriquecendo o conteúdo do mesmo. Se tem conteúdo interativo: vídeos, adicionar comentários, avaliar, partilhar, etc..
3. **Planear percursos automaticamente** - Planear automaticamente o percurso com base nos locais sugeridos. Tem em conta as sugestão dos eventos e locais com base no tempo disponível e configurável (horas, dias, etc.). O planeamento automático de percursos é efetuado tendo em conta os interesses do utilizador. Conteúdo totalmente gratuito ou tem algum tipo de licenças ou custos.
4. **Interação entre utilizadores** - A aplicação permite a troca de mensagens entre os vários utilizadores; permite seguir os interesses de outros utilizadores e receber sugestões baseadas nos utilizadores seguidos.
5. **Planear viagens** - Permite agrupar sugestões de um determinado local, planear percursos e guardar informação de uma viagem.
6. **Interação com mapa** - Permite utilizar mapa e deslocar com o auxílio do mesmo.
7. **Conteúdos offline** - permissão para navegação e visualização de conteúdos *offline*.
8. **Publicar eventos** - Quem tem permissão para criar eventos: qualquer utilizador, apenas utilizadores com uma competência específica ou só o administrador. Necessidade de aprovação após criação de evento.
9. **Suporte em vários idiomas** - disponibilização da aplicação em vários idiomas.
10. **Sistemas operativos** para o qual a aplicação foi desenvolvida - quais as plataformas nas quais se pode instalar a aplicação( *iOS, Android, etc.*).

### 2.1.1. Google Trips - Travel

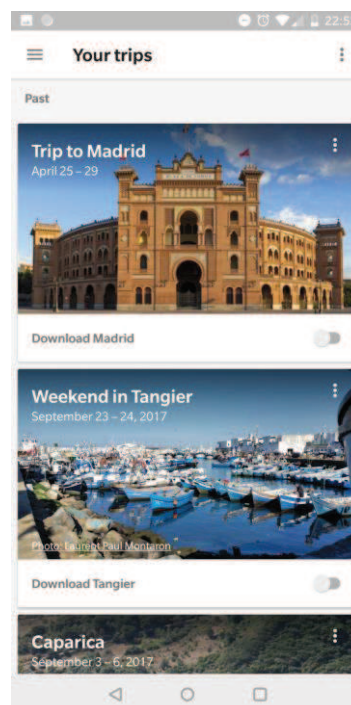
*Google Trips* (Google, 2017) é uma aplicação lançada recentemente com o objetivo de criar automaticamente percursos entre os pontos de interesse de determinada cidade. Ao entrar na aplicação, e depois de efetuado o *login*, esta inicia com o ecrã “Viagens”.

As viagens (**Figura 1** e **Figura 2**), podem ser adicionadas manualmente ou automaticamente:

- **Manual:** adicionando um novo item “viagem” na aplicação, escolhendo o local e os dias da viagem.
- **Automática:** caso a aplicação esteja associada à mesma conta *Google* em que foram realizadas as reservas dos voos, hotéis ou carros. Todas as viagens, anteriores ou futuras, são automaticamente carregadas na aplicação.



**Figura 1** - Lista de viagens futuras



**Figura 2** - Lista de viagens passadas

Ao selecionar uma viagem da lista anterior, ficará disponível informação sobre reservas efetuadas, percursos, informação útil e pontos de interesse (comida, bebida, monumentos ou vida noturna), como mostra a **Figura 3**. A opção “*Things to do*” contém os vários pontos de interesse organizados em diferentes categorias, dependendo estas da cidade em questão; *Google Trips* ajusta-se e poderá apresentar categorias diferentes de forma a adaptar-se ao que de melhor a respetiva cidade tem para visitar, como mostra a **Figura 4**. Não existe pesquisa de eventos nem é possível aplicar filtros nas listas.

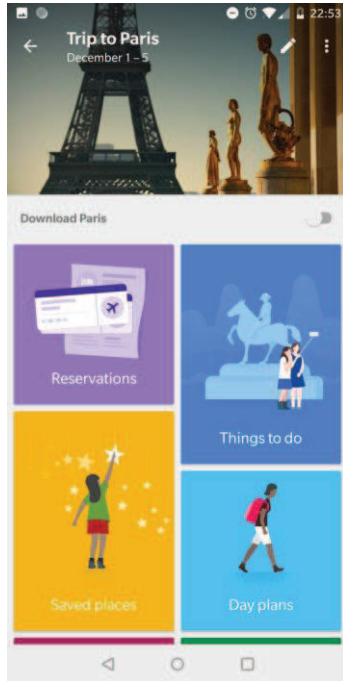


Figura 3 - Detalhe da viagem

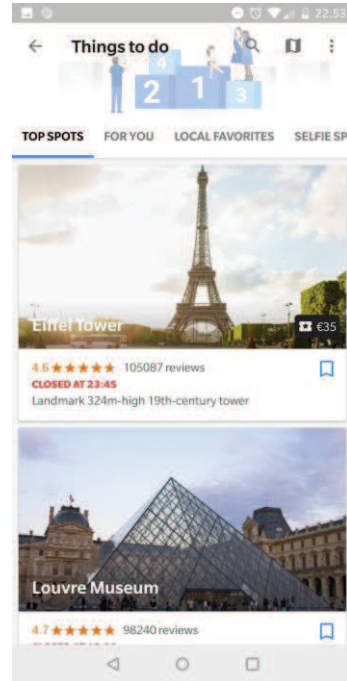


Figura 4 - Listas de categorias

Ao entrar no detalhe de determinado ponto de interesse, será possível consultar informação geral sobre o mesmo, breve descrição, localização, horário, fotos, comentários e avaliações dos diversos utilizadores -**Figura 5, Figura 6, e Figura 7**. No detalhe não é visível informação sobre o acesso para pessoas com necessidades especiais de locomoção. O conteúdo é totalmente gratuito e todo o conteúdo apresentado é o disponibilizado também no motor de pesquisa *Google*. Os utilizadores não podem adicionar na aplicação fotos da sua experiência.

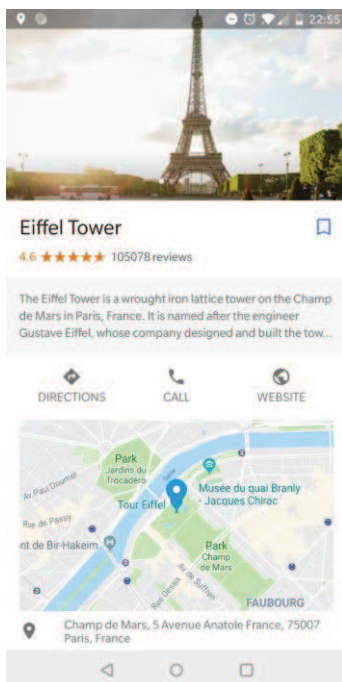


Figura 5 - Detalhe do evento

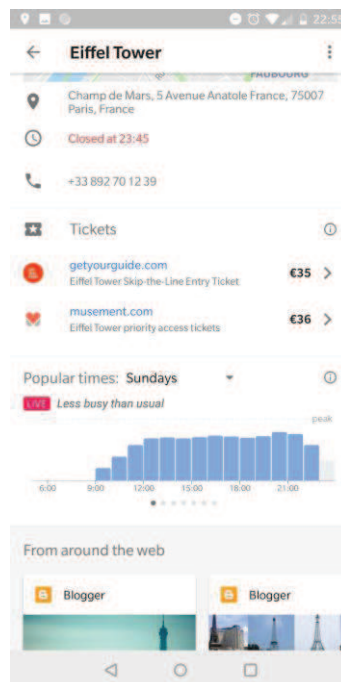


Figura 6 - Detalhe do evento

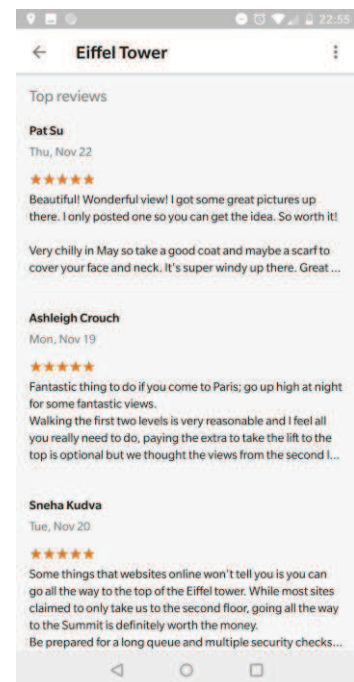
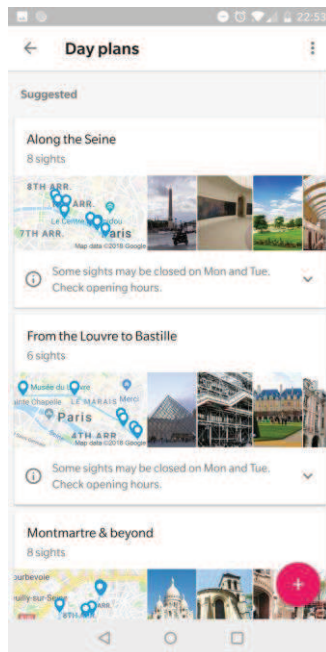
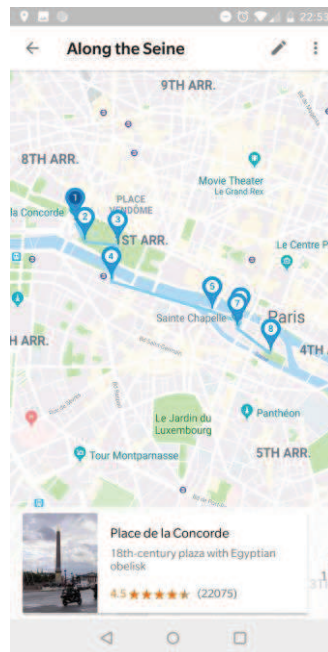


Figura 7 - Comentários ao evento

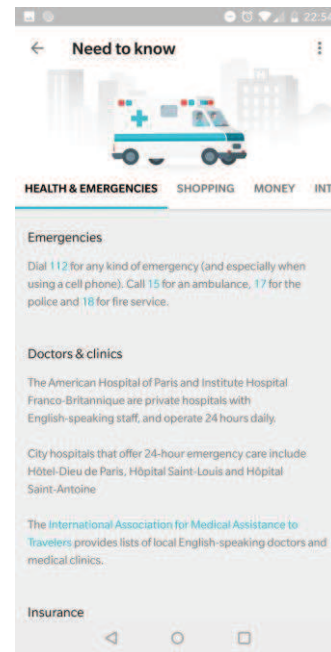
São sugeridos planos diários automaticamente, compostos por diferentes percursos, com base nos pontos de interesse, tempo disponível e localização - **Figura 8**. É possível escolher o ponto por onde se deseja começar o percurso, retirar ou bloquear pontos que já tenham sido visitados. O detalhe do percurso mostra num mapa todos os pontos que dele fazem parte, como é possível ver na **Figura 9**.



**Figura 8** - Percurso gerado automaticamente



**Figura 9** - Detalhe de um percurso



**Figura 10** - Informações úteis

A aplicação tem alguma informação relevante e relativa à cidade e país que estamos a visitar, como se pode ver na **Figura 10**.

Não é possível trocar mensagens ou seguir os interesses de outros utilizadores, não sendo identificada qualquer tipo de interação entre utilizadores. A interação com mapas existe no detalhe de cada ponto e, nos percursos automáticos, é possível depois navegar entre pontos utilizando o *google maps*.

Esta aplicação tem modo de funcionamento *offline*, em que o utilizador pode previamente fazer *download* de toda a informação da viagem. Apenas está disponível para o sistema operativo *Android*.

### 2.1.2. GuiaDeRodas

O *guiaderodas* (*guiaderodas*, 2017) é um guia colaborativo (entre várias entidades) destinado à avaliação e consulta de locais acessíveis para pessoas com dificuldade de locomoção.

Esta aplicação propõe a todos os utilizadores que, com pouco tempo, possam dar informação da acessibilidade nos vários pontos de interesse. Esta aplicação pretende estimular a importância deste tema junto da população, para que avaliem os diversos

estabelecimentos, permitindo assim uma vida mais autônoma e uma maior inclusão para as pessoas com dificuldade de locomoção.

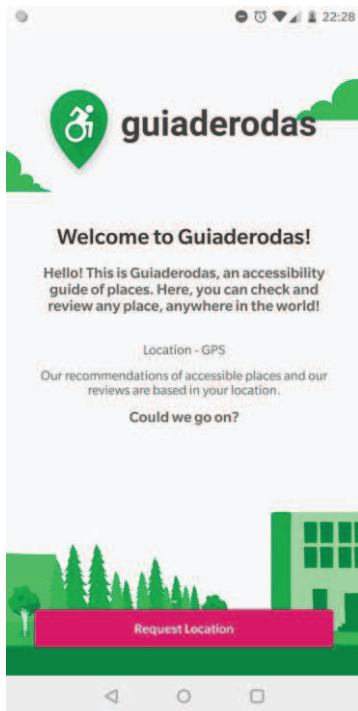


Figura 11 - Loading da aplicação

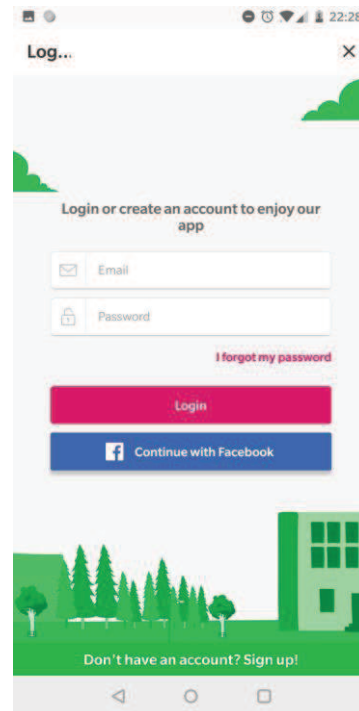


Figura 12 - Ecrã de login/registo

A **Figura 11** apresenta o *loading* da aplicação, em que é apresentada uma descrição e objetivo da mesma. De seguida, é apresentado o ecrã do *login*/registo, onde é possível utilizar conta ou Facebook - **Figura 12**.

“A metodologia de avaliação e análise do guiaderodas alia o conhecimento técnico de arquitetos especializados à vivência prática de pessoas com dificuldade de locomoção. Essa combinação entre teoria e prática possibilita o desenvolvimento de soluções inteligentes e eficazes de acessibilidade” (guiaderodas, 2017).

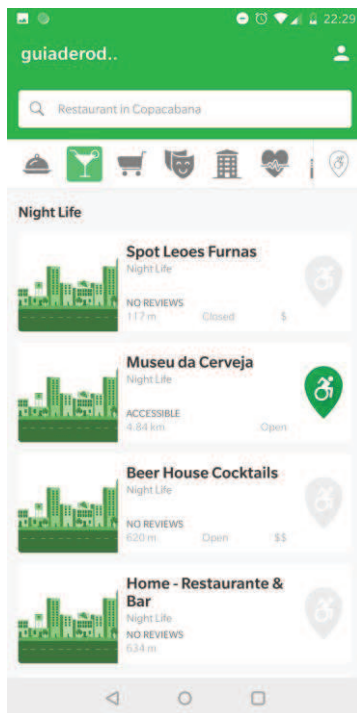
A comunidade responsável pela aplicação analisa cada espaço, propondo mudanças ao nível da acessibilidade para pessoas com mobilidade reduzida. Após análise e gerado o relatório, este é entregue no respetivo local com as sugestões de alterações. Caso as situações detetadas sejam corrigidas, é entregue um sinal de reconhecimento pelo trabalho realizado, e o mesmo pode ser exposto no local.

No âmbito da avaliação estão presentes os mais diversos tipos de edifícios, comércio, escolas, hospitais, residências, hotéis e etc. O guiaDeRodas sugere locais divididos em 6 categorias:

- Restaurantes;
- Vida Noturna;
- Compras;
- Diversão;
- Hospedagem;

- Saúde.

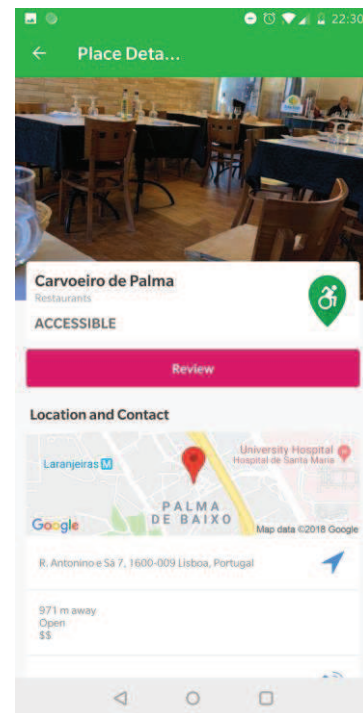
A aplicação permite mostrar os locais referidos ordenados por relevância, distância ou avaliação. Dentro de cada uma das categorias são listados os pontos de interesse, podendo ainda aplicar um filtro de se o local é acessível ou não a pessoas com dificuldade de locomoção. Para estas listas não é tido em conta qualquer tipo de interesse dos utilizadores. A **Figura 13** apresenta o anteriormente descrito.



**Figura 13** - Lista de pontos de interesse



**Figura 14** - Resumo de um ponto de interesse



**Figura 15** - Localização de um ponto de interesse com acessibilidade

Ao seleccionar um ponto de interesse, é apresentada informação sobre acessibilidade do mesmo - **Figura 14**. Caso o utilizador pretenda consultar o detalhe do ponto de interesse em questão, deverá seleccionar a opção 'Details' e ser-lhe-á apresentada informação principal da localização, uma fotografia do local, detalhes e comentários - **Figura 15** e **Figura 16**. Os locais não têm classificação por parte dos diferentes utilizadores mas será possível obter *feedback* sobre os mesmo através dos comentários - **Figura 17**. Não foi identificado qualquer conteúdo pago nesta aplicação.

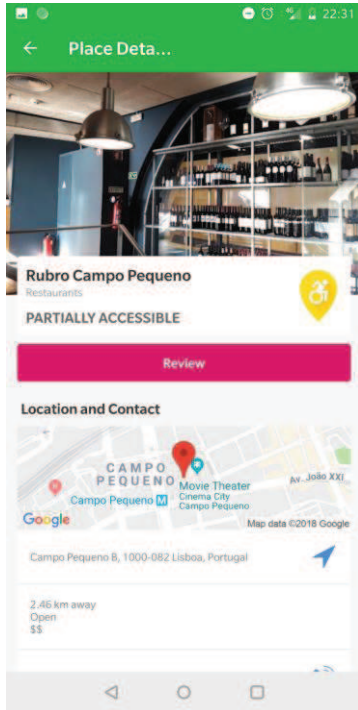


Figura 16 - Localização de um ponto de interesse com acessibilidade parcial

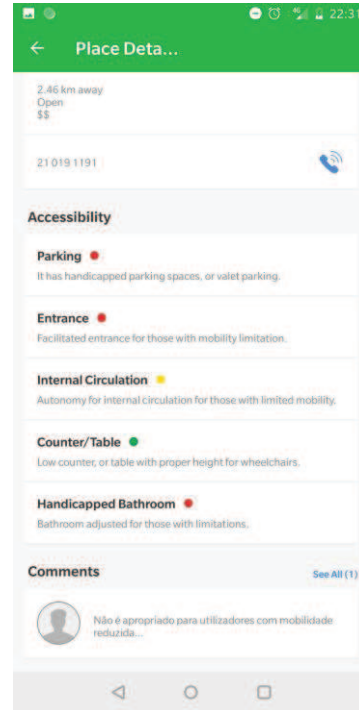


Figura 17 - Detalhe do ponto de interesse

Não é possível criar percursos e fazer planos diários com base nos pontos de interesse; apenas é possível consultá-los individualmente. Não é possível guardar os eventos ou informação relevante para a realização de uma viagem.

A interação entre utilizadores é inexistente, ou seja, não existe possibilidade de troca de mensagens ou de seguir outro utilizador. A aplicação não funciona em modo *offline*, podendo apenas ser visualizado os detalhes de um ponto de interesse se os dados do mesmo tiverem sido carregados previamente na aplicação.

Os eventos são adicionados pelos utilizadores, normalmente pelos donos dos espaços ou organizadores do evento, necessitando também de permissões para o efeito por parte dos administradores. A aplicação está disponível para *Android* e *iOS*.

### 2.1.3. JiTT.travel

JiTT.Travel (*JiTT.travel*, 2017) destaca-se pela disponibilização de vários guias para as principais cidades. Os guias são criados por diversas entidades (desde organizações turísticas, ao comum cidadão) e são posteriormente revistos pela equipa responsável. Após a revisão, os guias são disponibilizados na aplicação, sendo que existem vários guias por cada destino, como é possível observar na **Figura 20**.

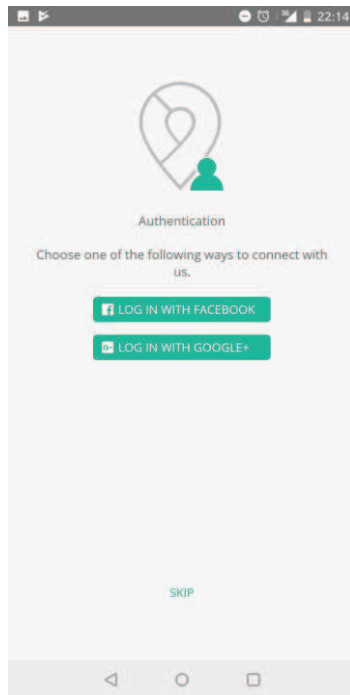


Figura 18 - Ecrã de login ou registo



Figura 19 - Idiomas de preferência

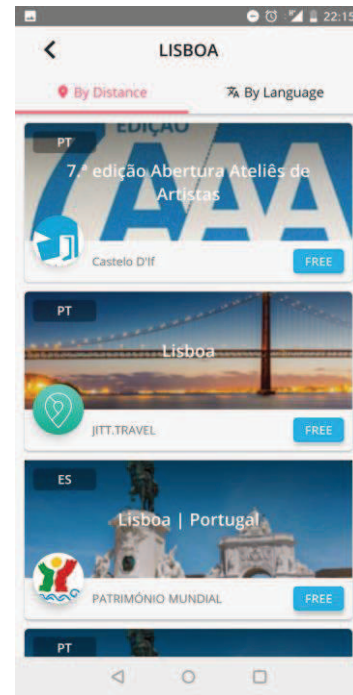


Figura 20 - Lista de guias de Lisboa

A aplicação permite que o *login* seja efetuado apenas utilizando o *Facebook* ou o *Google* (Figura 18). Na primeira utilização é necessário escolher 3 preferências linguísticas, sendo esses os idiomas para apresentação dos guias - Figura 19. Depois de escolhidos os idiomas preferenciais, o utilizador deverá escolher a cidade pretendida e será reencaminhado para o ecrã mostrado na Figura 20 (para o caso de Lisboa), onde poderá ver a lista de guias da cidade e o idioma de cada um.

Esta aplicação dispõe de (JiT.T.travel, 2017):

- Centenas de guias de destinos;
- Vários idiomas;
- Mapas *offline*;
- Pontos de Interesse indicados no mapa;
- Informações úteis, como pontos *WiFi*, postos de turismo, informações sobre transportes públicos e caixas multibanco, etc;
- Áudio integral, para uma melhor experiência de visita guiada em alguns locais;
- Informação e curiosidades sobre o local;
- Um planeador de rotas, que cria passeios com base no tempo disponível do utilizador.

A recomendação dos guias é realizada tendo em consideração a localização que o utilizador escolher. Estas recomendações não consideram os horários dos vários locais, tendo o utilizador de verificar cada um dos pontos. É possível aplicar filtros no detalhe de um guia, como por exemplo, filtrar a pesquisa a locais cobertos/interiores.

O conteúdo disponibilizado pode ou não ser pago, dependendo do guia. Esta distinção é feita ao nível do guia e não dos pontos de interesse, ou seja, se um guia for gratuito todos os pontos no seu interior também o são.

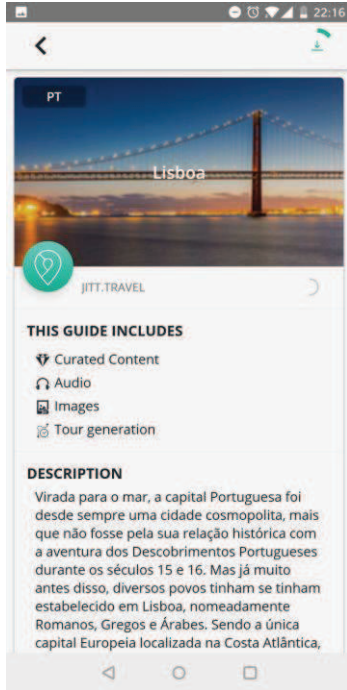


Figura 21 - Detalhe de um guia

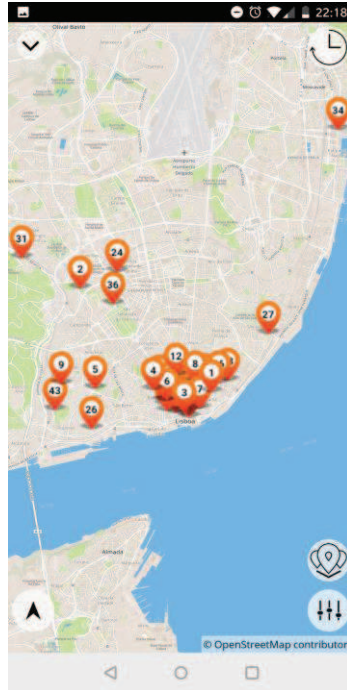


Figura 22 - Pontos do guia



Figura 23 - Opções do guia

Cada guia pode basear-se num único tema ou abranger um leque variado de interesses e locais. No detalhe de um guia é apresentado o tipo de conteúdo e uma breve descrição do mesmo, como apresenta a **Figura 21**. Ao aceder ao guia é apresentado o ecrã com os pontos de interesse - **Figura 22**, onde é possível visualizar a localização geral de todo o guia. Os pontos de interesse são apresentados num mapa, mas não é permitida a navegação através deste. Neste ecrã é possível, também, aceder às opções do guia, entre elas, o número de horas disponíveis - **Figura 23**. Ainda neste ecrã, na parte inferior, é possível visualizar um pequeno detalhe de cada ponto, como apresentado na **Figura 24**.

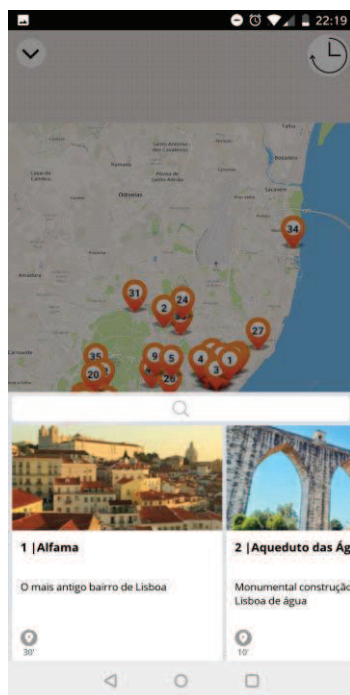


Figura 24 - Resumo do ponto de interesse do guia



Figura 25 - Detalhe do ponto de interesse do guia

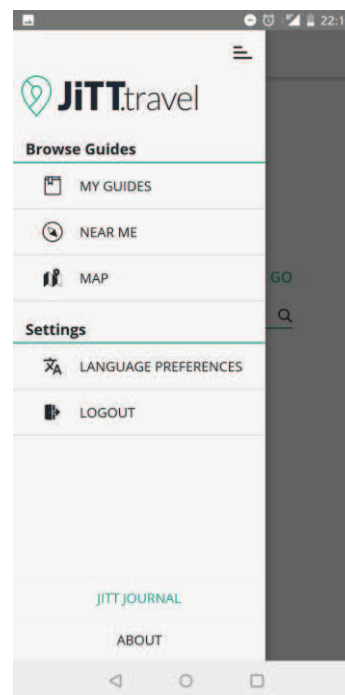


Figura 26 - Menu da JITT.travel

Para cada um destes pontos é disponibilizado conteúdo escrito ou interativo, que permite dar a conhecer as melhores histórias e informação sobre cada um, tal como apresentado na **Figura 25**. Não foi identificada nenhuma informação sobre condições de acesso a pessoas com mobilidade reduzida. Contudo, como a informação de cada guia depende da entidade que o criou, poderá em alguns casos ser possível adicionar informação sobre acessibilidade.

Os utilizadores não podem comentar, avaliar, adicionar imagens ou qualquer outro tipo de informação ao evento. A interação é unilateral, não podendo o utilizador expressar a sua opinião na plataforma. Não foi encontrado qualquer tipo de interação entre utilizadores, sendo que não é permitido seguir, comunicar ou qualquer outro tipo de interação.

Não existe qualquer tipo de conceito de viagem, ou seja, não é permitido criar uma viagem que se irá realizar no futuro ou guardar as viagens do passado.

Em cada guia, é permitido adicionar pontos que temos especial interesse em visitar ou eliminar pontos do percurso nos quais já estivemos ou não temos interesse em ver, personalizando assim todo o passeio.

A aplicação permite o *download* de cada guia, podendo assim trabalhar *offline*, economizando bateria e dados móveis. Os guias que o utilizador fez *download* podem ser acedidos através do menu principal, representado na **Figura 26**. Neste menu também é possível alterar as preferências linguísticas, como referido anteriormente.

Qualquer tipo de utilizador ou entidade poderá criar pontos de interesse e, posteriormente, os guias, necessitando apenas de permissões por parte dos administradores. A aplicação está disponível para *Android* e *iOS*.

### 2.1.4. Minube

Minube (Minube, 2017) é uma aplicação que permite criar roteiros/listas com pontos de interesse inseridos em várias categorias. As listas são constituídas pelos pontos de maior interesse num determinado local.

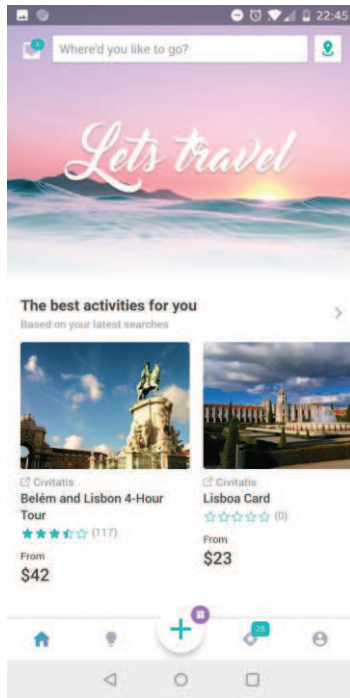


Figura 27 - Ecrã principal da aplicação Minube

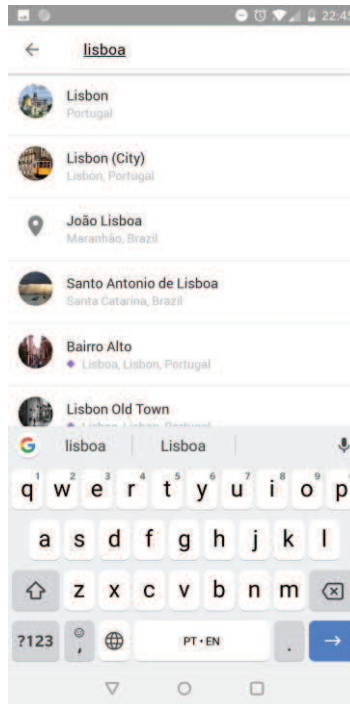


Figura 28 - Selecionar local

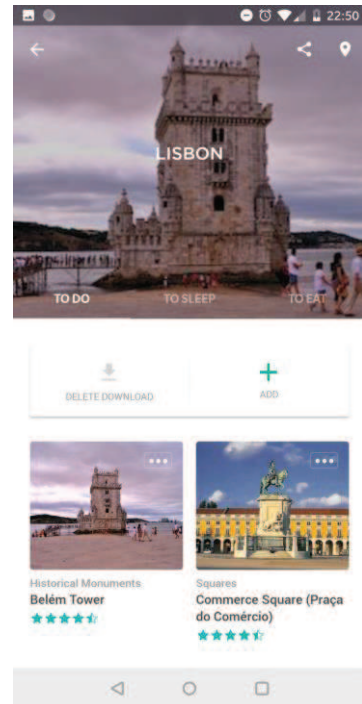


Figura 29 - Pontos de interesse para Lisboa

Ao iniciar a aplicação, esta tem acesso à localização do utilizador e disponibiliza sugestões exclusivamente com base nessa mesma localização - **Figura 27**. Caso o utilizador pretenda consultar pontos de interesse num outro local, poderá seleccionar uma outra localização - **Figura 28**.

Para cada cidade é disponibilizada uma lista de pontos de interesse, que depois é dividida em 3 categorias: ver, comer e dormir - **Figura 29**. As sugestões recebidas são as mesmas para todos os utilizadores, não havendo distinção entre as preferências de cada um.

O conteúdo apresentado para cada um dos pontos de interesse consiste apenas no título acompanhado com várias imagens do local - **Figura 30**. Não é apresentada qualquer informação sobre acessibilidade para pessoas com dificuldades de locomoção.

Os utilizadores podem comentar e avaliar os pontos de interesse, partilhando com os restantes utilizadores a sua experiência - **Figura 31** e **Figura 32**. O utilizador pode, ainda, criar a sua própria lista com base em listas já existentes, adicionando ou removendo pontos de interesse.

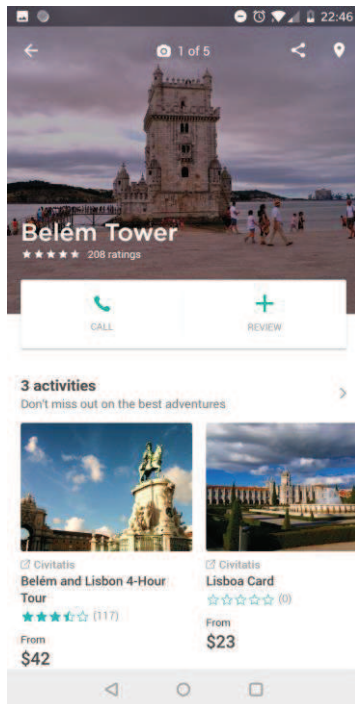


Figura 30 - Ponto de interesse

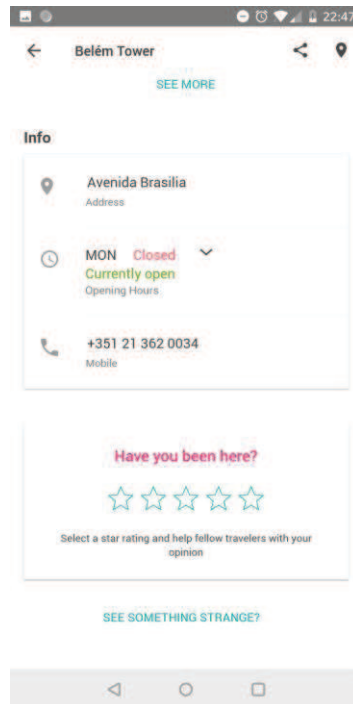


Figura 31 - Detalhe de um ponto de interesse

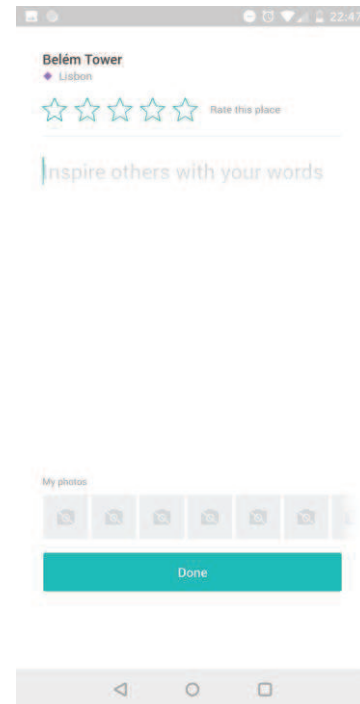


Figura 32 - Comentário e avaliação do local

A aplicação permite criar álbuns de forma automática, utilizando a informação disponível em cada fotografia do *smartphone*. Os utilizadores podem acrescentar ainda comentários e avaliações aos mesmos, podendo no final partilhar o respetivo álbum.

Esta aplicação tem ainda um modo muito interessante, com o nome “Inspirador”, que permite escolher entre distância, duração, cor, tipo (romântico, relaxar, cultura, aventura, com amigos, entre muitos outros) e continente. Através da seleção de uma das categorias referidas, são sugeridas viagens, com a lista já acima referida dividida em 3 (comer, ver e dormir).

A aplicação não tem a função de recomendar atividades baseando-se no tempo que o utilizador tem disponível, nem é possível criar percursos de forma manual ou automática. É possível guardar as listas em modo *offline* (formato .pdf).

A interação entre utilizadores é realizada através da partilha das viagens guardadas na aplicação, não sendo possível trocar mensagens entre os utilizadores, nem seguir os interesses dos outros.

A utilização do mapa serve apenas para visualizar a localização dos pontos de interesse. Os sistemas operativos para os quais esta aplicação se encontra disponível são *Android* e *iOS*.

### 2.1.5. TripCase

*TripCase* (tripcase, 2017) é uma aplicação que permite planejar viagens e organizar o dia-a-dia do utilizador. Esta é uma aplicação que foca a organização e gestão da viagem, sendo o utilizador quem adiciona manualmente cada item do seu interesse.

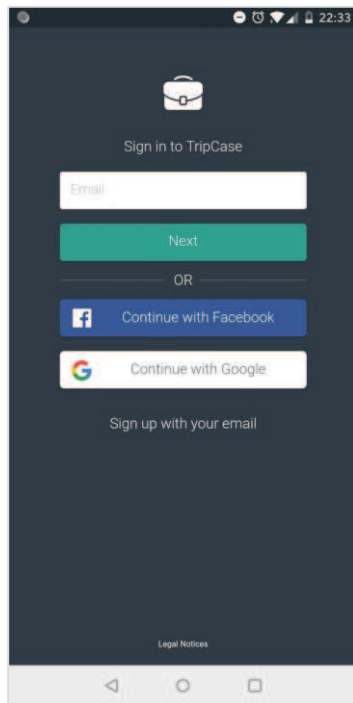


Figura 33 - Página principal da aplicação Trip Case

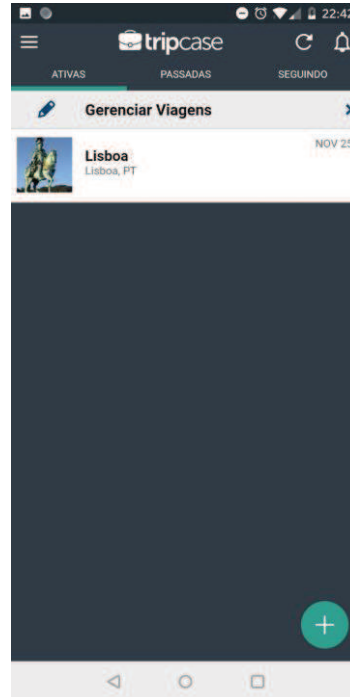


Figura 34 - Lista de viagens



Figura 35 - Lista de itens para uma viagem

A aplicação permite o *login* de três formas: *email*, *Facebook* ou conta *Google* - **Figura 33**. Ao entrar na aplicação, é possível criar um item “viagem” - **Figura 34**, ficando imediatamente disponível várias funcionalidades, tais como gerir os voos, reservar hotel, carro, etc. - **Figura 35**.

A aplicação não faz recomendações de qualquer ponto de interesse. Para que o ponto fique incluído na viagem, é necessário que o utilizador tenha reserva no local pretendido ou que efetue pesquisa pelo mesmo e o adicione à viagem - **Figura 36**.

O conteúdo sobre cada ponto contém apenas uma breve descrição sobre o local (se o utilizador tiver adicionado). No detalhe não contém qualquer tipo de informação sobre acessos a pessoas com necessidades especiais. O conteúdo é totalmente gratuito, não tendo sido identificadas qualquer tipo de licenças ou outro tipo de custos. O detalhe de um ponto é visível na **Figura 38**.

Depois de adicionados os pontos de interesse, é permitido obter direções e solicitar um táxi/uber para chegar ao destino. É possível receber mensagens com curiosidades e informações úteis do destino.

A viagem criada pode ser partilhada com outros utilizadores da aplicação, mantendo assim todas as pessoas que vão viajar atualizados dos planos de viagem, podendo ser esses planos visto cronologicamente pela ordem que estão a ser planeados - **Figura 37**.

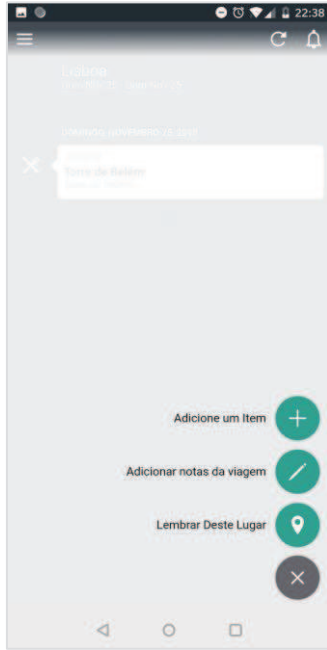


Figura 36 - Adicionar item à viagem



Figura 37 - Timeline da viagem

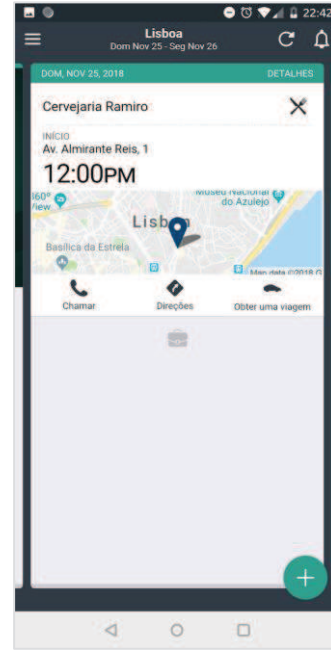


Figura 38 - Detalhe de um item da viagem

A principal funcionalidade desta aplicação, como já referido, é o planeamento e organização de viagens. Para que tal seja ainda mais fácil, a aplicação oferece integração com vários parceiros, de forma a importar automaticamente as reservas para a aplicação (tripcase, 2017).

A aplicação não contém mapas, mas é possível lançar outra aplicação de navegação instalado no dispositivo móvel. A aplicação permite guardar todo o conteúdo *offline* para uso posterior.

Existe a disponibilização de mapas de transportes públicos, como autocarros e metros, mas na aplicação o utilizador não tem acesso a um mapa interativo. A aplicação tem suporte em vários idiomas, sendo que muitas vezes o idioma usado é o do local do ponto de interesse.

A aplicação encontra-se disponível para as plataformas *Android* e *iOS*.

## 2.2. Análise Comparativa

A **Tabela 2** apresenta de forma resumida as características de cada uma das aplicações analisadas.

**Tabela 2** - Análise comparativa das aplicações

Vertentes	Google Trip	Guia de rodas	Jitt.Travel	Minube	TripCase
<b>Recomendação de eventos</b>	<b>Sim</b>	<b>Sim</b>	<b>Sim</b>	<b>Sim</b>	<b>Não</b>
Sugestões automáticas	Sim	Sim	Sim	Sim	Não
Baseadas em interesses	Sim	Não	Não	Sim	Não
Filtros e pesquisa	Não	Sim	Sim	Sim	Sim
<b>Conteúdo sobre o local</b>	<b>Sim/Não</b>	<b>Sim/Não</b>	<b>Sim/Não</b>	<b>Não</b>	<b>Não</b>
Conteúdo detalhado	Sim	Sim	Sim	Sim	Sim
Informação para pessoas com necessidades especiais	Não	Sim	Não	Não	Não
Utilizadores podem adicionar informação	Sim	Não	Não	Não	Não
Conteúdo interativo	Não	Não	Sim	Sim	Não
Conteúdo totalmente gratuito	Sim	Sim	Não	Sim	Sim
<b>Planear percursos</b>	<b>Sim</b>	<b>Não</b>	<b>Sim</b>	<b>Não</b>	<b>Não</b>
Automático	Sim	Não	Sim	Não	Não
Tem em conta tempo disponível e horários	Sim	Não	Sim	Não	Não
Tem por base o interesse do utilizador	Não	Não	Não	Não	Não
<b>Interação entre utilizadores</b>	<b>Não</b>	<b>Não</b>	<b>Não</b>	<b>Não</b>	<b>Não</b>
Troca de mensagens	Não	Não	Não	Não	Não
Funcionalidade seguir utilizadores	Não	Não	Não	Não	Não
<b>Planear viagem</b>	<b>Sim</b>	<b>Não</b>	<b>Não</b>	<b>Sim</b>	<b>Sim</b>
<b>Interação com mapas</b>	<b>Sim</b>	<b>Não</b>	<b>Sim</b>	<b>Sim</b>	<b>Não</b>
<b>Conteúdo offline</b>	<b>Sim</b>	<b>Não</b>	<b>Sim</b>	<b>.pdf</b>	<b>Não</b>
<b>Publicar eventos</b>	<b>Não</b>	<b>Sim</b>	<b>Sim</b>	<b>Não</b>	<b>Não</b>
<b>Suporte para vários idiomas</b>	<b>Sim</b>	<b>Sim</b>	<b>Sim</b>	<b>Sim</b>	<b>Sim</b>
<b>Sistemas operativos</b>	<b>Android/iOS</b>	<b>Android/iOS</b>	<b>Android/iOS</b>	<b>Android/iOS</b>	<b>Android/iOS</b>

O princípio base da maioria das aplicações analisadas é a **sugestão de eventos** e locais para visitar, aos seus utilizadores. Das aplicações anteriormente analisadas, apenas a *TripCase* não apresenta estas sugestões, pois o principal objetivo desta é a gestão/planeamento da viagem. As sugestões podem depois ser filtradas, ou seja, permite ao utilizador escolher o que mais lhe interessa naquele momento, limitando assim os locais sugeridos. Das aplicações analisadas, apenas o *Google Trips* não utiliza filtros e pesquisa.

No âmbito de sugestão de locais e eventos ao utilizador, é importante que a aplicação tenha em conta os horários de funcionamento desses mesmos locais. Também é importante que a aplicação cruze a informação sobre o tempo estimado de visita para cada ponto e a disponibilidade do utilizador. Das aplicações analisadas, apenas duas têm em conta os horários dos locais (*Jitt.Travel* e *Google Trips*) e nenhuma delas tem informação sobre o tempo mínimo estimado para visitar o local em questão.

No caso das aplicações analisadas, apenas uma (*Jitt.Travel*) apresenta **conteúdo pago**, que passa pela disponibilização de guias particulares, entre outros conteúdos *premium*.

A análise sobre a componente do **planeamento de viagens** diz respeito à possibilidade de o utilizador conseguir gerir todas as vertentes de uma viagem (voo, estadia, transporte, ocupação dos seus dias, etc.), na sua aplicação. Para isso, o planeamento de percursos automáticos é uma característica importante neste tipo de aplicação. Das aplicações analisadas apenas a *Jitt.Travel* e *Google Trips* têm essa componente.

Uma característica presente em todas as aplicações, é o suporte em **vários idiomas**. Esta é uma característica essencial neste tipo de aplicações, uma vez que o público alvo é maioritariamente turistas estrangeiros.

Os turistas/viajantes optam muitas vezes por viajar sem guias ou empresas de viagens e , como tal, surge a necessidade de recorrerem a aplicações para obter informação sobre os pontos a visitar, como uma pequena história ou fatos interessantes que dão uma beleza ao local visitado o proporcionando uma melhor experiência de viagem. Apesar de este ser um aspeto fundamental, apenas uma das aplicações apresenta **conteúdo interativo** (*Jitt.Travel*), como áudio e texto sobre os locais.

Após a análise das várias aplicações, concluiu-se que nenhuma delas reúne todas as vertentes consideradas essenciais . A aplicação desenvolvida ao longo deste dissertação tem como objetivo abranger cada uma das funcionalidades mais relevantes de cada uma das aplicações analisadas, bem como acrescentar algumas funcionalidades novas, que poderão enriquecer grandemente a aplicação:

- Sugestão automática de percursos baseados na estimativa de tempo de visita para cada local; sendo esse tempo calculado através das visitas anteriores a esse mesmo local e *feedback* de utilizadores;
- Sugestão de sítios para visitar de acordo com a experiência do utilizador ou o grupo de utilizadores onde se insere;
- Interação com outros utilizadores; visibilidade sobre localização dos utilizadores seguidos por mim ou com os mesmos interesses que eu;

### 3. Metodologia Adotada - ICONIX

A metodologia de trabalho escolhida para o desenvolvimento do sistema é a metodologia ICONIX. Neste capítulo, pretende-se dar a conhecer de forma sucinta alguns dos conceitos subjacentes à metodologia utilizada, permitindo que os leitores menos familiarizados com estes assuntos possam compreender o trabalho desenvolvido.

#### 3.1. Enquadramento

As metodologias de desenvolvimento de software estão hoje agrupadas em dois grupos: as metodologias ágeis e as metodologias tradicionais (Carvalho *et al.*, 2012).

Uma **metodologia ágil** distingue-se pela maior proximidade com o cliente e tempos de entrega muito menores do que nas metodologias tradicionais. Este tipo de metodologia proporciona ao cliente um acompanhamento mais próximo do processo de desenvolvimento de software, visto que vai recebendo versões de software mais regularmente, possibilitando, assim, que dê a sua opinião sobre a evolução do desenvolvimento e, conseqüentemente, minimizando o risco de um produto fora das suas expectativas. Este tipo de metodologia tornou-se muito popular nos últimos anos, devido também às tecnologias que surgiram e necessidades cada vez mais complexas dos clientes (Tomás, 2009).

Numa outra perspetiva, nas **metodologias tradicionais** a adaptação a alterações propostas pelo cliente é muito mais difícil, devido aos prazos de entrega elevados, causando assim a necessidade de um elevado número de recursos. A utilização deste tipo de metodologias é mais comum em projetos de grande dimensão (Carvalho *et al.*, 2012).

A **Tabela 3** apresenta as principais diferenças entre metodologias ágeis e metodologias tradicionais.

Tabela 3 - Metodologias Ágeis vs Metodologias Tradicionais

Característica	Metodologias Ágeis	Metodologias Tradicionais
<b>Tempo de entrega</b>	Curto	Elevado
<b>Proximidade ao cliente</b>	Maior proximidade	Menor proximidade
<b>Acompanhamento do processo por parte do cliente</b>	Cliente acompanha o processo	Cliente tem pouco contacto com todo o processo
<b>Quantidade de versões de software</b>	Entrega regular de versões melhoradas	Apenas uma entrega final

Após análise das grandes diferenças entre os dois tipos de metodologias, optou-se por seguir uma metodologia ágil no desenvolvimento deste projeto, devido à tecnologia e características do mesmo. De entre várias metodologias ágeis estudadas, a metodologia ICONIX destacou-se como sendo aquela que mais se aproximava ao pretendido e, como tal, foi a adotada para este projeto.

## 3.2. ICONIX

O ICONIX define-se como uma metodologia prática e simples, conseguindo ser poderosa e com uma componente de análise e representação dos problemas sólida e eficaz. Considera-se, tendo em conta o seu nível de complexidade, entre RUP (Kruchten, 2004) e a simplicidade do XP (Beck and Andres, 2012). De entre as suas características principais, destacam-se :

- Utiliza um subconjunto da UML: apenas 3 diagramas (casos de uso, classes e sequência), mais o de robustez, ao invés dos 14 diagramas da UML existentes;
- “Rastreabilidade dos Requisitos” - permite “obrigatoriamente” verificar em todas as fases se os requisitos estão a ser atendidos - todos os requisitos são associados a casos de uso e classes, que formam o eixo de sustentação do processo;
- É iterativo e incremental - várias iterações ocorrem entre o desenvolvimento do modelo de domínio e a modelação dos casos de uso, enquanto que o modelo estático é incrementalmente refinado pelo modelo dinâmico;
- É baseado nas questões fundamentais da orientação a objetos:
  - o que fazem os atores? - **Casos de Uso**
  - quais os objetos do mundo real? - **Modelo de Domínio**
  - quais os objetos relacionados com os casos de usos? – **Diagrama de Robustez**
  - como colaboram os objetos entre si? – **Diagrama de Sequência**
  - como será realmente construído o software? – **Diagrama de Classes**

Como se pode observar no diagrama apresentado na **Figura 39**, o processo ICONIX divide-se em dois fluxos, que andam em paralelo: um fluxo dinâmico, para representar os aspetos comportamentais do software, e um fluxo estático, para expressar os aspetos estruturais do software. O ICONIX possui 4 fases:

1. **Análise de Requisitos**, na qual são utilizados protótipos de interface, o modelo de casos de uso e o Modelo de Domínio;
2. **Análise e Projeto Preliminar**, na qual é utilizado o Diagrama de Robustez e atualizado o Modelo de Domínio;
3. **Projeto Detalhado**, na qual é modelado o Diagrama de Sequência e realizada uma atualização final do Modelo de Domínio, que passa a tornar-se no Diagrama de Classes;
4. **Implementação**, na qual há a codificação e a elaboração de testes unitários.

No fim de cada fase, deverá haver revisão crítica para verificar se os artefactos estão consistentes entre si. O ICONIX deve ser usado de maneira iterativa e incremental, no qual cada ciclo de desenvolvimento captura apenas um pequeno grupo de casos de uso, que direcionará o processo até se chegar à implementação (Barata, 2016).

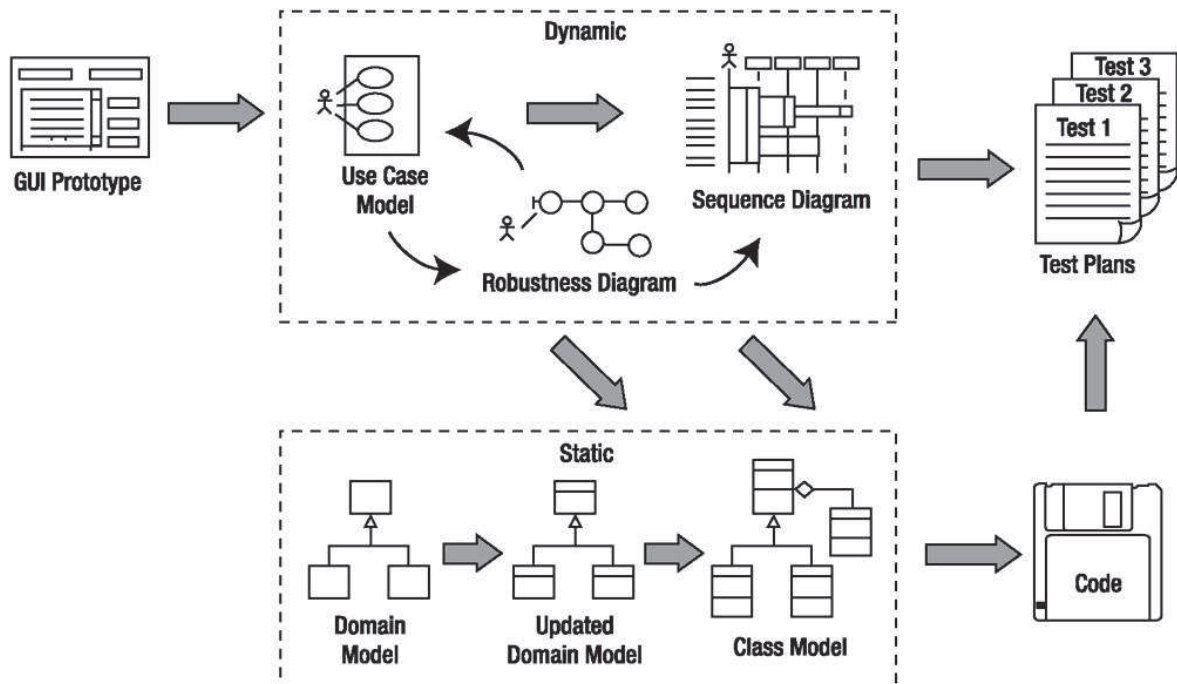


Figura 39 - ICONIX (ROSENBERG, STEPHENS and COLLINS-COPE, 2005)

### 3.2.1. Análise de Requisitos

Esta primeira fase começa pela identificação dos objetos do mundo real e todas as relações de generalização, associação e agregação entre esses mesmos objetos. Esta fase compreende as seguintes atividades: (Rosenberg and Stephens, 2007)

- **Requisitos Funcionais**

Define o que o sistema deve ser capaz de fazer. Muitas vezes, para além de um documento de especificação de requisitos, é desenvolvida a camada de interface gráfica do utilizador, de forma que utilizadores e clientes estejam em sintonia com o sistema pretendido.

- **Modelo de Domínio**

É usado para construir o primeiro glossário do projeto com o propósito de definir os principais conceitos envolvidos no software, facilitando a comunicação entre as partes interessadas. Serve também para auxiliar a construção dos casos de uso. Na prática, o Modelo de Domínio é uma versão simplificada do Diagrama de Classes, a sua representação gráfica limita-se a expressar as classes e como elas estão relacionadas, permitindo também o uso de agregação e generalização.

- **Casos de Uso**

O ICONIX usa uma abordagem baseada em cenários para representar as ações dos atores e resposta do software às mesmas. Os casos de uso detalham essas interações entre atores e sistema. São construídos com base nos requisitos funcionais e no modelo de domínio previamente definidos.

No final desta fase, deve-se realizar a revisão das atividades produzidas até o momento, de forma a assegurar que os requisitos foram suficientemente bem compreendidos, tanto

pela equipa de desenvolvimento como pelo cliente. Uma vez finalizada esta revisão, pode-se passar para a próxima fase, análise e desenho preliminar.

### 3.2.2. Análise e Desenho Preliminar

A fase de análise e desenho preliminar envolve uma exploração mais apurada dos requisitos, de forma a facilitar a passagem para a fase de projeto detalhado. Mais especificamente, o foco desta etapa é remover inconsistências encontradas nos requisitos e relacionar os casos de usos com os objetos do domínio. Essa fase engloba três atividades: (Rosenberg and Stephens, 2007)

- **Descrição dos Casos de Uso**

Descrição mais detalhada dos casos de uso, com base em textos ou tabelas com maior nível de detalhe em relação aos esquemas elaborados na primeira fase, descrevendo os principais cenários e cenários de exceção.

- **Análise de Robustez:**

A análise de robustez tem como objetivo principal suavizar a lacuna entre os requisitos e o projeto, bem como esclarecer os casos de uso e encontrar possíveis fontes de ambiguidade. Esta análise usa como suporte gráfico o Diagrama de Robustez, um híbrido entre o Diagrama de Classes e o Diagrama de Atividades da UML, que representa um caso de uso utilizando três estereótipos de classes: classes de interface, que são responsáveis pela comunicação do exterior (atores) com os componentes internos do software; classes de entidade, que são responsáveis pela retenção de dados, armazenando todas as informações essenciais ao software; e classes de controlo, que orquestram todas as trocas de mensagens entre as demais classes. Além de amenizar a dificuldade da passagem dos requisitos para o projeto, o Diagrama de Robustez leva à descoberta de novas classes que não foram percebidas na primeira versão do Modelo de Domínio (Jacobson, 1992).

- **Atualização do Modelo de Domínio:**

As classes e atributos descobertos pela atividade anterior forçam uma atualização do Modelo de Domínio criado na primeira fase.

A revisão no fim da segunda fase do processo tem como objetivo analisar se o Diagrama de Robustez corresponde exatamente ao seu caso de uso e se as classes e atributos descobertos foram incorporados ao Modelo de Domínio, mantendo a consistência entre estes três artefactos e assegurando a passagem para a etapa seguinte sem esses problemas.

### 3.2.3. Projeto

O objetivo desta fase é detalhar o comportamento de cada caso de uso por meio de Diagramas de Sequência, identificando as mensagens entre os diferentes objetos envolvidos. Esta etapa compreende duas atividades (Rosenberg and Stephens, 2007):

- **Diagrama de Sequência:**

O Diagrama de Sequência tem como objetivo mostrar a colaboração dinâmica (troca de mensagens) entre os vários objetos do software. No ICONIX, assim como na maioria dos

processos que usam a UML como linguagem de modelação, o comportamento de um caso de uso é detalhado por meio do Diagrama de Sequência. O principal objetivo desta atividade é designar comportamento, proveniente das classes de controlo, para as classes de entidade e interface (Gamma and Salgado, 2000).

- **Atualização do Modelo de Domínio:**

O Diagrama de Sequência é o responsável direto pela transformação do Modelo de Domínio em Diagrama de Classes. Após a execução da atividade anterior, o Modelo de Domínio é atualizado em termos de inclusão de métodos para as classes. Além disso, como é na fase de projeto que ocorre a transformação do domínio do problema para o domínio da solução, as classes de interface podem ser também incorporadas ao Diagrama de Classes. Assim, no final da atividade anterior, após a elaboração de um Diagrama de Sequência para um caso de uso, é necessário atualizar o Modelo de Domínio com os métodos e os seus tipos de retorno e classes de interface.

A revisão desta fase tem como objetivo verificar a qualidade do projeto, se todos os diagramas de sequência representam fielmente os casos de uso, se o Diagrama de Classes está completo, incluindo as classes relacionadas ao domínio da solução, etc.; ou seja, é aqui pretendido garantir que o “como” do projeto corresponde ao “o quê” das fases anteriores.

### 3.2.4. Implementação

Na quarta e última fase, o objetivo é transformar os Diagramas de Classes e de Sequências criados nas fases anteriores em código executável e testá-lo. Tendo em conta a necessidade do projeto, poderão ser elaborados diagramas de arquitetura, diagramas de instalação, componentes de forma a facilitar a integração do sistema assim como testes de sistema e aceitação. Esta fase compreende as seguintes etapas:(Rosenberg and Stephens, 2007)

- **Codificação:**

Realiza-se a transformação dos artefactos em código de acordo com a linguagem orientada a objetos escolhida (Java, C++, etc.).

- **Testes:**

Efetua-se os testes unitários, de integração e aceitação. Os demais testes são baseados nos casos de uso, incluindo tanto o fluxo básico como os fluxos alternativos.

A revisão desta fase tem basicamente como finalidade averiguar que o código e o projeto estão sincronizados.

## 4. Modelação do Sistema

Neste capítulo é apresentada toda a modelação do sistema que se pretende desenvolver. Seguindo a estrutura defendida pela metodologia adotada, este capítulo está dividido nas seguintes seções:

- **Análise de Requisitos:** apresentação dos requisitos funcionais da aplicação, suportados por modelo de domínio, interface gráfica com o utilizador, diagramas de navegação e casos de uso;
- **Análise e Desenho Preliminar:** análise mais detalhada dos requisitos funcionais, com base na descrição dos atores, descrição dos casos de uso, diagramas de robustez e atualização do modelo de domínio;
- **Projeto:** fase de desenho, composto pela modelação de base de dados, diagramas de sequência e atualização do modelo de domínio.

### 4.1. Análise de Requisitos

Neste subcapítulo é apresentada a análise realizada para que fosse possível o desenvolvimento da aplicação. São apresentados os requisitos funcionais (de forma a criar uma primeira ideia das tarefas a realizar), o modelo de domínio (que permite ilustrar as classes que vão dar origem à aplicação), a interface gráfica (onde são representadas as versões finais dos esboços dos ecrãs), os diagramas de navegação (que permitem ilustrar a navegabilidade dentro da aplicação) e diagramas de caso de uso (que permitem ilustrar as diferentes funcionalidades dos atores).

#### 4.1.1. Requisitos

Como referido anteriormente, o objetivo desta aplicação é o apoio ao turismo e diversas atividades económicas na área do lazer, bem como a melhoria na qualidade de vida dos turistas que visitam as nossas cidades e da população local.

No Capítulo 2, procedeu-se à análise das aplicações já existentes no mercado, com o objetivo de descobrir os pontos fortes e fracos das mesmas. Com base nessa análise foi possível avançar para a elaboração de um conjunto de requisitos fundamentais para a aplicação a desenvolver neste projeto.

Optou-se pela divisão da aplicação por vários perfis de utilizador de forma a que cada um deles fosse um ator com um papel específico na mesma. O sistema deve permitir a interação entre os vários atores com as diversas funções, sendo que cada um desempenha um papel importante no sistema para que este funcione corretamente.

#### **Atores do sistema**

- Visitante;
- Utilizador;
- Organizador;
- Revisor;

### **Requisitos funcionais iniciais:**

#### **Visitante**

- Aceder à aplicação;
- Criar conta;
- Ver lista de eventos.

#### **Utilizador**

- *Homepage* direcionada para as suas funções;
- Visualizar eventos;
- Aceder à lista de eventos sugeridos tendo em conta os seus interesses;
- Adicionar fotos, fazer comentários e avaliar eventos;
- Organizar viagens;
- Ver percursos automáticos com base nos seus interesses;
- Aceder e trocar mensagens com outros atores;
- Procurar utilizadores com os mesmos interesses.

#### **Organizador**

- Criar e publicar eventos;
- Receber observações sobre os seus eventos rejeitados, por parte do revisor;
- Aceder à lista de todos os seus eventos (eventos aprovados, pendentes e rejeitados);
- Aceder e trocar mensagens com os outros atores;
- Listas de eventos direcionadas para si
- Consultar relatórios com o estado dos seus eventos.

#### **Revisor**

- Aprovar ou rejeitar eventos;
- Adicionar observações ao rejeitar eventos;
- Consultar relatórios sobre os eventos que aprova ou rejeita;
- Consultar listas de eventos que já aprovou;
- Aceder e trocar mensagens com outros atores.

**Nota:** Estes requisitos foram identificados numa fase inicial, podendo haver uma evolução dos mesmos ou alteração consoante o desenvolvimento da aplicação.

### 4.1.2. Modelo de Domínio

Na **Figura 40** está representado o modelo de domínio, que tem como objetivo agrupar todos os objetos e relações que podem existir entre eles. Este é o modelo de domínio resultante da fase inicial e será atualizado durante as várias fases da metodologia ICONIX.

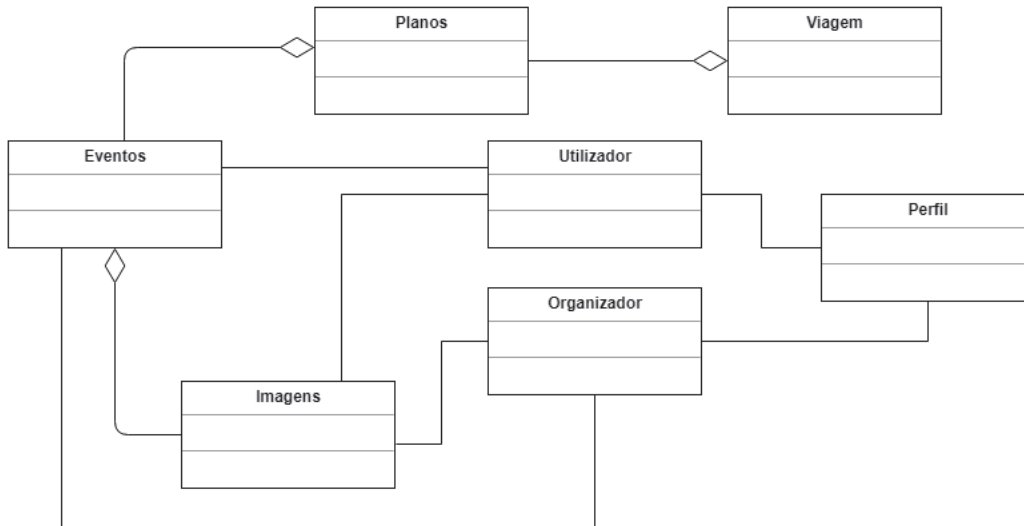


Figura 40 - Modelo de domínio

### 4.1.3. Interface Gráfica

O protótipo da interface gráfica retrata os principais cenários de navegabilidade, dos principais atores, bem como as principais vistas da aplicação. A elaboração destes ecrãs tem como objetivo associar os requisitos a algo gráfico e poder identificar logo de início eventuais incompatibilidades dos ecrãs com os requisitos. Este protótipo foi realizado com recurso à ilustração dos *storyboards*.

#### **Login**

Terminado o *loading* da aplicação e caso o utilizador não tenha o *login* já efetuado, deverá ser apresentado o ecrã de *login*, representado na **Figura 41**.

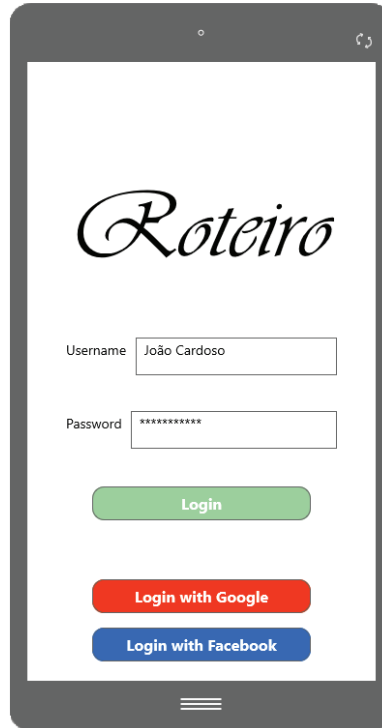


Figura 41 - Ecrã ilustrativo de login ou registo

### **Menu**

Após *login* efetuado com sucesso, deverá ser apresentado o menu da aplicação - **Figura 42**. O menu contém todas as funcionalidades existentes para cada ator e deverá ser um dos principais meios de navegação dentro da aplicação. Inicialmente foi pensado que deveria existir uma barra no fundo de cada página com as ações principais para cada ator, para que fosse mais fácil aceder às principais funcionalidades de cada um. Esta ideia acabou por ser substituída por um *dashboard*.

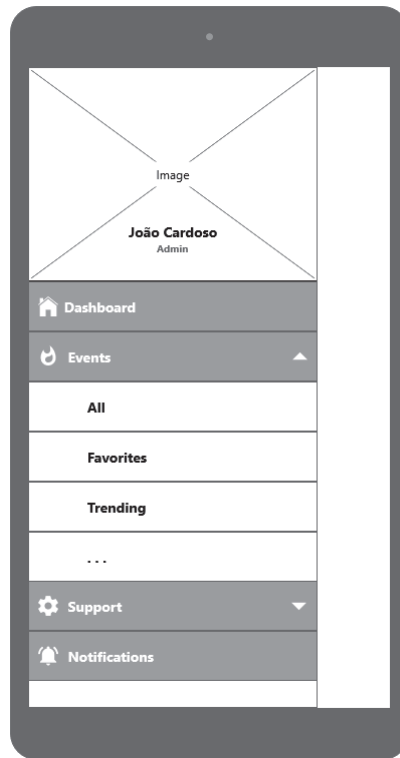


Figura 42 - Ecrã ilustrativo do menu

### Dashboard

Na **Figura 43** é possível observar a distribuição no *dashboard* de todas as ações dos vários atores. Serão sempre mostradas as 6 funcionalidades principais, que poderão variar de ator para ator. Ainda neste ecrã, é possível perceber se o utilizador foi notificado por parte de outros utilizadores.

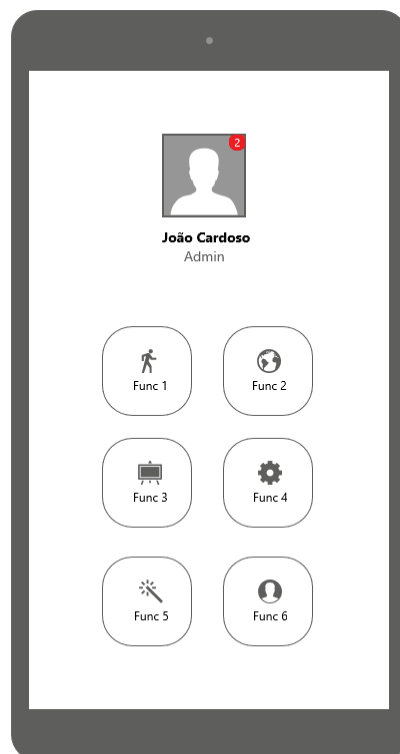


Figura 43 - Ecrã ilustrativo do dashboard

### **Lista de Eventos**

Tal como representado na **Figura 44**, pretende-se que a lista base da aplicação seja a listagem dos eventos genéricos. Partindo daqui, cada ator terá depois acesso a conteúdo diferente e diferentes filtros, dependendo sempre do seu perfil e funcionalidade escolhida em cada momento.

Todas as listas deverão estar devidamente estruturadas e conter a informação essencial dos eventos, para que o utilizador consiga rapidamente identificar quais os que lhe poderão interessar e, aí sim, avançar para o detalhe dos mesmos, e para que os organizadores tenham acesso aos eventos populares ou pendentes.

Cada evento da lista deve, então, reunir as principais características detalhadas abaixo:

- **Categoria:** cada evento pode ter mais que uma categoria, sendo que aquela que for definida como a principal será a apresentada na lista de eventos; as restantes poderão ser consultadas no detalhe do evento. Exemplos de categorias: "*Nightlife*", "*Tasting*", etc.
- **Título;**
- **Localização;**
- **Motivo** pelo qual este evento surge na lista: pode ser baseado nos interesses do utilizador, baseado nos interesses dos seus amigos, devido à classificação do evento, etc.;
- **Número de pessoas que vão estar presentes no evento:** caso existiam utilizadores (seguidos) que vão a esse evento serão apresentados;
- **Data e Horário:** se for um evento pontual, será apresentada a data do evento e horário; se for um evento recorrente, será apresentada a frequência (por exemplo, 'diariamente') e o horário;
- **Preço;**
- **Mostrar interesse:** o evento tem disponível um símbolo na forma de coração onde é permitido mostrar o interesse pelo evento em questão (apenas disponível para o ator utilizador).

A **Figura 44**, mostra o ecrã representativo da lista de eventos.

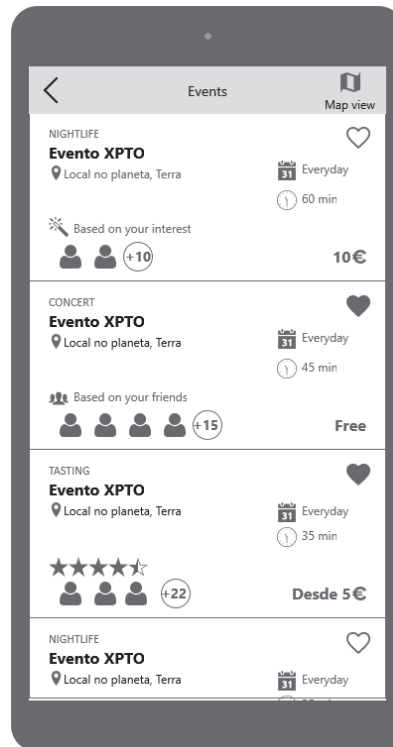


Figura 44 - Ecrã ilustrativo da lista de eventos

A lista pode ser visualizada em modo mapa, onde são apresentados todos os pontos de interesse – semelhante ao ilustrado na **Figura 48**.

### **Criação de Evento**

A criação de eventos está apenas acessível ao organizador, ou seja, apenas este pode criar e divulgar eventos.

A criação do evento foi organizada em 3 fases:

- **O destaque:** título e descrição são duas características do evento que devem destacar o mesmo e são configurados na primeira fase;
- **O detalhe:** nesta fase é configurado o tempo que deve ser necessário para visitar o evento, as categorias nas quais o evento se insere e a opção de publicar automaticamente no final da criação;
- **A Localização:** na última fase é colocada a localização do evento.

A **Figura 45** representa como foi desenhada a criação de evento.

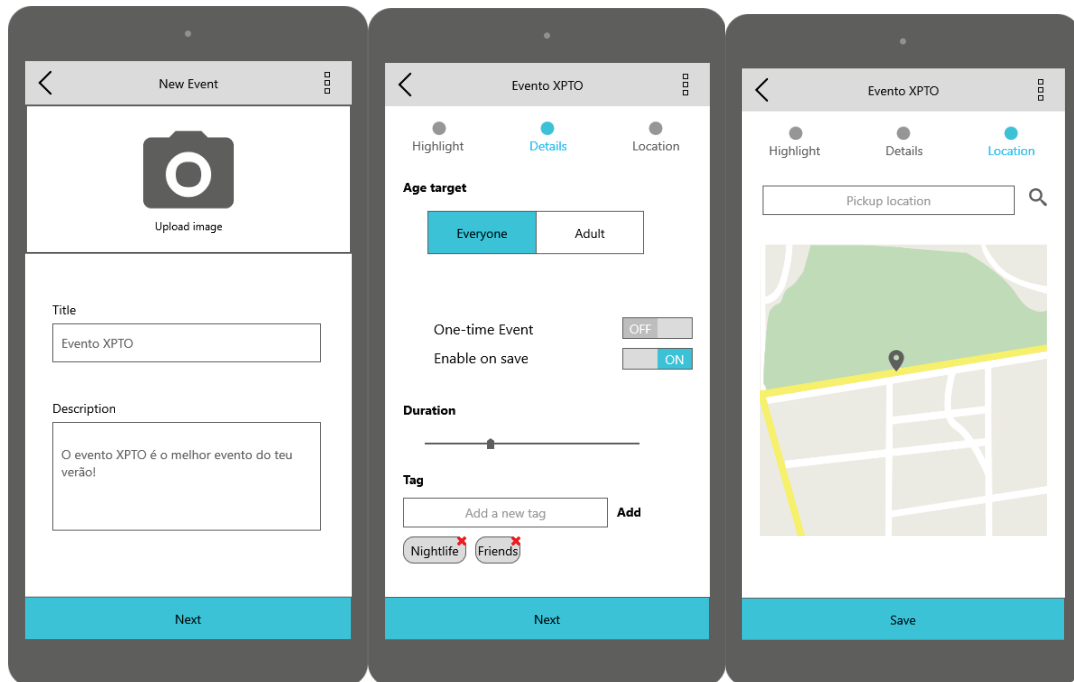


Figura 45 - Ecrã de criação do evento (3 fases)

### **Aprovação e Rejeição de Evento**

A aplicação foi desenhada para que qualquer pessoa com perfil de organizador possa colocar eventos disponíveis na mesma. Contudo, isto poderá gerar conteúdos repetidos, conteúdos fora do âmbito da aplicação, informação errada, etc. Assim, para validar a informação inserida, foi definido o perfil de revisor. A função do revisor é aceitar ou rejeitar eventos e, para tal, tem à sua disposição o ecrã representado na **Figura 46**.



Figura 46 - Ecrã de aprovação ou rejeição de eventos

Este ecrã apresenta todos os eventos dos quais o revisor é responsável. O revisor poderá filtrar os seus eventos com base no estado dos mesmos (aprovados, rejeitados, pendentes). Quando um evento é rejeitado terá de conter uma nota com o motivo da rejeição.

### Evento

É considerado evento tudo o que seja atividade e possa ser divulgado: um monumento, um ponto de interesse, uma festa, um restaurante, uma feira de vinhos, uma exposição, etc. Neste tipo de aplicação, o evento é o ponto principal. Como tal, pretende-se que o ecrã representativo do mesmo esteja bem estruturado e apelativo, de forma a cativar os utilizadores para o evento em questão.

*“Pictures are not only more effortless to recognize and process than words, but also easier to recall. When words enter long-term memory they do so with a single code. Pictures, on the other hand, contain two codes: one visual and the other verbal, each stored in different places in the brain (Paivio). The dual-coding nature of images allows for two independent ways of accessing visual memories, increasing the odds of remembering at least one of them.” (Dewan, 2015)*

Como é referido, as imagens cativam mais os utilizadores do que texto; portanto, optou-se por colocar as principais imagens no topo da página do evento, de forma a criar um primeiro impacto positivo aos utilizadores.

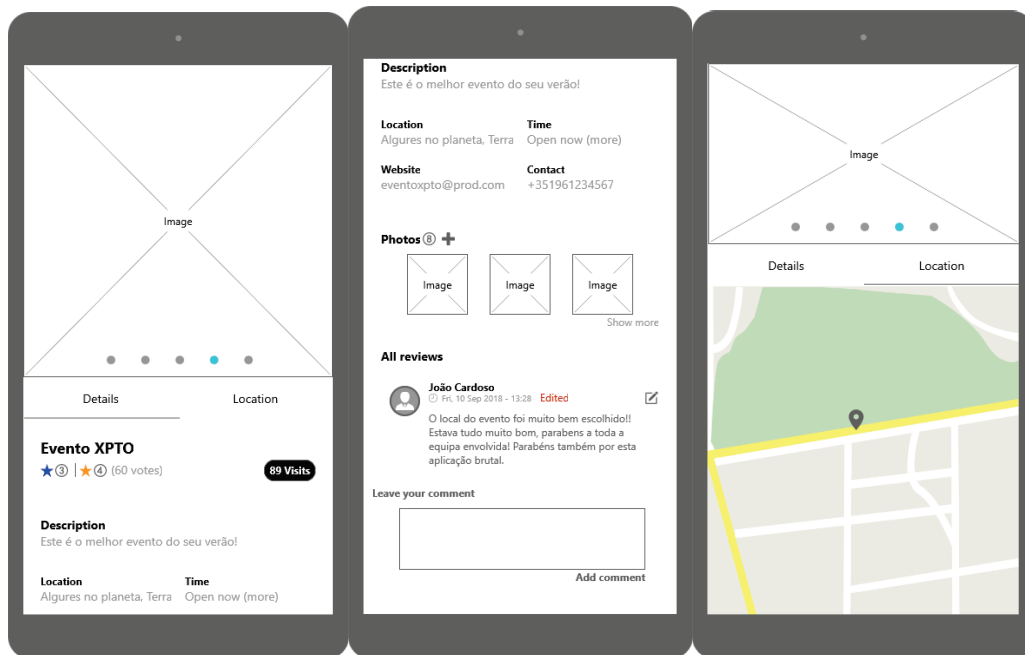
Na página do evento, depois das imagens do mesmo, é possível navegar para o seu detalhe ou localização – **Figura 47**.

Na página de detalhe pretende-se que sejam apresentadas as principais características do evento, tais como:

- Título;
- Número de utilizadores que já foram e que estão a planear ir ao evento;
- Classificação média dada pelos utilizadores;
- Classificação dada pelo utilizador, caso já a tenha efetuado;
- As principais imagens que os utilizadores já colocaram. Caso ninguém tenha colocado imagens, é feito um incentivo para que aquele utilizador seja o primeiro a fazer o *upload* de fotos;
- Comentários ao evento.

Na página da localização é apresentado um mapa com as coordenadas do ponto de interesse.

Na **Figura 47** é possível ver os ecrãs representativos do detalhe e localização do evento.



**Figura 47** - Ecrã do detalhe de um evento

## **Plano**

Esta é uma das funcionalidades inovadoras da aplicação, o planeamento de atividades com base na relevância que os vários pontos de interesse têm para o utilizador. O ecrã deve ser simples e direto, de forma a permitir navegar entre os vários pontos e ter uma visão geral do plano e os pontos de interesses que dele fazem parte.

À semelhança da lista, neste ecrã deve existir também a vista de mapa, com o objetivo de o utilizador ter a visibilidade da localização de todos os eventos. A representação deste ecrã está na **Figura 48**.

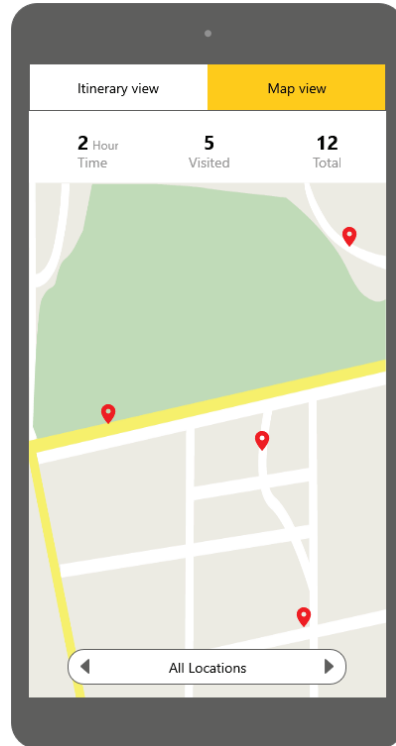


Figura 48 - Ecrã de planeamento de atividades - visão do mapa

O ecrã do detalhe do planeamento contém o tempo total do planeamento, os eventos totais e os eventos onde o utilizador já foi. Pretende-se também adicionar um *timeline* para o utilizador ter controlo do seu plano e uma visibilidade temporal. Este ecrã está representado na **Figura 49**.

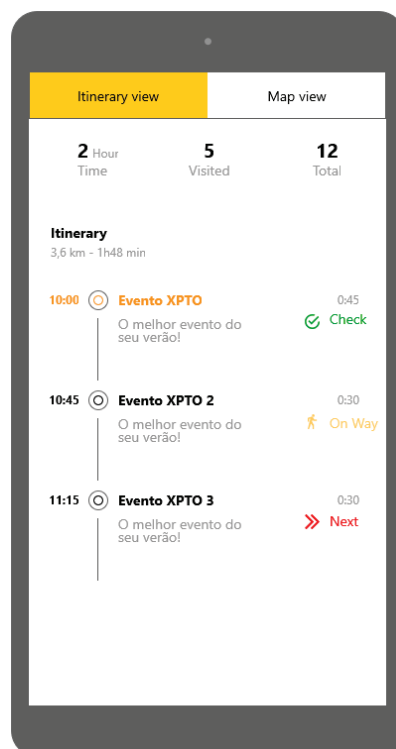


Figura 49 - Ecrã de planeamento de atividades - visão do itinerário

### **Report Gráfico**

O ecrã de *report* deverá ser um ecrã onde organizador e revisor podem ver o panorama geral do seu trabalho. O organizador poderá ver os eventos com mais adesão, mais comentários, etc. O revisor poderá ver o estado de todos os seus pedidos de publicação de eventos, quantos dos eventos aprovados têm falhas reportadas, etc.

#### **4.1.4. Diagramas de Navegação**

No capítulo anterior foram apresentados isoladamente os principais cenários de navegabilidade, dos principais atores, bem como as principais vistas da aplicação. Neste capítulo, pegou-se nos ecrãs desenhados anteriormente e criou-se um diagrama de navegação para cada um dos atores, com o objetivo de facilitar a compreensão do funcionamento da aplicação nas suas principais funcionalidades. A navegação pode sofrer atualizações até a fase final do trabalho desenvolvido.

## Utilizador



Figura 50 - Diagrama de navegação do utilizador

## Organizador



Figura 51 - Diagrama de navegação do organizador

## Revisor



Figura 52 - Diagrama de navegação do revisor

### 4.1.5. Casos de Uso

Os casos de uso são uma ferramenta que permite identificar e representar visualmente as funcionalidades de cada ator no sistema. Assim, nesta secção, optou-se por apresentar os casos de uso de cada ator num diagrama, para permitir ao leitor ter uma visão global dos casos de uso existentes para cada um. Posteriormente, na secção 4.2.2, serão descritos detalhadamente todos os casos de uso aqui representados.

Adicionalmente, é aqui apresentado um diagrama com a visão global do sistema, representativa da interação dos vários tipos de atores com o sistema - **Figura 53**.

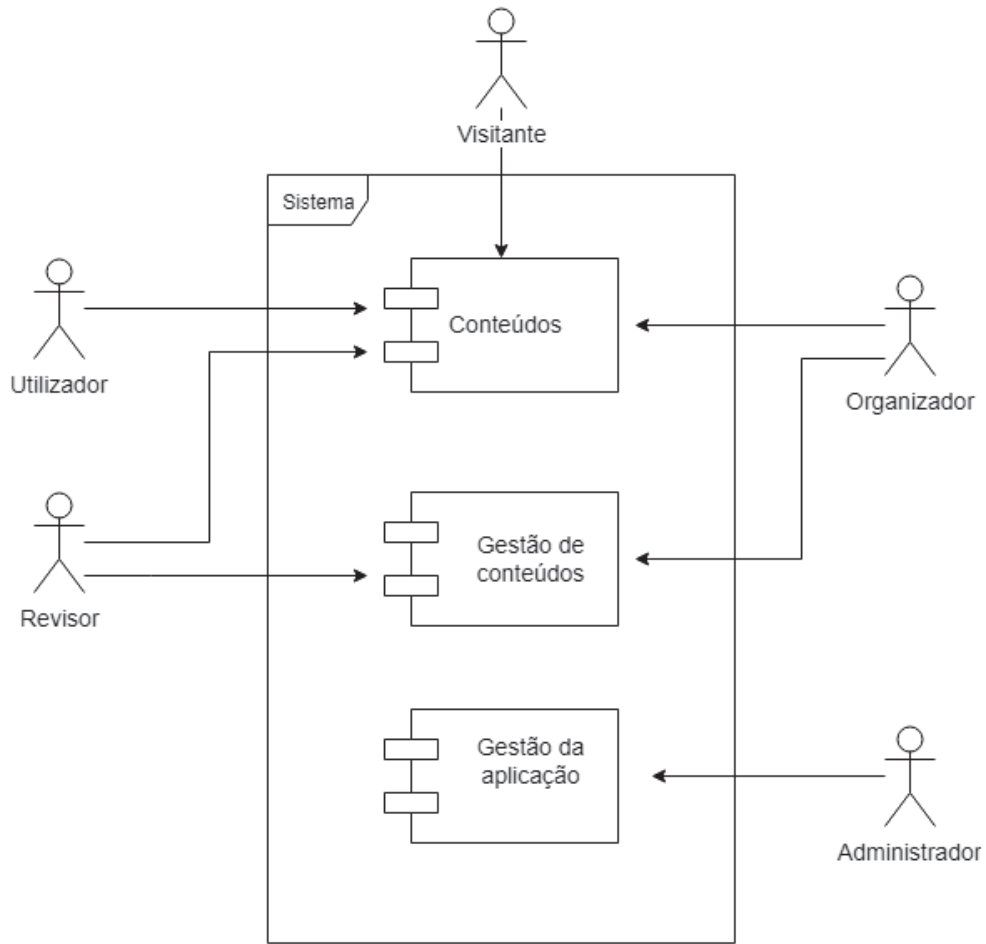


Figura 53 - Interação dos atores com o sistema - Visão Global

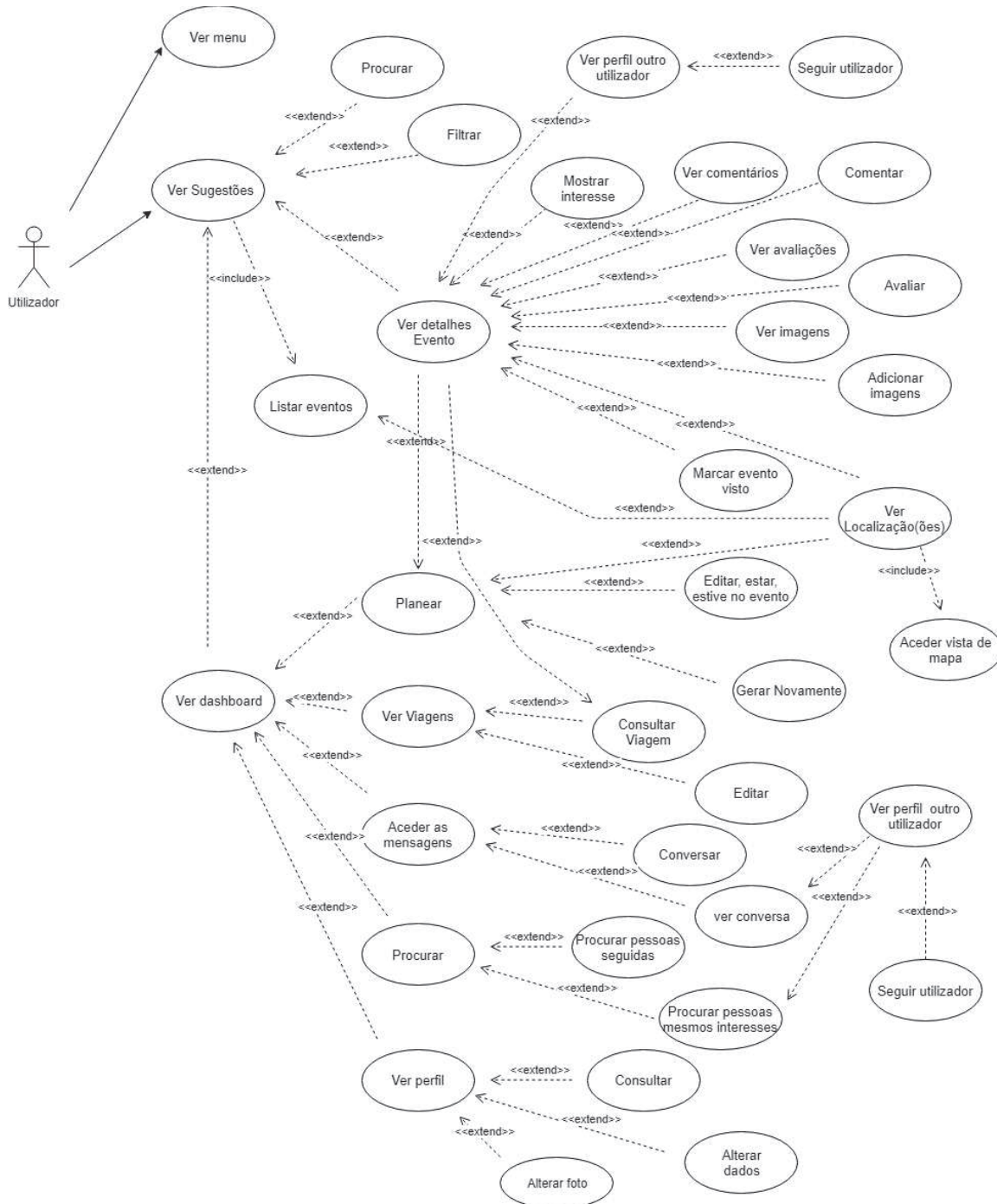
**Visitante**



Figura 54 - Diagrama de casos de uso - Visitante

Na **Figura 54** podemos ver a interação que o visitante tem na aplicação. Este ator pode efetuar o registo na aplicação e posteriormente o *login*, tornando-se assim num dos outros tipos de atores.

**Utilizador**



**Figura 55 - Diagrama de casos de uso - Utilizador**

O diagrama da **Figura 55** mostra os casos de uso do utilizador no sistema. O utilizador é o ator com mais funcionalidades na aplicação, sendo que é o ator fundamental desta.

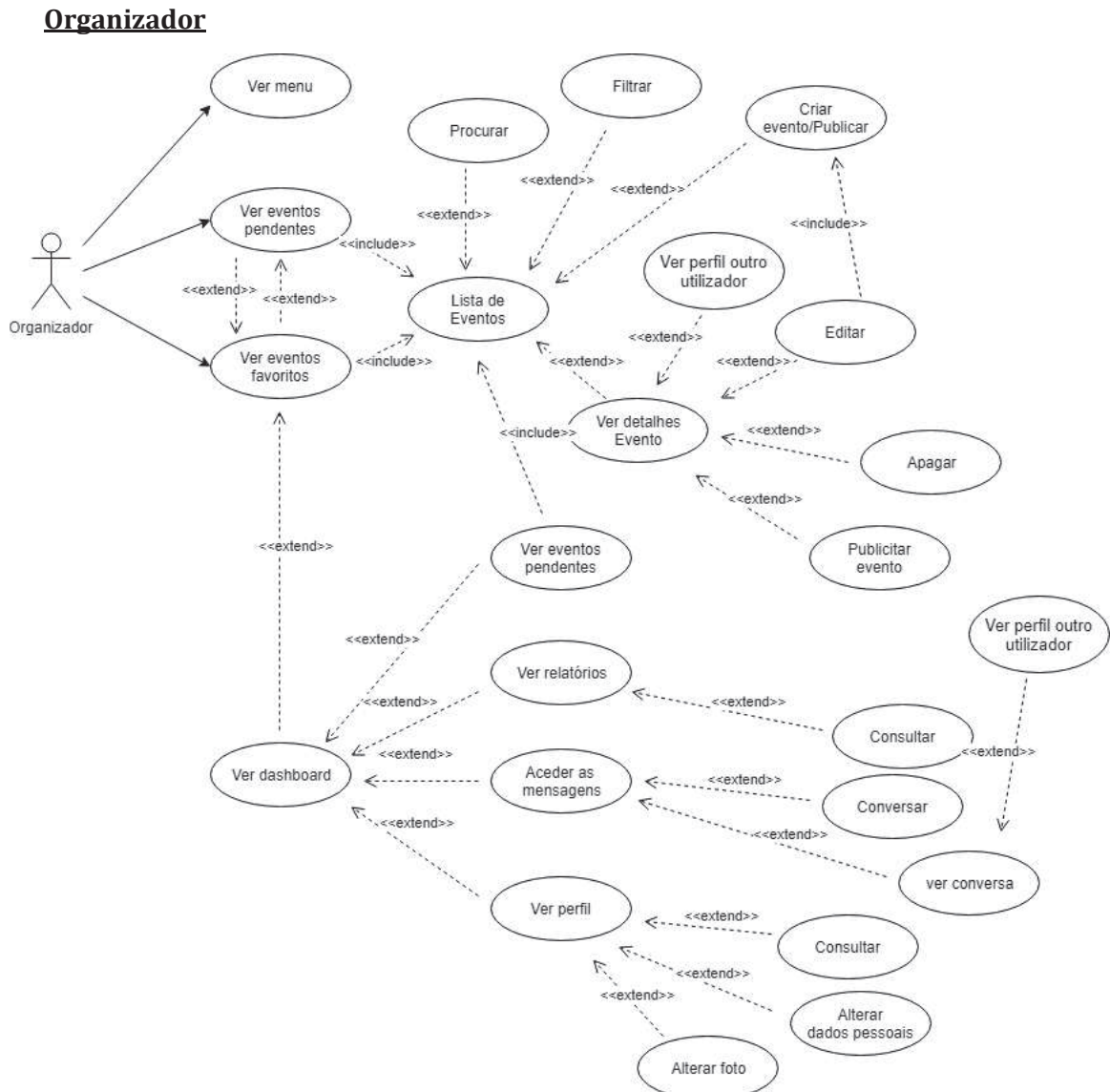


Figura 56 - Diagrama de casos de uso - Organizador

Na **Figura 56** são apresentados os casos de uso do organizador na aplicação. Este é o ator responsável pela criação de eventos.

## Revisor

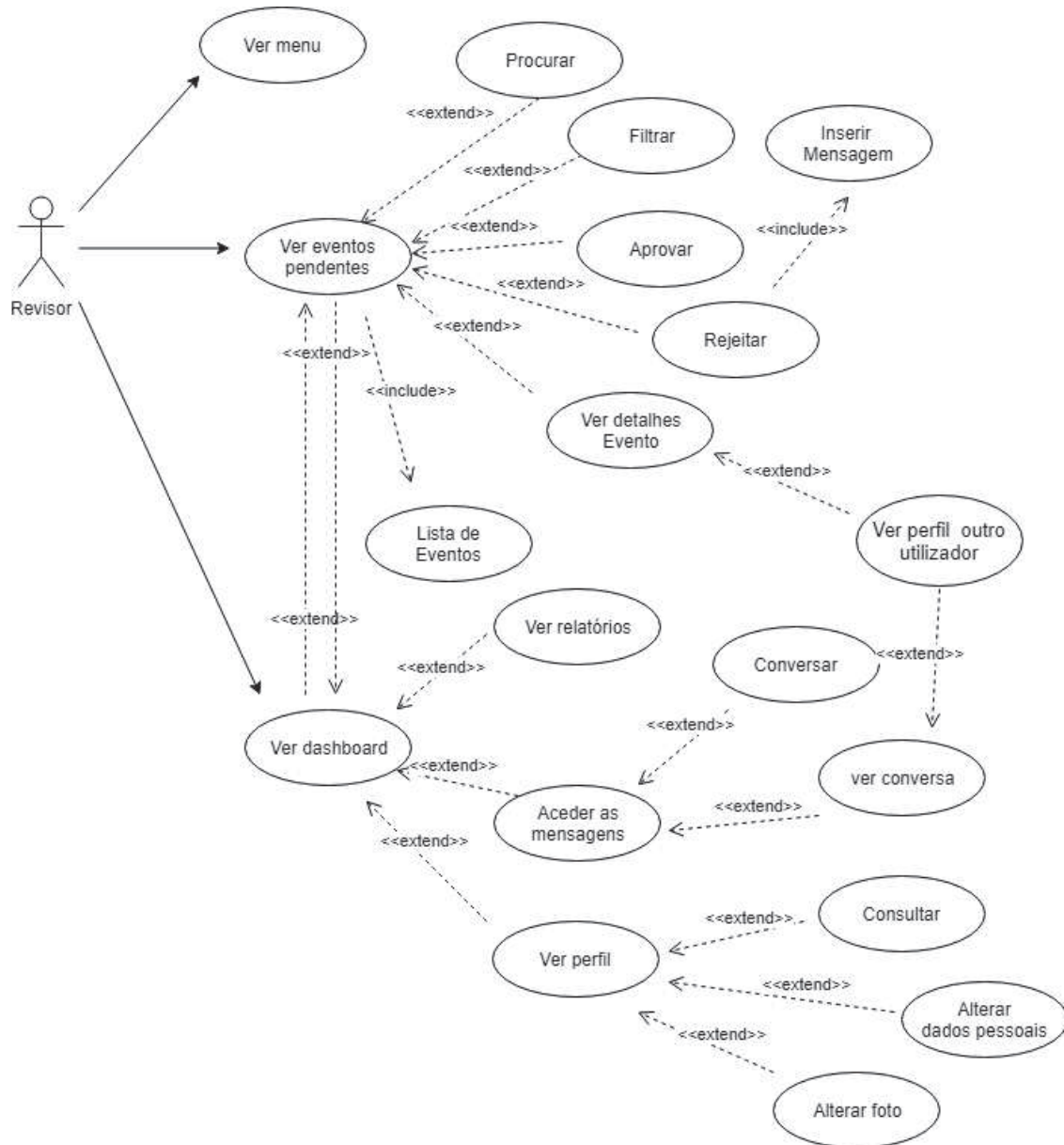


Figura 57 - Diagrama de casos de uso - Revisor

Na **Figura 57** são apresentados os casos de uso do revisor. O revisor é o ator responsável pela aprovação dos conteúdos disponibilizados.

## 4.2. Análise e Desenho Preliminar

Neste subcapítulo é apresentada a análise e o desenho preliminar. Começa por ser apresentada a descrição dos atores e as suas características; seguidamente, é feita a descrição dos casos de uso apresentados no subcapítulo anterior, de forma a descrever cada funcionalidade detalhadamente; segue-se a análise de robustez, que tem como suporte gráfico o diagrama de robustez; por fim, é feita a atualização do modelo de domínio.

### 4.2.1. Descrição dos Atores

#### Visitante

Ator que tem menos impacto na aplicação. Este apenas poderá efetuar o registo na aplicação, e, posteriormente, fazer *login*. Ao efetuar o *login*, este ator tornar-se-á num dos restantes, conforme o seu perfil. O visitante tem ainda acesso a uma lista de eventos na sua localização, aleatoriamente escolhidos.

#### **Funcionalidades:**

- Efetuar registo e posteriormente *login*;
- Ver lista de eventos.

#### Utilizador

Este ator tem acesso a muito mais ecrãs que o ator anterior, pois é o ator com o principal papel na aplicação. É maioritariamente a este ator que se destina a aplicação e neste ator que se enquadra a maior parte dos registos da mesma.

#### **Funcionalidades:**

- Visualizar listas de eventos sugeridos;
- Procurar e filtrar eventos na lista;
- Aceder ao mapa com os eventos dessa lista;
- Aceder ao detalhe dos eventos;
- Ver comentários e comentar;
- Ver avaliação e avaliar;
- Ver imagens de um evento e adicionar as que desejar;
- Reportar erros;
- Mostrar interesse nos eventos e confirmar que esteve presente;
- Ver localização dos eventos no mapa;
- Seguir utilizadores com os mesmos interesses;
- Ver quem vai ou foi a um evento;
- Planear viagens (ver lista das suas viagens, ver detalhe de cada uma e editar);
- Planeamento de percursos automáticos com base nas características do utilizador;
- Procurar utilizadores por perto com os mesmos interesses;
- Procurar pessoas que segue;
- Falar com outros atores do sistema;
- Ver perfil de outros utilizadores;
- Ver e enviar mensagens;
- Editar o seu perfil;

- Aceder ao menu.

### **Organizador**

É o ator responsável por inserir os eventos no sistema, para que possam ser depois consultados pelos restantes atores. Este ator poderá ser uma pessoa individual ou uma instituição que queira publicitar o seu espaço ou os monumentos e eventos de uma cidade (por exemplo, câmaras municipais).

#### **Funcionalidades:**

- Criar e publicar eventos;
- Ver lista de eventos pendentes de ações suas;
- Ver lista de eventos populares;
- Aceder ao detalhe de cada evento;
- Ver e enviar mensagens;
- Gerir locais e eventos – inserir, editar e apagar eventos. A inserção de eventos deve respeitar parametrizações definidas pelo administrador, como por exemplo, o evento deve conter obrigatoriamente as coordenadas GPS, horário, entre outros;
- Visualizar relatório sobre os seus eventos;
- Aplicar filtros e procurar nas suas listas;
- Aceder ao seu perfil.

### **Revisor**

Ator que tem como objetivo rever os conteúdos de eventos e permitir a publicação dos mesmos. A publicação do evento por parte do organizador só finaliza depois de aceite pelo revisor. No caso de rejeição, o revisor necessita de anexar uma observação com o motivo dessa mesma rejeição. Este evento rejeitado passa novamente para o organizador, para que este corrija o problema detetado. Se o evento for aprovado, este é disponibilizado para todos os utilizadores do sistema.

#### **Funcionalidades:**

- Aceder ao evento publicado recentemente ou editado (que se encontra sem revisão);
- Possibilidade de negar a publicação de determinado evento ou aprovar;
- Ver relatórios;
- Gerir propriedades de eventos;
- Ver e enviar mensagens;
- Ver eventos;
- Aplicar filtros e procurar nas suas listas;
- Ver detalhes dos eventos.

#### 4.2.2. Descrição dos Casos de Uso

Nesta secção são descritos os casos de uso identificados na secção 4.1.5 e explicada a importância de cada um deles. Há casos de uso exclusivos de apenas um ator, enquanto que outros são comuns a vários atores. Como tal, serão aqui apresentados todos os casos existentes e será mencionado na tabela a que ator estão associados. Assim sendo, cada tabela descritiva do caso de uso contém informação sobre:

- Título;
- Ator Primário;
- *Stakeholders*;
- Pré-Condição;
- Garantia Mínima;
- Sucesso Garantido;
- *Trigger*;
- Cenário Principal de Sucesso;
- Exceções.

##### **Caso de Uso Registrar**

Uma das primeiras ações no sistema é o registo na aplicação, feito pelo ator visitante. É após este passo que o ator visitante evolui para um dos outros atores. Assim sendo, o registo é essencial para que existam os diversos atores no sistema. O detalhe deste caso de uso é descrito na **Tabela 4**.

**Tabela 4** - Caso de uso registrar

<i><b>Título:</b> Registrar</i>	
<i><b>Ator Primário:</b></i>	Visitante
<i><b>Stakeholders:</b></i>	Visitante
<i><b>Pré-Condição:</b></i>	Nenhuma
<i><b>Garantia Mínima:</b></i>	O sistema exibe uma mensagem de erro caso não seja possível aceder à funcionalidade registrar o visitante
<i><b>Sucesso Garantido:</b></i>	O ator fica registado no sistema
<i><b>Trigger:</b></i>	Nenhum
<i><b>Cenário Principal de Sucesso:</b></i>	<ol style="list-style-type: none"> <li>1. O visitante clica no botão “Registrar”</li> <li>2. O visitante regista-se</li> <li>3. O visitante preenche o formulário de inscrição</li> <li>4. O visitante utiliza outro método de autenticação disponível (Google, Facebook)</li> <li>5. Os dados são enviados para o sistema</li> <li>6. O sistema valida o registo do utilizador</li> <li>7. O sistema indica que o registo foi efetuado com sucesso</li> </ol>
<i><b>Exceções:</b></i>	<ul style="list-style-type: none"> <li>• No ponto 4, caso a ligação ao servidor ou à base de dados falhe, o sistema retorna uma mensagem de erro e o caso de uso termina</li> <li>• No ponto 5, caso o sistema tenha dificuldades em contactar com a aplicação, o sistema não retorna a mensagem sobre o registo efetuado</li> </ul>

### **Caso de Uso Login**

O acesso a **Guia.me** é realizado através do *login* (autenticação) por qualquer um dos atores, sendo que antes do acesso o ator tem o papel de visitante. O *login* permite o acesso a todas as funcionalidades do respetivo ator. O caso de uso representativo do *login* é detalhado na **Tabela 5**.

**Tabela 5** - Caso de uso login

<b>Titulo:</b>	<i>Login</i>
<b>Ator Primário:</b>	Visitante
<b>Stakeholders:</b>	Visitante, organizador, revisor, utilizador
<b>Pré-Condição:</b>	Utilizador registado no sistema
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro
<b>Sucesso Garantido:</b>	O visitante efetua <i>login</i> e passa a usufruir do seu perfil
<b>Trigger:</b>	O visitante seleciona uma das opções de <i>login</i>
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>1. O visitante introduz as credenciais de acesso</li> <li>2. O visitante clica no botão de “<i>Login</i>”</li> <li>3. O visitante é autenticado no sistema</li> <li>4. O visitante passa a ser um dos 3 atores no sistema (utilizador, organizador ou revisor)</li> </ol>
<b>Exceções:</b>	<ul style="list-style-type: none"> <li>• No ponto 1, a aplicação não carrega com sucesso e o ecrã de <i>login</i> não é carregado, aparecendo uma mensagem de erro</li> <li>• No ponto 2, a aplicação não consegue aceder ao servidor para validar o acesso; o sistema retorna uma mensagem de erro a pedir que tente mais tarde</li> <li>• No ponto 2, as credenciais não são válidas e a autenticação falha; o sistema retorna uma mensagem de erro para o visitante introduzir as credenciais corretas</li> <li>• Nos pontos 3 e 4, a ligação à base de dados falha, o sistema retorna uma mensagem de erro e o caso de uso termina</li> </ul>

### **Caso de Uso Recuperar Palavra-Passe**

Caso o ator o pretenda, poderá recuperar a sua palavra-passe de acesso ao **Guia.me**, através da opção **Recuperar palavra-passe** - **Tabela 6**.

**Tabela 6** - Caso de uso recuperar palavra-passe

<b>Titulo:</b>	<i>Recuperar palavra-passe</i>
<b>Ator Primário:</b>	Visitante
<b>Stakeholders:</b>	Visitante, organizador, revisor, utilizador
<b>Pré-Condição:</b>	Nenhuma
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro
<b>Sucesso Garantido:</b>	O visitante recupera a sua palavra passe
<b>Trigger:</b>	O visitante seleciona a opção de “recuperar palavra-passe”
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>1. O visitante seleciona a opção de “recuperar palavra-passe”</li> <li>2. O sistema apresenta o ecrã para que o visitante possa preencher</li> </ol>

<b>Sucesso:</b>	o formulário de validação 3. A palavra-passe é recuperada e enviada por email
<b>Exceções:</b>	<ul style="list-style-type: none"> <li>Nos pontos 1 e 2, os ecrãs não são carregados e o sistema retorna uma mensagem de erro</li> <li>No ponto 3, caso a ligação à base de dados falhe e não seja possível recuperar a palavra passe, o sistema retorna uma mensagem de erro</li> </ul>

### **Caso de Uso Ver Sugestões**

No momento em que o *login* no **Guia.me** é feito pelo utilizador, é lhe apresentado uma lista com os eventos mais relevantes, de acordo com os seus interesses. Ao acesso a esta lista de eventos direcionadas para um utilizador foi dado o nome de '**Ver Sugestões**' - **Tabela 7**.

**Tabela 7** - Caso de uso ver sugestões

<b>Titulo:</b>	<i>Ver Sugestões</i>
<b>Ator Primário:</b>	Utilizador
<b>Stakeholders:</b>	Utilizador
<b>Pré-Condição:</b>	O utilizador foi validado no sistema e o perfil é de utilizador
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro
<b>Sucesso Garantido:</b>	O utilizador tem acesso a lista de eventos sugeridos
<b>Trigger:</b>	O utilizador entra na aplicação depois de autenticado O utilizador clica na opção "Eventos" no seu <i>dashboard</i>
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>Acesso à sugestão de eventos <ol style="list-style-type: none"> <li>Após <i>login</i> efetuado</li> <li>Clica na opção "Eventos" no seu <i>dashboard</i></li> </ol> </li> <li>A aplicação acede à localização do utilizador</li> <li>Os eventos são carregados de acordo com a sua relevância para o utilizador</li> <li>O utilizador tem acesso a lista de eventos sugeridos</li> </ol>
<b>Exceções:</b>	<ul style="list-style-type: none"> <li>Nos pontos 1 e 4, o ecrã não é carregado e o sistema retorna uma mensagem de erro</li> <li>No ponto 3, a aplicação não consegue aceder à base de dados e o sistema retorna uma mensagem de erro</li> <li>Nos pontos 2 e 3, a aplicação tem erro no processo e o sistema retorna uma mensagem de erro</li> </ul>

### **Caso de Uso Procurar Evento**

Apesar de ser disponibilizada ao ator que está a usar a aplicação uma lista de eventos de acordo com os seus interesses, este poderá procurar diretamente determinado evento na aplicação, através do caso de uso "**Procurar Evento**" - **Tabela 8**.

**Tabela 8** - Caso de uso procurar evento

<b>Titulo:</b>	<i>Procurar evento</i>
<b>Ator Primário:</b>	Utilizador, organizador ou revisor
<b>Stakeholders:</b>	Utilizador, organizador ou revisor
<b>Pré-Condição:</b>	O ator está no ecrã da lista de eventos
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro ao ator
<b>Sucesso Garantido:</b>	O sistema apresenta os eventos que procurou
<b>Trigger:</b>	O ator clica no botão de lupa (presente no ecrã da lista de eventos)
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>1. O ator seleciona a opção de procurar evento (lupa) no ecrã lista de eventos</li> <li>2. O sistema abre uma caixa de texto para o ator inserir o texto que deseja procurar</li> <li>3. É realizada a procura nos eventos da base de dados</li> <li>4. O evento é apresentado ao utilizador no ecrã lista de eventos</li> </ol>
<b>Exceções:</b>	<ul style="list-style-type: none"> <li>• Nos pontos 2 e 4, o ecrã não é carregado e o sistema retorna uma mensagem de erro</li> <li>• No ponto 3, a aplicação não consegue aceder à base de dados e o sistema retorna uma mensagem de erro</li> </ul>

**Caso de Uso Filtrar**

Sobre a lista de eventos disponibilizada ao ator, poderá ser aplicado um filtro de pesquisa, para obter apenas os eventos que cumpram o critério de pesquisa usado. O caso de uso representativo desta situação está representado na **Tabela 9**.

**Tabela 9** - Caso de uso filtrar

<b>Titulo:</b>	<i>Filtrar</i>
<b>Ator Primário:</b>	Utilizador, organizador ou revisor
<b>Stakeholders:</b>	Utilizador, organizador ou revisor
<b>Pré-Condição:</b>	O ator está no ecrã da lista de eventos
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro ao ator
<b>Sucesso Garantido:</b>	O sistema apresenta os eventos filtrados
<b>Trigger:</b>	O ator clica na opção de filtros
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>1. O ator seleciona a opção de filtrar eventos</li> <li>2. O sistema valida o tipo de ator e disponibiliza os tipos de filtros mediante o resultado anterior</li> <li>3. O <i>popup</i> de filtros é apresentado ao ator</li> <li>4. O ator seleciona o(s) filtro(s) pretendido(s)</li> <li>5. Se houver um filtro com base na distância, é invocada a API da <i>Google</i> para calcular a distância aos eventos</li> <li>6. Os eventos são filtrados</li> <li>7. O ecrã lista de eventos é apresentado ao ator com os eventos filtrados</li> </ol>
<b>Exceções:</b>	<ul style="list-style-type: none"> <li>• Nos pontos 2, 3, 4 e 7, o ecrã não é carregado e o sistema retorna uma mensagem de erro</li> </ul>

- No ponto 6, a aplicação não consegue aceder à base de dados e o sistema retorna uma mensagem de erro
- No ponto 5, a aplicação não consegue invocar a API da *Google* e o sistema retorna uma mensagem de erro

### **Caso de Uso Ver Detalhes do Evento**

**Ver detalhes do evento** é o caso de uso que permite ao ator ver a informação completa e detalhada sobre um evento em específico - **Tabela 10**.

**Tabela 10** - Caso de uso ver detalhe de evento

<b>Titulo:</b>	<i>Ver detalhes do evento</i>
<b>Ator Primário:</b>	Utilizador, organizador ou revisor
<b>Stakeholders:</b>	Utilizador, organizador ou revisor
<b>Pré-Condição:</b>	O ator está no ecrã da lista de eventos
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro ao ator
<b>Sucesso Garantido:</b>	É apresentado o ecrã do detalhe do evento que o ator selecionou
<b>Trigger:</b>	O ator seleciona o nome do evento para o qual quer ver o detalhe
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>1. O utilizador seleciona o evento pretendido</li> <li>2. Os detalhes do evento são carregados da base de dados</li> <li>3. O ecrã de eventos é mostrado ao utilizador</li> </ol>
<b>Exceções:</b>	<ul style="list-style-type: none"> <li>• Nos pontos 1 e 3, o ecrã não é carregado e o sistema retorna uma mensagem de erro</li> <li>• No ponto 2, a aplicação não consegue aceder à base de dados e o sistema retorna uma mensagem de erro</li> </ul>

### **Caso de Uso Mostrar Interesse**

O utilizador pode assinalar num evento que tem interesse em participar no mesmo. Para tal, surgiu o caso de uso **Mostrar interesse** - **Tabela 11**.

**Tabela 11** - Caso de uso mostrar interesse

<b>Titulo:</b>	<i>Mostrar interesse</i>
<b>Ator primário:</b>	Utilizador
<b>Stakeholders:</b>	Utilizador
<b>Pré-Condição:</b>	O utilizador estar no ecrã detalhes do evento
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro
<b>Sucesso Garantido:</b>	O utilizador sinaliza o evento como sendo do seu interesse
<b>Trigger:</b>	O utilizador clica no botão mostrar interesse
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>1. O utilizador clica no botão de interesse</li> <li>2. A aplicação regista na base de dados o interesse do utilizador naquele evento</li> <li>3. O detalhe do evento mostra uma legenda na qual diz que o utilizador está interessado</li> </ol>
<b>Exceções:</b>	<ul style="list-style-type: none"> <li>• Nos pontos 1 e 3, o ecrã não é carregado e o sistema retorna uma mensagem de erro</li> </ul>

- No ponto 2, a aplicação não consegue aceder à base de dados e o sistema retorna uma mensagem de erro

### **Caso de Uso Comentar/Ver Comentários**

Foi criada a funcionalidade **comentar e ver comentários** para que fosse possível ao utilizador e organizador incluir opiniões e informação relevante no detalhe dos eventos – **Tabela 12.**

**Tabela 12** - Caso de uso comentar / ver comentários

<b>Titulo:</b>	<i>Comentar/ Ver comentários</i>
<b>Ator Primário:</b>	Utilizador, organizador
<b>Stakeholders:</b>	Utilizador, organizador
<b>Pré-Condição:</b>	O utilizador estar no detalhe do evento O organizador ser responsável pelo respetivo evento
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro ao ator
<b>Sucesso Garantido:</b>	O comentário escrito pelo ator é publicado no detalhe do evento
<b>Trigger:</b>	Escrever comentário e adicionar
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>1. O ator posiciona-se na zona dos comentários (<b>ver comentários</b>)</li> <li>2. O ator escreve o comentário</li> <li>3. O ator clica no botão adicionar</li> <li>4. A aplicação insere o comentário na base de dados</li> <li>5. O ator visualiza o seu comentário junto dos restantes, ficando disponível para todos os utilizadores após sincronização</li> </ol>
<b>Exceções:</b>	<ul style="list-style-type: none"> <li>• Nos pontos 1, 2, 3 e 5, o ecrã não é carregado e o sistema retorna uma mensagem de erro</li> <li>• No ponto 4, a aplicação não consegue aceder à base de dados e o sistema retorna uma mensagem de erro</li> </ul>

### **Caso de Uso Avaliar/Ver Avaliações**

As **avaliações** de um evento permitem aos utilizadores ver a classificação desse mesmo evento, tendo em conta opinião dos utilizadores que nele já participaram - **Tabela 13.**

**Tabela 13** - Caso de uso avaliar / ver avaliações

<b>Titulo:</b>	<i>Avaliar / Ver avaliações</i>
<b>Ator Primário:</b>	Utilizador
<b>Stakeholders:</b>	Utilizador
<b>Pré-Condição:</b>	O utilizador está no ecrã no ecrã de detalhe do evento
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro
<b>Sucesso Garantido:</b>	O utilizador atribui uma avaliação ao evento
<b>Trigger:</b>	O utilizador clica na <i>range</i> de estrelas existente no ecrã de detalhe do evento

<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>1. O utilizador posiciona-se na zona das avaliações (<b>ver avaliações</b>)</li> <li>2. O utilizador clica nas estrelas que deseja dar como classificação</li> <li>3. A aplicação insere a avaliação na base de dados</li> <li>4. É feito o cálculo médio da classificação do evento, tendo em conta a nova avaliação</li> <li>5. O evento é atualizado com a nova classificação média</li> <li>6. É apresentada a nova pontuação média e a pontuação dada pelo utilizador</li> </ol>
<b>Exceções:</b>	<ul style="list-style-type: none"> <li>• Nos pontos 1, 2 e 6, o ecrã não é carregado e o sistema retorna uma mensagem de erro</li> <li>• No ponto 2, a aplicação não consegue aceder à base de dados, o sistema retorna uma mensagem de erro</li> <li>• No ponto 4, a aplicação obtém um erro inesperado de cálculo e o valor devolvido como médio é o existente, ficando pendente o cálculo</li> </ul>

### **Caso de Uso Adicionar Imagens/Ver Imagens**

Os utilizadores podem **ver e adicionar novas imagens** a um evento para que seja possível partilhar e ver a experiência de outros utilizadores - **Tabela 14**.

**Tabela 14** - Caso de uso adicionar imagens / ver imagens

<b>Titulo:</b>	<i>Adicionar imagens / Ver imagens</i>
<b>Ator Primário:</b>	Utilizador
<b>Stakeholders:</b>	Utilizador
<b>Pré-Condição:</b>	O utilizador está no ecrã detalhes do evento
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro
<b>Sucesso Garantido:</b>	A imagem é adicionada às restantes imagens no detalhe do evento
<b>Trigger:</b>	O utilizador clica no botão "+" na zona das imagens do evento
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>1. O utilizador acede à zona de imagens (<b>ver imagens</b>)</li> <li>2. Clica no adicionar imagens</li> <li>3. Escolhe a imagem da sua galeria ou tira uma foto naquele momento</li> <li>4. A imagem é inserida na base de dados</li> <li>5. O utilizador vê a nova imagem no detalhe do evento</li> <li>6. A imagem fica disponível para todos os utilizadores após sincronização</li> </ol>
<b>Exceções:</b>	<ul style="list-style-type: none"> <li>• Nos pontos 1, 2 e 5, o ecrã não é carregado e o sistema retorna uma mensagem de erro</li> <li>• No ponto 3, erro no formato, o sistema retorna uma mensagem de erro</li> <li>• Nos pontos 4 e 6, a aplicação não consegue aceder à base de dados, o sistema retorna uma mensagem de erro</li> </ul>

### **Caso de Uso Ver Localização(ões) do(s) evento(s)**

Os diversos atores devem ter a visibilidade de onde se situa o evento (ou eventos, se estiverem posicionados numa lista). Para tal, foi criado um mapa com as coordenadas do(s) evento(s) - **Tabela 15**.

**Tabela 15** - Caso de uso ver localização

<b>Titulo:</b>	<i>Ver localização(ões) do(s) evento(s)</i>
<b>Ator Primário:</b>	Utilizador, organizador ou revisor
<b>Stakeholders:</b>	Utilizador, organizador ou revisor
<b>Pré-Condição:</b>	O ator está no ecrã da lista de eventos, ou no ecrã de detalhe do evento, ou no ecrã de planeamento
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro ao ator
<b>Sucesso Garantido:</b>	O ator vê a(s) localização(ões) do(s) evento(s)
<b>Trigger:</b>	Utilizador clica no <i>icon</i> do mapa presente no canto superior direito do ecrã O ator clica em localização na parte superior do ecrã
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>1. O ator acede a vista de mapa</li> <li>2. A aplicação carrega os dados da(s) localização(ões) para o módulo do mapa</li> <li>3. O mapa gera os pontos com a informação</li> <li>4. É apresentado o ecrã com o mapa, onde estão presentes todos os pontos dos eventos. É possível navegar entre os pontos com um navegador</li> </ol>
<b>Exceções:</b>	<ul style="list-style-type: none"> <li>• Nos pontos 1 e 4, o ecrã não é carregado e o sistema retorna uma mensagem de erro</li> <li>• Nos pontos 2 e 3, a aplicação não consegue aceder à base de dados, o sistema retorna uma mensagem de erro</li> </ul>

### **Caso de Uso Planear**

Como referido anteriormente, **Guia.me** pretendia distinguir-se pela capacidade de gerar percursos e planos automáticos com base nos eventos relevantes para um determinado ator. Para tal, foi criada a opção de **planear** - **Tabela 16**.

**Tabela 16** - Caso de uso planear

<b>Titulo:</b>	<i>Planear</i>
<b>Ator Primário:</b>	Utilizador
<b>Stakeholders:</b>	Utilizador
<b>Pré-Condição:</b>	O utilizador está no seu <i>dashboard</i>
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro
<b>Sucesso Garantido:</b>	O sistema gera um ecrã com um plano baseado nos eventos sugeridos, em forma de linha de tempo e com a possibilidade de ver no mapa
<b>Trigger:</b>	Na <i>homepage</i> o utilizador seleciona a opção de plano
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>1. O utilizador acede ao plano</li> <li>2. A aplicação acede à localização do utilizador</li> <li>3. A aplicação acede à base de dados para carregar todos os eventos</li> </ol>

<b>Sucesso:</b>	<p>e a informação necessária</p> <ol style="list-style-type: none"> <li>4. É realizado o cálculo da relevância</li> <li>5. É gerado o plano com base em toda a informação processada;</li> <li>6. Apresentação do plano em linha de tempo e mapa</li> </ol>
<b>Exceções:</b>	<ul style="list-style-type: none"> <li>• Nos pontos 1 e 6, o ecrã não é carregado e o sistema retorna uma mensagem de erro</li> <li>• No ponto 2, não é possível obter a localização do utilizador, este valor é retirado do cálculo e o caso de uso continua</li> <li>• No ponto 3, a aplicação não consegue aceder à base de dados, o sistema retorna uma mensagem de erro</li> <li>• Nos pontos 4 e 5, a aplicação tem um erro no processamento, o sistema retorna uma mensagem de erro</li> </ul>

### **Caso de Uso Editar Plano**

Depois de visualizar o plano automático, o utilizador poderá remover eventos desse mesmo plano através da opção de **editar** - **Tabela 17**.

**Tabela 17** - Caso de uso editar plano

<b>Titulo:</b>	<i>Editar plano</i>
<b>Ator Primário:</b>	Utilizador
<b>Stakeholders:</b>	Utilizador
<b>Pré-Condição:</b>	O utilizador está no ecrã do plano
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro
<b>Sucesso Garantido:</b>	O utilizador edita o plano como pretende
<b>Trigger:</b>	O utilizador remove eventos do plano
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>1. O utilizador encontra-se no plano</li> <li>2. O utilizador não gosta, não quer ir ou está atrasado para um evento</li> <li>3. O utilizador remove o evento do plano, edita a hora inicial, tempo de demora, etc</li> <li>4. O plano atualiza automaticamente podendo ir buscar outro evento</li> <li>5. O ecrã atualiza o plano</li> </ol>
<b>Exceções:</b>	<ul style="list-style-type: none"> <li>• Nos pontos 1, 2, e 5, o ecrã não é carregado e o sistema retorna uma mensagem de erro</li> <li>• No ponto 2, não é possível obter a localização do utilizador, este valor é retirado do cálculo e o caso de uso continua</li> <li>• No ponto 4, a aplicação não consegue aceder a base de dados, o sistema retorna uma mensagem de erro</li> <li>• Nos pontos 3 e 4, a aplicação tem um erro no processamento, o sistema retorna uma mensagem de erro</li> </ul>

### **Caso de Uso Participar no Evento**

Para informar a aplicação **Guia.me** da sua presença num determinado evento, o utilizador deverá fazer essa marcação manualmente na aplicação ou deverá ter o GPS ligado enquanto se movimenta pelos vários eventos de um plano - **Tabela 18**.

**Tabela 18** - Caso de uso participar no evento

<b>Titulo:</b>	<i>Participar no evento</i>
<b>Ator Primário:</b>	Utilizador
<b>Stakeholders:</b>	Utilizador
<b>Pré-Condição:</b>	O utilizador está no ecrã do seu plano
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro
<b>Sucesso Garantido:</b>	O sistema marca o evento como iniciado e mostra informação útil sobre o evento
<b>Trigger:</b>	O utilizador marca o evento como presente O sistema acede à localização e percebe que o utilizador está no local do evento
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>1. O utilizador encontra-se no plano</li> <li>2. O utilizador desloca-se até ao evento</li> <li>3. O sistema monitoriza a localização do utilizador através do sistema de GPS do dispositivo do utilizador</li> <li>4. O utilizador chega ao evento</li> <li>5. A aplicação acede à base de dados e atualiza o plano</li> <li>6. A aplicação acede à base de dados e retorna toda a informação sobre o evento</li> <li>7. A informação é apresentada no ecrã</li> </ol>
<b>Exceções:</b>	<ul style="list-style-type: none"> <li>• Nos pontos 1 e 7, o ecrã não é carregado e o sistema retorna uma mensagem de erro</li> <li>• Nos pontos 2, 3 e 4, não é possível obter a localização do utilizador, o utilizador necessita de interagir com a aplicação e o caso de uso continua</li> <li>• Nos pontos 5 e 6, a aplicação não consegue aceder à base de dados, o sistema retorna uma mensagem de erro</li> <li>• Nos pontos 2, 3 e 4, a aplicação tem um erro no processamento, o sistema retorna uma mensagem de erro</li> </ul>

**Caso de Uso Marcar Estive**

O utilizador pode marcar a sua participação passada num evento através do detalhe do evento ou do plano ou ainda de forma automática, se o seu dispositivo móvel tiver o sensor GPS ativo - **Tabela 19**.

**Tabela 19** - Caso de uso "eu estive" no evento

<b>Titulo:</b>	<i>Marcar estive</i>
<b>Ator Primário:</b>	Utilizador
<b>Stakeholders:</b>	Utilizador
<b>Pré-Condição:</b>	O utilizador está no ecrã do seu plano O utilizador está no detalhe do evento
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro
<b>Sucesso Garantido:</b>	O sistema marca o evento como visitado Caso o utilizador esteja no seu ecrã de plano, guia o utilizador até ao evento seguinte
<b>Trigger:</b>	O utilizador marca o evento onde está como concluído no ícone do lado

	<p>direito do ecrã do seu plano</p> <p>O utilizador está a executar o seu plano e, através do sensor de GPS do seu dispositivo móvel, a aplicação percebe que o utilizador se deslocou do evento para outro ponto</p> <p>O utilizador marca o evento como 'estive' no ecrã de detalhe de evento</p>
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>1. O utilizador encontra-se no plano</li> <li>2. O utilizador desloca-se para fora do evento</li> <li>3. O sistema monitoriza a localização do utilizador através do sistema de GPS do dispositivo do utilizador</li> <li>4. O utilizador está fora do evento ou marca o evento como visto</li> <li>5. A aplicação acede à base de dados e atualiza o plano e o evento</li> <li>6. A informação é apresentada no ecrã</li> </ol>
<b>Exceções:</b>	<ul style="list-style-type: none"> <li>• Nos pontos 1 e 6, o ecrã não é carregado e o sistema retorna uma mensagem de erro</li> <li>• Nos pontos 2,3 e 4, não é possível obter a localização do utilizador, o utilizador necessita de interagir com a aplicação e o caso de uso continua</li> <li>• No ponto 5, a aplicação não consegue aceder à base de dados, o sistema retorna uma mensagem de erro</li> <li>• Nos pontos 2, 3 e 4, a aplicação tem um erro no processamento, o sistema retorna uma mensagem de erro</li> </ul>

### Caso de Uso Ver Viagens

**Guia.me** tem a possibilidade de guardar informação agrupada para um determinado local, criando assim o conceito de **viagem** - **Tabela 20**.

**Tabela 20** - Caso de uso Ver Viagens

<b>Titulo:</b>	<i>Ver Viagens</i>
<b>Ator Primário:</b>	Utilizador
<b>Stakeholders:</b>	Utilizador
<b>Pré-Condição:</b>	O utilizador está no seu <i>dashboard</i>
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro
<b>Sucesso Garantido:</b>	É apresentado um ecrã com as viagens do utilizador
<b>Trigger:</b>	O utilizador carrega na opção de viagens no seu <i>dashboard</i>
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>1. O utilizador está no seu <i>dashboard</i></li> <li>2. O utilizador clica no ícone de viagens</li> <li>3. A aplicação acede à base de dados e retorna a informação sobre as viagens</li> <li>4. É apresentado o ecrã viagens do utilizador</li> </ol>
<b>Exceções:</b>	<ul style="list-style-type: none"> <li>• Nos pontos 1 e 4, o ecrã não é carregado, o sistema retorna uma mensagem de erro</li> <li>• No ponto 3, a aplicação não consegue aceder à base de dados, o sistema retorna uma mensagem de erro</li> </ul>

### **Caso de Uso Consultar Viagem**

Através da sua lista de viagens é possível consultar o detalhe de cada uma delas, com toda a informação guardada pelo utilizador - **Tabela 21**.

**Tabela 21** - Caso de uso Consultar viagem

<b>Titulo:</b>	<i>Consultar Viagem</i>
<b>Ator Primário:</b>	Utilizador
<b>Stakeholders:</b>	Utilizador
<b>Pré-Condição:</b>	O utilizador está no ecrã de viagens
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro
<b>Sucesso Garantido:</b>	O utilizador acede ao conteúdo disponibilizado relativo à viagem selecionada
<b>Trigger:</b>	O utilizador clica na viagem pretendida
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>1. O utilizador está no ecrã de viagens</li> <li>2. O utilizador clica sobre uma das suas viagens</li> <li>3. A aplicação acede à base de dados e carrega toda a informação sobre a respetiva viagem</li> <li>4. A informação é apresentada num ecrã com todo o detalhe da viagem</li> </ol>
<b>Exceções:</b>	<ul style="list-style-type: none"> <li>• Nos pontos 1 e 4, o ecrã não é carregado, o sistema retorna uma mensagem de erro</li> <li>• No ponto 3, a aplicação não consegue aceder à base de dados, o sistema retorna uma mensagem de erro</li> </ul>

### **Caso de Uso Ver Mensagens**

Os diversos atores podem trocar mensagens entre si, para esclarecer dúvidas sobre o evento ou informar de alguma situação relacionada com o mesmo. As notificações a reportar problemas nos eventos também são enviadas por mensagem - **Tabela 22**.

Optou-se por designar de conversa a um conjunto de mensagens para um mesmo utilizador.

**Tabela 22** - Caso de uso Ver Mensagens

<b>Titulo:</b>	<i>Ver Mensagens</i>
<b>Ator Primário:</b>	Utilizador, organizador ou revisor
<b>Stakeholders:</b>	Utilizador, organizador ou revisor
<b>Pré-Condição:</b>	O ator está no seu <i>dashboard</i>
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro
<b>Sucesso Garantido:</b>	O ator vê as todas as conversas ou apenas a pretendida
<b>Trigger:</b>	O ator clica sobre as notificações de mensagens ou clica na opção de mensagens no seu <i>dashboard</i>
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>1. O ator encontra-se no <i>dashboard</i></li> <li>2. O ator acede a funcionalidade de mensagens através de duas hipóteses: <ol style="list-style-type: none"> <li>a. O ator clica no ícone de mensagens</li> </ol> </li> </ol>

<b>Exceções:</b>	<ol style="list-style-type: none"> <li>b. O ator clica no ícone de notificações</li> <li>3. A aplicação acede à base de dados e retorna a informação pretendida</li> <li>4. As conversas são apresentadas no ecrã</li> <li>5. O ator acede a uma das conversas que tem para visualizar</li> <li>6. A aplicação acede à base de dados e retorna as mensagens da conversa pretendida</li> <li>7. É apresentado um ecrã com as mensagens</li> </ol>
	<ul style="list-style-type: none"> <li>• Nos pontos 1, 2, 4, 5 e 7, o ecrã não é carregado, o sistema retorna uma mensagem de erro</li> <li>• Nos pontos 3 e 6, a aplicação não consegue aceder à base de dados, o sistema retorna uma mensagem de erro</li> <li>• Nos pontos 3, 4, 6 e 7, a aplicação tem um erro no processamento, o sistema retorna uma mensagem de erro</li> </ul>

### **Caso de Uso Conversar**

Posicionado no ecrã de mensagens, o ator pode enviar mensagens a outros utilizadores. Esta ação foi apelidada como caso de uso **conversar - Tabela 23**.

**Tabela 23** - Caso de uso Conversar

<b>Titulo:</b>	<i>Conversar</i>
<b>Ator Primário:</b>	Utilizador, organizador ou revisor
<b>Stakeholders:</b>	Utilizador, organizador ou revisor
<b>Pré-Condição:</b>	O ator está no ecrã de mensagens O ator está posicionado no ecrã do detalhe de uma conversa
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro ao ator
<b>Sucesso Garantido:</b>	O ator envia e recebe mensagens
<b>Trigger:</b>	O ator carrega sobre uma nova mensagem ou escolhe uma das conversas já iniciadas
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>1. O ator inicia uma conversa             <ol style="list-style-type: none"> <li>a. Seleciona uma das conversas existentes</li> <li>b. Clica no ícone nova conversa</li> </ol> </li> <li>2. A aplicação acede à base de dados e retorna as mensagens da respetiva conversa</li> <li>3. As mensagens são apresentadas no ecrã</li> <li>4. O ator escreve o pretendido na caixa de texto disponibilizada no ecrã</li> <li>5. O ator clica no botão de enviar</li> <li>6. A aplicação guarda a mensagem na base de dados, ficando associada à respetiva conversa</li> <li>7. O ecrã é atualizado e a mensagem enviada passa a ser mostrada na conversa</li> <li>8. O outro ator receberá a mensagem assim que entrar na sua aplicação</li> </ol>
<b>Exceções:</b>	<ul style="list-style-type: none"> <li>• Nos pontos 1, 3 e 7, o ecrã não é carregado, o sistema retorna uma mensagem de erro</li> <li>• Nos pontos 2, 5 e 8, a aplicação não consegue aceder à base de</li> </ul>

- dados, o sistema retorna uma mensagem de erro
- Nos pontos 5, 6 e 8, a aplicação tem um erro no processamento, o sistema retorna uma mensagem de erro

### **Caso de Uso Procurar Utilizadores Seguidos**

*Guia.me* oferece a possibilidade de seguir outros utilizadores e ver eventos nos quais esses utilizadores estão interessados, na zona onde o utilizador da aplicação se encontra - **Tabela 24.**

**Tabela 24** - Caso de uso procurar utilizadores seguidos

<b>Titulo:</b>	<i>Procurar utilizador seguidos</i>
<b>Ator Primário:</b>	Utilizador
<b>Stakeholders:</b>	Utilizador
<b>Pré-Condição:</b>	O utilizador está no seu <i>dashboard</i>
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro
<b>Sucesso Garantido:</b>	Todos os outros utilizadores seguidos pelo utilizador são apresentados no ecrã
<b>Trigger:</b>	O utilizador carrega na opção procurar
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>O utilizador clica no botão de procurar do <i>dashboard</i></li> <li>Acede ao ecrã procurar</li> <li>A aplicação acede à base de dados e retorna os utilizadores seguidos do utilizador que estejam em eventos perto</li> <li>É apresentado um ecrã com todos esses utilizadores</li> </ol>
<b>Exceções:</b>	<ul style="list-style-type: none"> <li>Nos pontos 1, 2 e 4, o ecrã não é carregado, o sistema retorna uma mensagem de erro</li> <li>No ponto 3, a aplicação não consegue aceder à base de dados, o sistema retorna uma mensagem de erro</li> </ul>

### **Caso de Uso Procurar Utilizadores Com os Mesmos Interesses**

Ainda na opção anterior, é possível deslizar para a esquerda e ter acesso a utilizadores que vão aos mesmos eventos que o utilizador - **Tabela 25.**

**Tabela 25** - Caso de uso procurar utilizadores com os mesmos interesses

<b>Titulo:</b>	<i>Procurar utilizadores com os mesmos interesses</i>
<b>Ator Primário:</b>	Utilizador
<b>Stakeholders:</b>	Utilizador
<b>Pré-Condição:</b>	O utilizador está no ecrã procurar
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro
<b>Sucesso Garantido:</b>	O utilizador vê os utilizadores que têm interesse nos mesmos eventos
<b>Trigger:</b>	O utilizador desliza para a esquerda no ecrã procurar
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>O utilizador desliza para a esquerda no ecrã procurar</li> <li>A aplicação acede à base de dados e retorna os utilizadores com interesses semelhantes ao utilizador</li> <li>É apresentado um ecrã com todos esses utilizadores</li> </ol>

<b>Exceções:</b>	<ul style="list-style-type: none"> <li>Nos pontos 1 e 3, o ecrã não é carregado e o sistema retorna uma mensagem de erro</li> <li>No ponto 2, a aplicação não consegue aceder à base de dados, o sistema retorna uma mensagem de erro</li> </ul>
------------------	--

### **Caso de Uso Consultar Perfil**

Os utilizadores podem consultar o seu perfil através da opção de perfil no seu *dashboard* - **Tabela 26.**

**Tabela 26** - Caso de uso Consultar perfil

<b>Titulo:</b>	<i>Consultar perfil</i>
<b>Ator Primário:</b>	Utilizador, organizador ou revisor
<b>Stakeholders:</b>	Utilizador, organizador ou revisor
<b>Pré-Condição:</b>	O utilizador está no ecrã de <i>homepage</i>
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro ao ator
<b>Sucesso Garantido:</b>	O sistema apresenta o perfil do ator
<b>Trigger:</b>	O ator clica na opção perfil do seu <i>dashboard</i> .
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>O ator encontra-se no <i>dashboard</i></li> <li>O ator seleciona a opção de perfil</li> <li>A aplicação acede à base de dados e retorna os detalhes do perfil do ator</li> <li>É apresentado o ecrã de detalhe do perfil do ator</li> </ol>
<b>Exceções:</b>	<ul style="list-style-type: none"> <li>Nos pontos 1 e 4, o ecrã não é carregado, o sistema retorna uma mensagem de erro</li> <li>No ponto 3, a aplicação não consegue aceder à base de dados, o sistema retorna uma mensagem de erro</li> </ul>

### **Caso de Uso Ver Perfil de Outro Utilizador**

Os atores podem ainda consultar o perfil de outros atores da aplicação de forma a saber informação básica sobre os mesmos ou seguir utilizadores - **Tabela 27.**

**Tabela 27** - Caso de uso ver perfil de outro utilizador

<b>Titulo:</b>	<i>Ver perfil de outro utilizador</i>
<b>Ator Primário:</b>	Utilizador, organizador ou revisor
<b>Stakeholders:</b>	Utilizador, organizador ou revisor
<b>Pré-Condição:</b>	O ator está presente nas mensagens, no detalhe do evento ou na lista de utilizadores com os mesmos interesses
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro ao ator
<b>Sucesso Garantido:</b>	O sistema apresenta o perfil do ator selecionado
<b>Trigger:</b>	O ator clica no nome de outro ator
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>O ator clica no nome de outro ator</li> <li>A aplicação valida se o perfil é publico</li> <li>A aplicação acede à base de dados e retorna os detalhes do perfil do ator</li> </ol>

<b>Exceções:</b>	4. É apresentado o ecrã de detalhe do perfil do ator, com a informação recolhida no ponto anterior
	• Nos pontos 1 e 4, o ecrã não é carregado, o sistema retorna uma mensagem de erro
	• Nos pontos 2 e 3, a aplicação não consegue aceder à base de dados, o sistema retorna uma mensagem de erro

### **Caso de Uso Seguir Utilizador**

Os utilizadores podem **seguir outros utilizadores** com o objetivo de perceberem quais são os eventos onde vão e os seus interesses - **Tabela 28**.

**Tabela 28** - Caso de uso seguir utilizador

<b>Titulo:</b>	<i>Seguir utilizador</i>
<b>Ator Primário:</b>	Utilizador
<b>Stakeholders:</b>	Utilizador
<b>Pré-Condição:</b>	O utilizador está no ecrã perfil de outro utilizador
	O utilizador está na lista de utilizadores que vão aos mesmos eventos
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro
<b>Sucesso Garantido:</b>	O utilizador começa a seguir o outro utilizador
<b>Trigger:</b>	O utilizador clica na opção de seguir
<b>Cenário Principal de Sucesso:</b>	1. O utilizador clica em seguir
	2. A aplicação acede à base de dados e cria a relação entre utilizadores
<b>Exceções:</b>	3. É apresentada uma mensagem de confirmação
	• Nos pontos 1 e 3, o ecrã não é carregado, o sistema retorna uma mensagem de erro
	• No ponto 2, a aplicação não consegue aceder à base de dados, o sistema retorna uma mensagem de erro

### **Caso de Uso Alterar Foto**

O caso de uso **alterar foto** pode ser aplicado a diversas situações na aplicação (**Tabela 29**):

- O ator quer alterar a sua foto de perfil;
- O organizador quer colocar ou alterar a foto do evento;
- O utilizador quer adicionar mais fotos ao evento.

**Tabela 29** - Caso de Uso Alterar Foto

<b>Titulo:</b>	<i>Alterar foto</i>
<b>Ator Primário:</b>	Utilizador, organizador ou revisor
<b>Stakeholders:</b>	Utilizador, organizador ou revisor
<b>Pré-Condição:</b>	O ator está no ecrã de perfil
	O utilizador esta no ecrã detalhes do evento O organizador está a criar ou editar um evento
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro

<b>Sucesso Garantido:</b>	A foto é alterada
<b>Trigger:</b>	O ator clica sobre a sua foto de perfil no ecrã de perfil O utilizador clica no “+” no ecrã detalhe de evento O organizador clica na foto do evento
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>1. O ator clica em alterar foto: <ol style="list-style-type: none"> <li>a. O ator clica na sua foto de perfil</li> <li>b. O utilizador clica no botão “+” no detalhe do evento</li> <li>c. O organizador clica na foto do evento</li> </ol> </li> <li>2. É apresentado um <i>popup</i> com a opção de <i>upload</i></li> <li>3. O ator clica no botão de <i>upload</i></li> <li>4. O ator escolhe uma nova foto <ol style="list-style-type: none"> <li>a. Tira uma foto com a câmara do dispositivo móvel</li> <li>b. Seleciona uma foto guardada na galeria</li> </ol> </li> <li>5. A aplicação guarda a nova foto na base de dados</li> <li>6. A foto é atualizada no ecrã correspondente</li> </ol>
<b>Exceções:</b>	<ul style="list-style-type: none"> <li>• Nos pontos 1, 2 e 6, o ecrã não é carregado, o sistema retorna uma mensagem de erro</li> <li>• No ponto 5, a aplicação não consegue aceder à base de dados, o sistema retorna uma mensagem de erro</li> <li>• No pontos 2, 4, 5 e 6, a aplicação tem um erro no processamento, o sistema retorna uma mensagem de erro</li> </ul>

### **Caso de Uso Alterar Dados (Perfil)**

Houve a necessidade de criar a funcionalidade de alterar dados de perfil para permitir aos diversos atores alterarem as suas preferências e dados pessoais - **Tabela 30**.

**Tabela 30** - Caso de uso Alterar Dados do Perfil

<b>Titulo:</b>	<i>Alterar dados (perfil)</i>
<b>Ator Primário:</b>	Utilizador, organizador ou revisor
<b>Stakeholders:</b>	Utilizador, organizador ou revisor
<b>Pré-Condição:</b>	O ator está no ecrã de detalhes de perfil
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro ao ator
<b>Sucesso Garantido:</b>	Os dados do perfil são alterados
<b>Trigger:</b>	O ator seleciona um dos seus dados para alterar
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>1. O ator está no ecrã perfil</li> <li>2. O ator edita dados do seu perfil</li> <li>3. A aplicação acede à base de dados e atualiza os respetivos dados</li> <li>4. O ecrã é atualizado com os novos dados</li> </ol>
<b>Exceções:</b>	<ul style="list-style-type: none"> <li>• Nos pontos 1 e 4, o ecrã não é carregado, o sistema retorna uma mensagem de erro</li> <li>• N ponto 3 a aplicação não consegue aceder à base de dados, o sistema retorna uma mensagem de erro</li> </ul>

### **Caso de Uso Criar Evento/Publicar**

Os eventos existem no **Guia.me** porque foram criados previamente pelo ator organizador; para isso o organizador tem a funcionalidade **criar evento/publicar** - **Tabela 31**.

**Tabela 31** - Caso de Uso Criar Evento/Publicar

<b>Titulo:</b>	<i>Criar evento/publicar</i>
<b>Ator Primário:</b>	Organizador
<b>Stakeholders:</b>	Organizador
<b>Pré-Condição:</b>	O organizador está no ecrã lista de eventos
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro
<b>Sucesso Garantido:</b>	O organizador publica o evento
<b>Trigger:</b>	O organizador seleciona a opção adicionar evento
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>1. O organizador clica no botão “+”</li> <li>2. O ecrã de “<b>destaque do evento</b>” é carregado</li> <li>3. O organizador preenche o formulário (nome e descrição do evento)</li> <li>4. Clica no ícone de foto</li> <li>5. No <i>popup</i> escolhe a foto que deseja atribuir ao evento</li> <li>6. Clica no botão seguinte</li> <li>7. A aplicação acede à base de dados e cria o novo evento</li> <li>8. É apresentado o ecrã de “<b>detalhes</b>” do evento ao organizador</li> <li>9. O organizador preenche o formulário com detalhes sobre o evento</li> <li>10. O organizador clica em seguinte</li> <li>11. A aplicação atualiza o evento com a nova informação</li> <li>12. É apresentado o terceiro ecrã de configuração do evento, “<b>localização</b>”</li> <li>13. O organizador seleciona a localização que pretende dar ao evento</li> <li>14. Clica em publicar</li> <li>15. A aplicação atualiza o evento na base de dados com a nova informação</li> <li>16. O evento fica com o estado pendente de aprovação do revisor na base de dados</li> <li>17. É apresentada uma mensagem ao organizador, de que o evento está pendente de aprovação</li> </ol>
<b>Exceções:</b>	<ul style="list-style-type: none"> <li>• Nos pontos 2, 7, 12 e 17, o ecrã não é carregado, o sistema retorna uma mensagem de erro</li> <li>• Nos pontos 7, 11, 15 e 16, a aplicação não consegue aceder à base de dados, o sistema retorna uma mensagem de erro</li> <li>• Nos pontos 6, 10 e 14, a aplicação tem um erro no processamento, o sistema retorna uma mensagem de erro</li> </ul>

### **Caso de Uso Ver Eventos Pendentes (Rejeitados ou com Problemas)**

Depois de publicar um evento, o utilizador pode ver os eventos que ainda estão pendentes (rejeitados ou outros problemas); sendo a lista de eventos semelhante às referidas anteriormente - **Tabela 32**.

**Tabela 32 - Caso de Uso Eventos Rejeitados**

<b>Titulo:</b>	<i>Ver eventos pendentes (rejeitados ou com problemas)</i>
<b>Ator Primário:</b>	Organizador
<b>Stakeholders:</b>	Organizador
<b>Pré-Condição:</b>	Ser organizador
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro
<b>Sucesso Garantido:</b>	O organizador vê todos os eventos que foram rejeitados
<b>Trigger:</b>	Ao entrar na aplicação, se o organizador for o responsável por algum evento que tenha sido rejeitado, o mesmo aparece no ecrã inicial. Outra forma de chegar a este ecrã é utilizando a opção respetiva no <i>dashboard</i>
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>1. O organizador entra na aplicação</li> <li>2. A aplicação acede à base de dados e valida se existem eventos pendentes do respetivo organizador</li> <li>3. Os eventos pendentes são apresentados no ecrã lista de eventos</li> </ol>
<b>Exceções:</b>	<ul style="list-style-type: none"> <li>• Nos pontos 1 e 3, o ecrã não é carregado, o sistema retorna uma mensagem de erro</li> <li>• No ponto 2, a aplicação não consegue aceder à base de dados, o sistema retorna uma mensagem de erro e o organizador é redirecionado diretamente para o seu <i>dashboard</i></li> </ul>

**Caso de Uso Ver Eventos Favoritos (Populares)**

Na listagem de eventos pendentes, o utilizador poderá deslizar o ecrã para a esquerda e tem acesso aos seus **eventos mais populares ou favoritos - Tabela 33**.

**Tabela 33 - Caso de Uso Eventos Favorito (Populares)**

<b>Titulo:</b>	<i>Ver eventos favoritos (populares)</i>
<b>Ator Primário:</b>	Organizador
<b>Stakeholders:</b>	Organizador
<b>Pré-Condição:</b>	Ser organizador
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro
<b>Sucesso Garantido:</b>	O organizador vê os seus eventos que são populares
<b>Trigger:</b>	Ao entrar na aplicação, o organizador é responsável por eventos populares Através do seu <i>dashboard</i> seleciona a opção eventos
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>1. O organizador entra na aplicação</li> <li>2. A aplicação acede à base de dados e valida se existem eventos populares para aquele organizador</li> <li>3. Os eventos populares são apresentados no ecrã lista de eventos</li> </ol>

**Exceções:**

- Nos pontos 1 e 3, o ecrã não é carregado, o sistema retorna uma mensagem de erro
- No ponto 2, a aplicação não consegue aceder à base de dados, o sistema retorna uma mensagem de erro e o organizador é redirecionado diretamente para a *homepage*

Os casos de uso representados nas tabelas anteriores (**Tabela 32** e **Tabela 33**) dão acesso aos casos de uso facultados nas listas de eventos, como o procurar, filtrar e ver detalhes de evento, representados nas **Tabela 8**, **Tabela 9**, e **Tabela 10**, respetivamente.

**Caso de Uso Publicar Evento**

**Publicar o evento** é uma funcionalidade que permite ao utilizador partilhar o seu evento nas redes sociais - **Tabela 34**. O evento ganha destaque adicional na aplicação, ganhando relevância durante um tempo limitado.

**Tabela 34** - Caso de Uso Publicar Evento

<b>Titulo:</b>	<i>Publicar evento</i>
<b>Ator Primário:</b>	Organizador
<b>Stakeholders:</b>	Organizador
<b>Pré-Condição:</b>	O organizador estar no ecrã detalhes do evento
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro
<b>Sucesso Garantido:</b>	O evento é publicitado
<b>Trigger:</b>	O organizador clica no ícone de publicar evento
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>1. O organizador escolhe um evento da lista e acede aos seus detalhes</li> <li>2. O organizador clica no botão de publicar</li> <li>3. É apresentado um <i>popup</i> de confirmação</li> <li>4. O evento é atualizado na base de dados</li> <li>5. O evento é publicitado e apresentada uma mensagem de sucesso ao utilizador</li> </ol>
<b>Exceções:</b>	<ul style="list-style-type: none"> <li>• Nos pontos 1, 3 e 5, o ecrã não é carregado, o sistema retorna uma mensagem de erro</li> <li>• No ponto 4, a aplicação não consegue aceder à base de dados, o sistema retorna uma mensagem de erro</li> </ul>

**Caso de Uso Editar Evento**

O organizador do evento poderá editar o evento por si criado. Para tal, foi criada a funcionalidade de **editar evento** (**Tabela 35**). Esta funcionalidade permite:

- Alterar detalhes;
- Alterar localização;
- Alterar ou adicionar imagens;
- Apagar evento.

Cada ponto acima referido culminará na funcionalidade **publicar**, que enviará novamente o evento para aprovação por parte de um revisor.

**Tabela 35 - Caso de uso Editar evento**

<b>Titulo:</b>	<i>Editar evento</i>
<b>Ator Primário:</b>	Organizador
<b>Stakeholders:</b>	Organizador
<b>Pré-Condição:</b>	O organizador está no ecrã de detalhes do evento
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro
<b>Sucesso Garantido:</b>	O evento do organizador é editado
<b>Trigger:</b>	O organizador no ecrã de detalhe do evento edita o evento
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>1. O organizador encontra-se no detalhe de um dos seus eventos</li> <li>2. O organizador seleciona a opção de editar evento</li> <li>3. O evento é carregado em modo de edição, no mesmo formato <i>wizzard</i> de 3 fases usado na criação</li> <li>4. O organizador pode editar o <b>destaque do evento</b> e quando terminar clica em seguinte</li> <li>5. A aplicação acede à base de dados e atualiza o evento</li> <li>6. A base de dados retorna a informação sobre os detalhes do evento</li> <li>7. É apresentada a segunda fase de edição do evento no ecrã de edição <b>detalhe do evento</b></li> <li>8. O organizador altera o pretendido e clica em seguinte</li> <li>9. A aplicação acede à base de dados e faz a atualização do evento</li> <li>10. A base de dados retorna a informação sobre a localização do evento</li> <li>11. É apresentada a informação na terceira e última fase de edição, no ecrã de edição <b>localização do evento</b></li> <li>12. O organizador faz as alterações pretendidas e clica em atualizar</li> <li>13. A aplicação acede à base de dados, faz atualização da localização do evento e coloca novamente o evento no estado pendente de aprovação para um dos revisores</li> <li>14. É apresentado o ecrã de lista de eventos novamente</li> </ol>
<b>Exceções:</b>	<ul style="list-style-type: none"> <li>• Nos pontos 1, 3, 7, 11 e 14, o ecrã não é carregado, o sistema retorna uma mensagem de erro</li> <li>• Nos pontos 3, 5, 6, 9, 10 e 13, a aplicação não consegue aceder à base de dados, o sistema retorna uma mensagem de erro</li> <li>• Nos pontos 3, 4, 8, 12 e 13, a aplicação tem um erro no processamento, o sistema retorna uma mensagem de erro</li> </ul>

**Caso de Uso Ver Relatório**

Para que fosse possível os organizadores e revisores terem uma ideia de como está a progredir o seu trabalho na aplicação, foi criada a funcionalidade **Ver relatório - Tabela 36**.

**Tabela 36 - Caso de Uso Ver Relatório**

<b>Titulo:</b>	<i>Ver relatório</i>
<b>Ator Primário:</b>	Organizador ou revisor
<b>Stakeholders:</b>	Organizador ou revisor
<b>Pré-Condição:</b>	O ator está no seu <i>dashboard</i>
<b>Garantia</b>	O sistema retorna uma mensagem de erro ao ator

<b>Mínima:</b>	
<b>Sucesso</b>	O ator vê o ecrã com os relatórios direcionados para si
<b>Garantido:</b>	
<b>Trigger:</b>	O ator clica na opção de relatórios do seu <i>dashboard</i>
<b>Cenário</b>	1. O organizador está no <i>dashboard</i> e clica na opção relatórios
<b>Principal de</b>	2. A aplicação acede à base de dados e procura os dados sobre os eventos do ator
<b>Sucesso:</b>	3. A informação é recolhida, processada e são gerados os relatórios 4. Os relatórios são apresentados no ecrã relatório
<b>Exceções:</b>	<ul style="list-style-type: none"> <li>• Nos pontos 1 e 4, o ecrã não é carregado, o sistema retorna uma mensagem de erro ao ator</li> <li>• No ponto 2, a aplicação não consegue aceder à base de dados, o sistema retorna uma mensagem de erro ao ator</li> <li>• No ponto 3, a aplicação tem um erro no processamento, o sistema retorna uma mensagem de erro ao ator</li> </ul>

### **Caso de Uso Ver Eventos Pendentes de Aprovação**

No momento em que os eventos são publicados, estes são atribuídos a um revisor que irá validar se estes devem ou não ser disponibilizados para todos os utilizadores. Enquanto aguardam essa validação, os eventos ficam numa lista de **eventos pendentes** (aprovar ou rejeitar) - **Tabela 37**.

**Tabela 37** - Caso de uso ver eventos pendentes de aprovação

<b>Título:</b>	<i>Ver eventos pendentes de aprovação</i>
<b>Ator Primário:</b>	Revisor
<b>Stakeholders:</b>	Revisor
<b>Pré-Condição:</b>	Ser Revisor.
<b>Garantia</b>	O sistema retorna uma mensagem de erro
<b>Mínima:</b>	
<b>Sucesso</b>	O revisor tem acesso a uma lista de eventos que necessitam de aprovação
<b>Garantido:</b>	
<b>Trigger:</b>	Este ecrã esta disponível quando um o revisor acede a aplicação ou através do seu <i>dashboard</i>
<b>Cenário</b>	1. O revisor inicia a aplicação
<b>Principal de</b>	2. A aplicação acede à base de dados e valida se existem eventos pendentes de aprovação para ele
<b>Sucesso:</b>	3. É apresentado o ecrã com os eventos pendentes
<b>Exceções:</b>	<ul style="list-style-type: none"> <li>• No ponto 3, o ecrã não é carregado, o sistema retorna uma mensagem de erro</li> <li>• No ponto 2, a aplicação não consegue aceder à base de dados, o sistema retorna uma mensagem de erro</li> <li>• No ponto 2, a aplicação tem um erro no processamento, o sistema retorna uma mensagem de erro</li> <li>• No ponto 2, caso não existam eventos pendentes, o caso de uso termina e o revisor é direcionado o seu <i>dashboard</i></li> </ul>

O caso de uso representado na tabela anterior (**Tabela 37**) dá acesso aos casos de uso facultados nas listas de eventos, como o procurar, filtrar e ver detalhes de evento, representados nas **Tabela 8**, **Tabela 9**, e **Tabela 10**, respetivamente.

### **Caso de Uso Aprovar Evento**

Na lista de eventos pendentes, o revisor poderá validar os eventos e, caso cumpram com todas as regras, poderão ser aprovado - **Tabela 38**.

**Tabela 38** - Caso de Uso Aceitar Evento

<b>Titulo:</b>	<i>Aprovar evento</i>
<b>Ator Primário:</b>	Revisor
<b>Stakeholders:</b>	Revisor
<b>Pré-Condição:</b>	O revisor está no ecrã da lista de eventos pendentes
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro
<b>Sucesso Garantido:</b>	O revisor aceita o evento e o mesmo fica disponível para todos os atores da aplicação
<b>Trigger:</b>	O revisor clica no botão de aceitar o evento
<b>Cenário Principal de Sucesso:</b>	<ol style="list-style-type: none"> <li>1. O revisor esta no ecrã dos eventos pendentes, onde tem uma lista</li> <li>2. O revisor aceita o evento: <ol style="list-style-type: none"> <li>a. Acede ao caso de uso detalhes do evento e depois aceita o evento</li> <li>b. Clica no botão aceitar na lista de eventos pendentes</li> </ol> </li> <li>3. A aplicação acede à base de dados e muda o estado do evento para aprovado</li> <li>4. O evento é colocado visível para todos os atores</li> </ol>
<b>Exceções:</b>	<ul style="list-style-type: none"> <li>• No ponto 2a, o ecrã não é carregado, o sistema retorna uma mensagem de erro</li> <li>• Nos pontos 2a e 3, a aplicação não consegue aceder à base de dados, o sistema retorna uma mensagem de erro</li> <li>• Nos pontos 2b e 4, a aplicação tem um erro no processamento, o sistema retorna uma mensagem de erro</li> </ul>

### **Caso de Uso Rejeitar Evento**

Caso determinado evento pendente não esteja em conformidade com as regras, o revisor **rejeita o evento**, adicionando um mensagem com o motivo de rejeição - **Tabela 39**.

**Tabela 39** - Caso de Uso Rejeitar Evento

<b>Titulo:</b>	<i>Rejeitar evento</i>
<b>Ator Primário:</b>	Revisor
<b>Stakeholders:</b>	Revisor, organizador
<b>Pré-Condição:</b>	O revisor está no ecrã da lista de eventos pendentes
<b>Garantia Mínima:</b>	O sistema retorna uma mensagem de erro
<b>Sucesso Garantido:</b>	O revisor rejeita o evento juntamente com uma observação pelo qual foi rejeitado
<b>Trigger:</b>	O revisor clica no botão de rejeitar o evento

**Cenário  
Principal de  
Sucesso:**

1. O revisor esta no ecrã dos eventos pendentes, onde tem uma lista
2. O revisor rejeita o evento:
  - a. Acede ao caso de uso **detalhes do evento** e depois rejeita o evento
  - b. Clica no botão rejeitar na lista de eventos
3. A aplicação acede à base de dados e muda o estado do evento para rejeitado
4. É apresentado um *popup* ao revisor onde deve escrever o motivo de rejeição
5. A aplicação atualiza a base de dados
6. O evento fica disponível para o organizador editar e agir em conformidade

**Exceções:**

- Nos pontos 2<sup>a</sup> e 4, o ecrã não é carregado, o sistema retorna uma mensagem de erro
- Nos pontos 2a, 3 e 5, a aplicação não consegue aceder à base de dados, o sistema retorna uma mensagem de erro
- Nos pontos 2b, 4 e 6, a aplicação tem um erro no processamento, o sistema retorna uma mensagem de erro

### 4.2.3. Diagramas de Robustez

Os diagramas de robustez, utilizados principalmente na metodologia ICONIX, têm como objetivo suavizar a passagem da análise para o desenho. Com estes diagramas, é possível fazer a verificação da compatibilidade entre os casos de uso e a colocação em prática dos mesmos, ou seja, valida a possibilidade de realizar ou não o pretendido. Além de amenizar a dificuldade da passagem dos requisitos para o projeto, estes diagramas podem levar à descoberta de novas classes que não foram percebidas na primeira versão do Modelo de Domínio (Jacobson, 1992).

Foram elaborados diagramas de robustez correspondentes aos casos de uso detalhados no capítulo anterior e todos eles seguiram as seguintes regras (DELFIM, 2017);

- Atores interagem apenas com objetos de fronteira (*boundary*);
- Objetos de fronteira podem interagir com atores e controladores;
- Objetos de entidade interagem apenas com controladores;
- Controladores interagem com qualquer outro tipo de objeto e também com outros controladores, mas nunca com atores.

Na **Figura 58**, **Figura 59**, **Figura 60**, **Figura 61** e **Figura 62**, estão representados cinco diagramas de robustez, representativos dos casos de uso Ver Sugestões, Filtrar, Ver Viagens, Ver Detalhe do Evento e Rejeitar, respetivamente. Os restantes diagramas de robustez poderão ser consultados nos anexos (III - Diagramas de Robustez).

**Diagrama de Robustez Ver Sugestões**

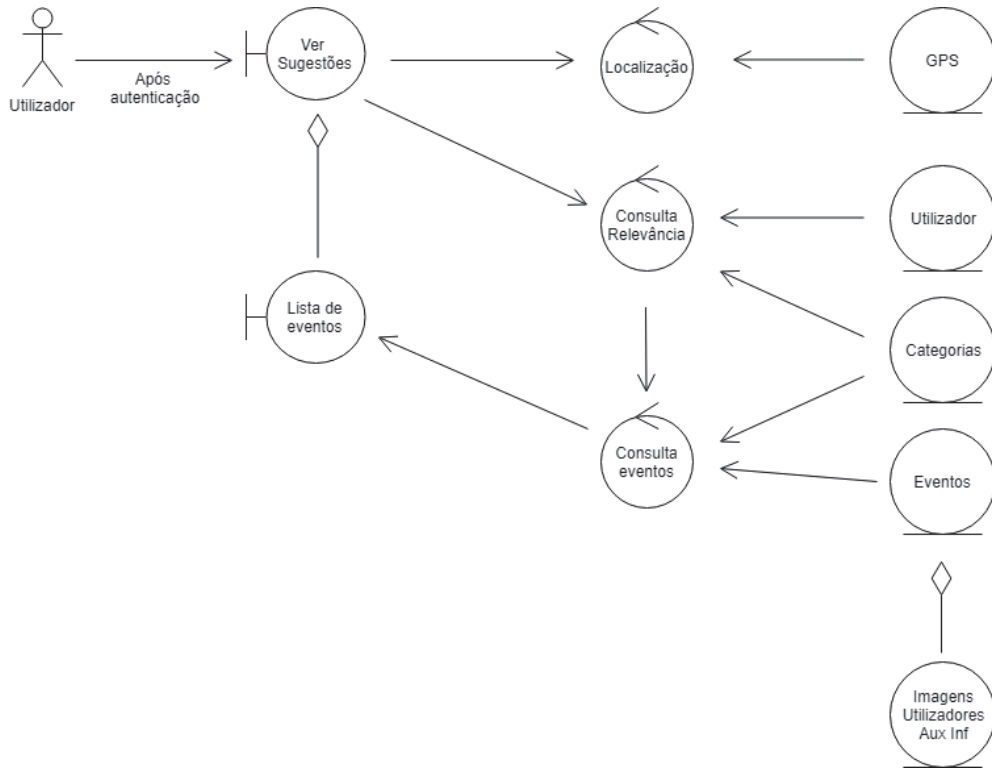


Figura 58 - Diagrama de robustez - ver sugestões

**Diagrama de Robustez Filtrar**

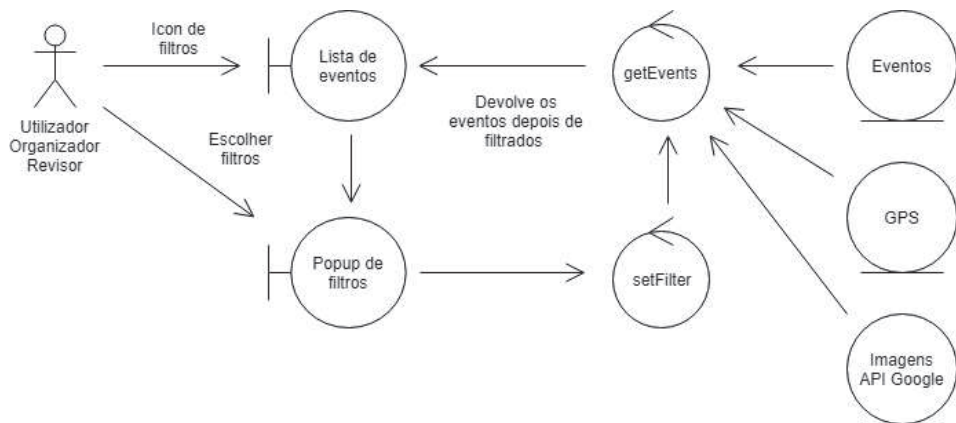


Figura 59 - Diagrama de robustez - filtrar

**Diagrama de Robustez Ver Viagens**

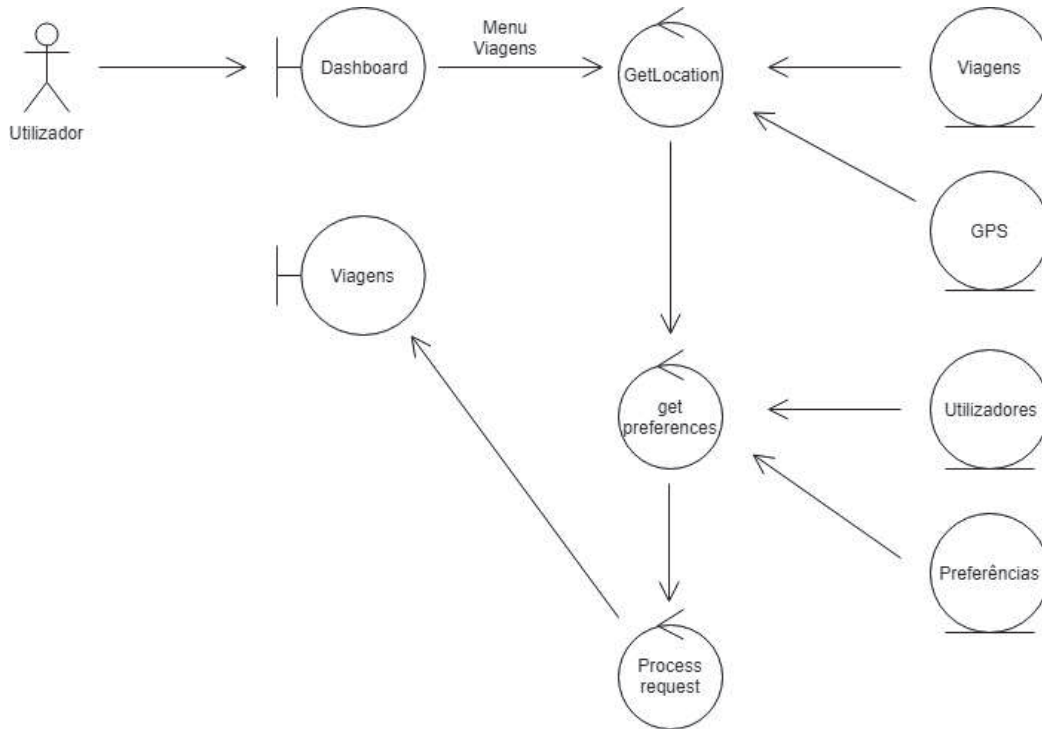


Figura 60 - Diagrama de robustez - ver viagens

**Diagrama de Robustez Ver Detalhe do Evento**

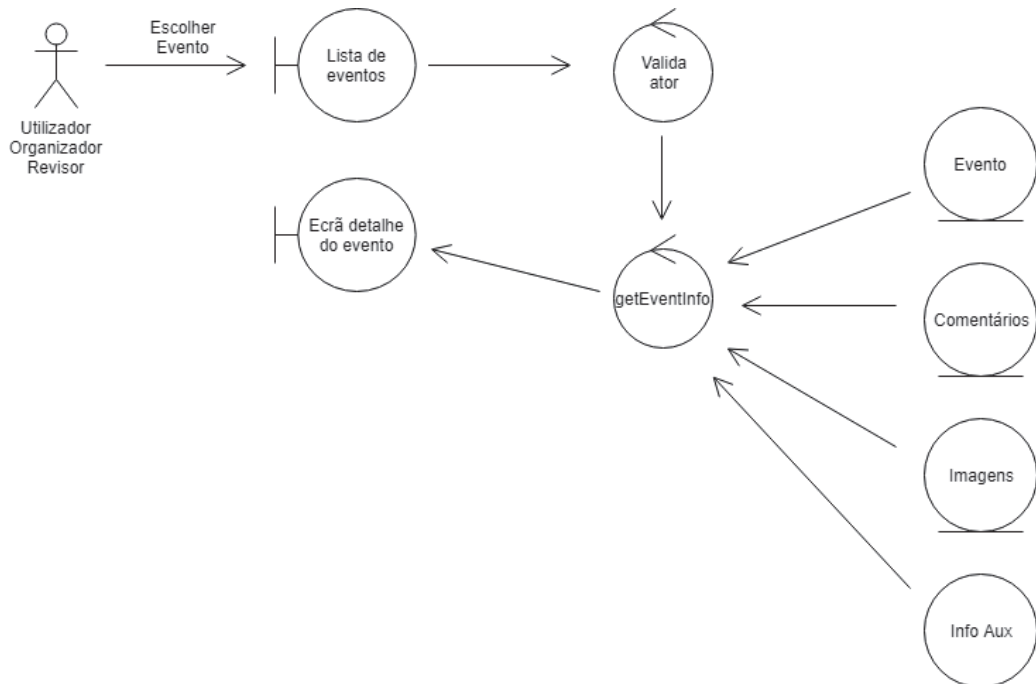


Figura 61 - Diagrama de robustez - ver detalhe do evento

**Diagrama de Robustez Rejeitar**

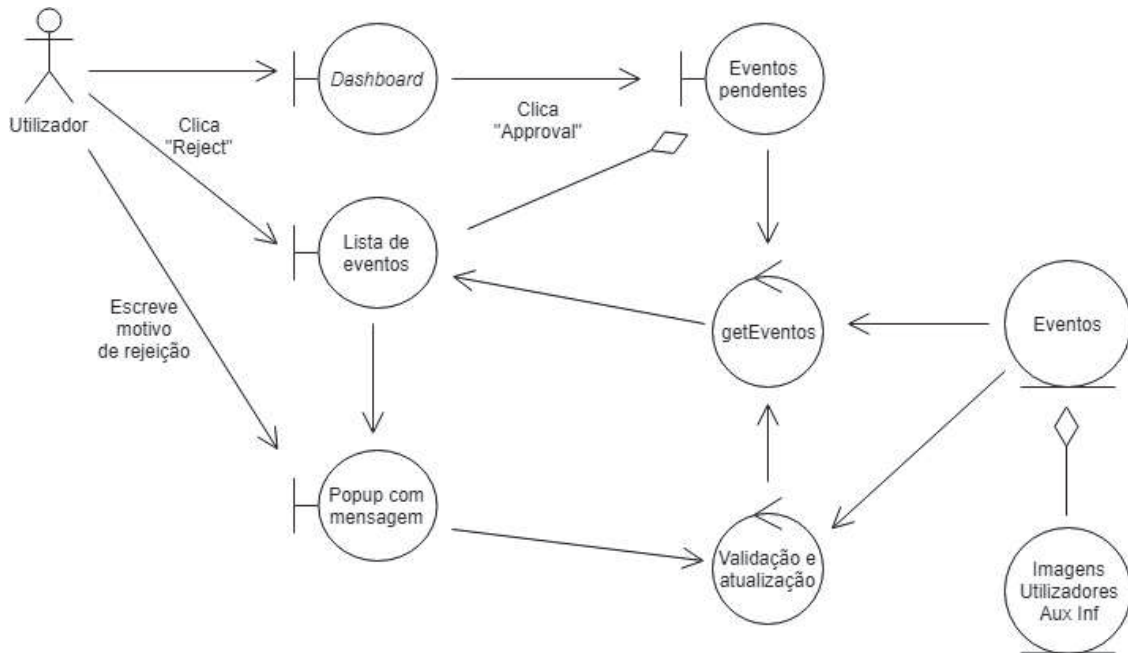


Figura 62 - Diagrama de robustez - Rejeitar

**4.2.4. Atualização do Modelo de Domínio**

Nesta seção apresenta-se a atualização do modelo de domínio identificado na primeira fase, com base no resultado do trabalho desenvolvido neste subcapítulo. Os atores aliados aos seus casos de uso e posteriormente ao desenvolvimento dos diagramas de robustez permitiram a descoberta de novos atributos e classes, representados na **Figura 63**.

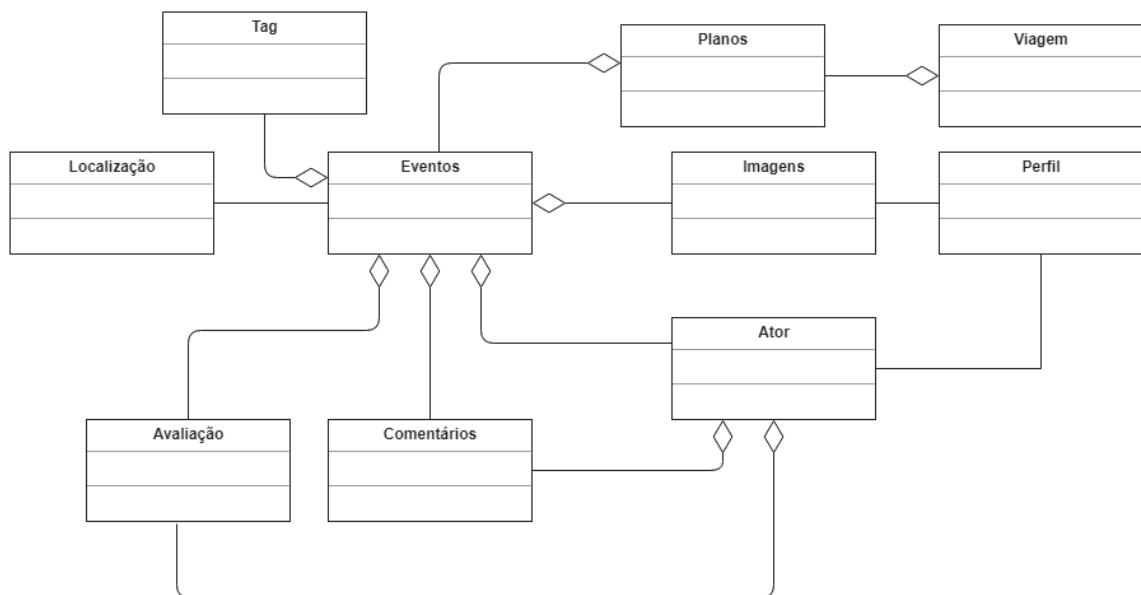


Figura 63 - Modelo de Domínio atualizado

## 4.3. Projeto

Neste subcapítulo, é abordada a modelação de base de dados, que consiste no desenho do modelo E/R e o modelo relacional que dele resultará. São detalhadas as entidades e atributos de cada uma dessas entidades e é identificada a troca de mensagens entre objetos, através dos diagramas de sequência. Por fim, é atualizado o modelo de domínio, dando origem ao modelo de domínio final.

### 4.3.1. Modelação de Base de Dados

#### Modelo E/R

A construção do modelo Entidade Relacionamento (E/R) é importante para que se consiga obter posteriormente o modelo relacional de base de dados, que servirá para guardar toda a informação gerada pela aplicação. Do modelo E/R fazem parte os objetos básicos, que se denominam de entidades, e as relações entre esses mesmos objetos - **Figura 64**.

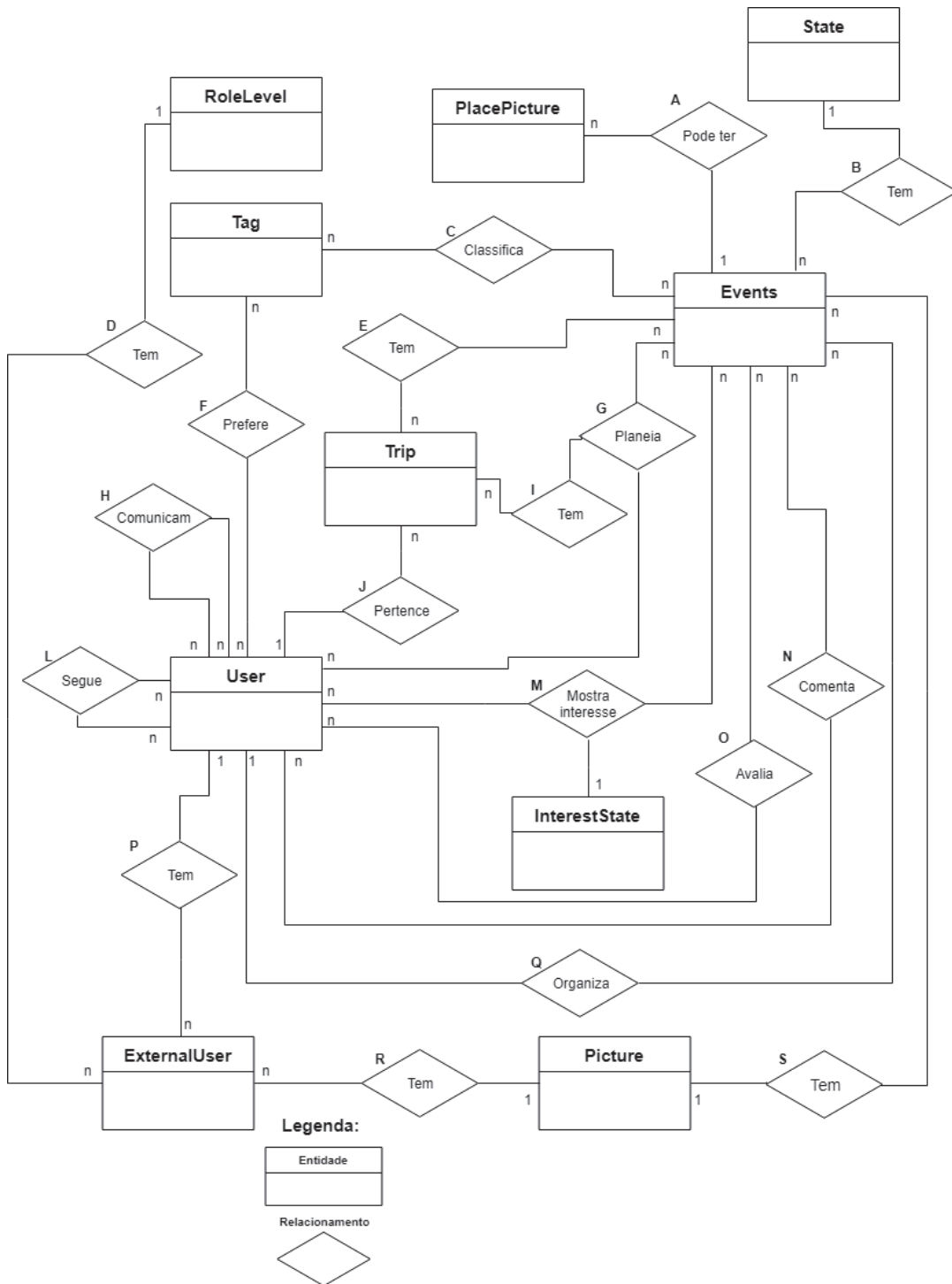


Figura 64 - Modelo E/R

**A – PlacePicture\_Events** é a relação (n para 1) entre as entidades *PlacePicture* (imagens) e *Events* (eventos); ou seja, uma imagem pertence a um evento e um evento pode ter várias imagens. Desta relação não resulta nenhuma entidade nova.

**B – State\_Events** é a relação (1 para n) entre as entidades *State* (estados de referência) e *Events* (eventos); ou seja, um estado pode pertencer a vários eventos, mas um evento só pode conter um estado. Desta relação não resulta nenhuma entidade nova.

**C - *Tag\_Events*** é a relação (n para n) entre as entidades *Tag* (categoria) e *Events* (eventos); ou seja, uma categoria pode ter vários eventos e um evento pode ter várias categorias. Desta relação resulta a entidade *EventTag*.

**D - *UserRoleLevel*** é a relação (n para 1) entre as entidades *ExternalUser* (utilizador externo) e *RoleLevel* (função de referência); ou seja, uma função pode pertencer a vários utilizadores mas um utilizador só pode ter uma função. Desta relação não resulta nenhuma entidade nova.

**E - *EventTrips*** é a relação de (n para n) entre as entidades *Events* (eventos) e *Trips* (viagens); ou seja, um evento pode estar em n viagens e uma viagem pode conter n eventos. Desta relação resulta a entidade *EventTrips*.

**F - *UserTag*** é a relação (n para n) entre as entidades *User* (utilizador) e *Tag* (categoria); ou seja, um utilizador pode ter preferência por várias categorias e uma categoria pode ser da preferência de vários utilizadores. Desta relação resulta uma nova entidade, *Preferences*.

**G - *UserEvents\_Plan*** é a relação (n para n) das entidades *User* (utilizador) e *Events* (eventos); ou seja, um utilizador pode ter no seu plano ir a vários eventos e um evento pode estar no plano de vários utilizadores. Desta relação resulta a entidade *Plans*.

**H - *Messages*** é a relação (n para n) entre a entidade *User* (utilizador) com ela própria; ou seja, um utilizador pode enviar mensagens para vários utilizadores e também pode receber mensagens de vários utilizadores. Desta relação resulta uma nova entidade, *Messages*.

**I - *TripPlans*** é a relação (n para 1) entre as entidades *Trip* (viagens) e *Plans* (planos); ou seja, uma viagem pode ter vários planos mas um plano só pode pertencer a uma viagem. Desta relação não resulta nenhuma nova entidade.

**J - *UserTrip*** é a relação (1 para n) entre as entidades *User* (utilizador) e *Trip* (viagens); ou seja, um utilizador pode ter várias viagens mas uma viagem só pode pertencer a um utilizador. Desta relação resultará a entidade *UserTrip*.

**L - *Followers*** é a relação (n para n) entre a entidade *User* (utilizador) com ela própria; ou seja, resulta que o utilizador pode seguir vários utilizadores e ao mesmo tempo ser seguido por vários utilizadores.

**M - *UserEventState*** é a relação (n(n para n) para 1) entre as entidades *User* (utilizador), *Events* (eventos) e *InterestState* (estado de interesse de referência); ou seja, um utilizador pode ter interesse em n eventos e um evento pode ser do interesse de n utilizadores, sendo que a relação daí resultante só pode ter um estado e esse estado pode estar em n relações. Daqui resulta a nova entidade *Interest*.

**N - *UserEventsComments*** é a relação (n para n) entre *User* (utilizador) e *Events* (eventos); ou seja, um utilizador pode fazer n comentários em um ou n eventos, e um evento pode ter n comentários de um ou n utilizadores. Desta relação resulta a entidade *Reviews*.

**O - *UserEventsRate*** é a relação (n para n) entre *User* (Utilizadores) e *Events* (eventos); ou seja, um utilizador pode avaliar n eventos e um evento pode ter n avaliações de n utilizadores. Desta relação resulta a entidade *Rating*.

**P - *ExternalUser*** é a relação (1 para 1) entre as entidades *User* (utilizador) e *ExternalUser* (utilizador externo); ou seja, um utilizador externo pertence apenas a um utilizador e um utilizador pertence apenas a um utilizador externo. Desta relação não resulta nenhuma entidade.

**Q - *UserEvent*** é a relação (1 para n) entre as entidades *User* (utilizador) e *Events* (eventos); ou seja, um evento só pode ter um organizador mas um organizador pode ter vários eventos. Desta relação não resultam novas entidades.

**R - *UserPicture*** é a relação (n para 1) entre as entidades *ExternalUser* (utilizador externo) e *Picture* (imagens); ou seja, um utilizador só pode ter uma imagem mas uma imagem pode estar em vários utilizadores. Desta relação não resultam novas entidades.

**S - *EventPicture*** é a relação (n para 1) entre as entidades *Event* (evento) e *Picture* (imagens); ou seja, um evento só pode ter uma imagem mas uma imagem pode estar em n eventos. Desta relação não resultam novas entidades.

## Modelo Relacional

O modelo relacional, à semelhança do modelo E/R, tem como objetivo identificar as entidades do sistema. Neste modelo são representadas todas as tabelas resultantes do relacionamento com as respectivas chaves primárias e forasteiras. O modelo relacional é representado na **Figura 65**.

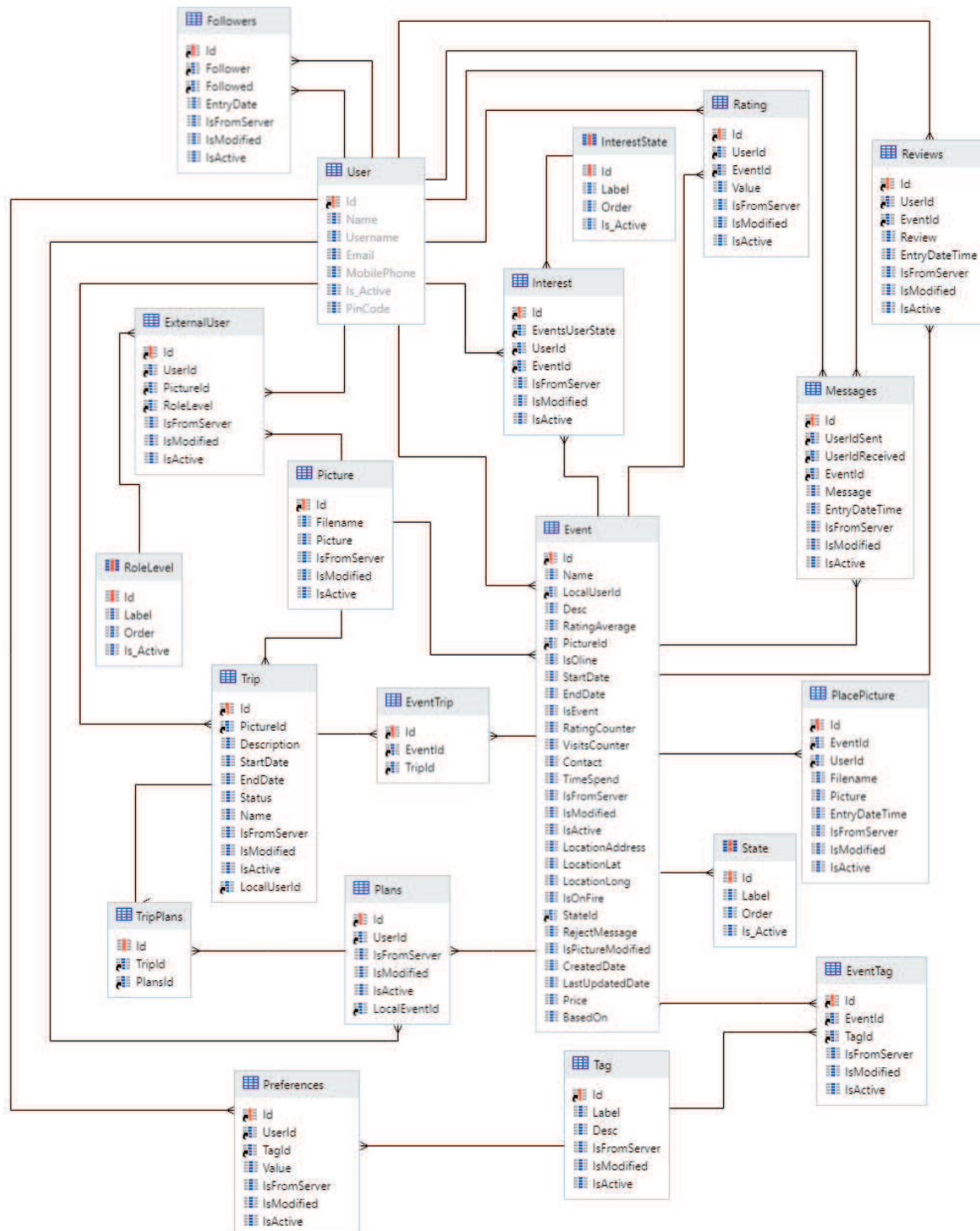


Figura 65 - Modelo relacional

### 4.3.2. Tabelas de Dados

O sistema caracteriza-se pela possibilidade de utilização de informação tanto *online* como *offline*. Para tal, existe a necessidade de utilizar o armazenamento local disponível em cada dispositivo móvel e o armazenamento no servidor ou *cloud*.

O servidor e todos os dados estão num espaço atualmente disponibilizado pela OutSystems para realização de pequenas aplicações por qualquer pessoa que pretenda experimentar a tecnologia. Todos os dados locais são guardados numa base de dados local do dispositivo móvel, depois de instalada a aplicação e feita a configuração e sincronização inicial.

Nesta fase, optou-se por desenhar modelo de dados do servidor e modelo de dados local bastante semelhantes, minimizando a complexidade na componente de sincronização entre dados nos diversos dispositivos, componente essa detalhada na secção 6.2.2.

Nesta secção são apresentadas as tabelas de dados e armazenamento necessários para o funcionamento da aplicação. Serão apresentadas quatro das vinte entidades existentes na base de dados; as restantes poderão ser consultadas em anexo I (Tabelas usadas).

#### ***Event***

Esta é a tabela onde é armazenada a informação do evento (**Tabela 40**). Nela podemos encontrar elementos básicos, como nome e descrição do evento, assim como ligações para a restante informação, por exemplo:

- **Entidade *User*** – Organizador que criou o evento;
- **Entidade *Picture*** – Imagem principal do evento.

Optou-se por guardar atributos com informação auxiliar, resultante de processos que são executados a uma determinada hora do dia, podendo ser os seguintes:

- ***RatingAverage*** – Classificação média dada pelos utilizadores ao evento;
- ***RatingCounter*** – Número de utilizadores que classificaram o evento;
- ***VisitsCounter*** – Número de utilizadores que já visitaram um evento;
- ***AttendingCounter*** – Número de utilizadores que querem ir ao evento.

Esta informação ficou nos atributos do evento para que estivesse sempre acessível na informação do evento e não mediante de cálculos executados no momento.

**Tabela 40** - Tabela de eventos

Campo	Tipo	Descrição	Observações
<b><i>Id</i></b>	<i>Long Integer</i>		Chave primária
<b><i>Name</i></b>	<i>Text</i>	Título do evento	
<b><i>Desc</i></b>	<i>Text</i>	Descrição do evento	
<b><i>RatingAverage</i></b>	<i>Decimal</i>	Classificação média do evento	
<b><i>PictureId</i></b>	<i>PictureIdentifier</i>	Identificador da imagem principal do evento	Chave externa para entidade <i>Picture</i>

<b>IsOnline</b>	<i>Boolean</i>	Se verdadeiro, o evento está disponível para os utilizadores	
<b>StartDate</b>	<i>DateTime</i>	Data de início do evento, caso se aplique	
<b>EndDate</b>	<i>DateTime</i>	Data de fim do evento, caso se aplique	
<b>TimeTableId</b>	<i>TimeTableIdentifier</i>	Identificador do calendário do evento	Chave externa para entidade <i>TimeTable</i>
<b>IsUnique</b>	<i>Boolean</i>	Evento único, sem repetição	
<b>RatingCounter</b>	<i>Integer</i>	Contador de avaliações no evento	
<b>VisitsCounter</b>	<i>Integer</i>	Contador de visitas ao evento	
<b>AttendingCounter</b>	<i>Integer</i>	Contador de pessoas com interesse no evento	
<b>Contact</b>	<i>Text</i>	Contato do organizador	
<b>TimeSpendAvg</b>	<i>Time</i>	Tempo médio gasto por utilizadores no evento	
<b>LocationAddress</b>	<i>Text</i>	Localização do evento	
<b>LocationLat</b>	<i>Decimal</i>	Latitude do evento	
<b>LocationLong</b>	<i>Decimal</i>	Longitude do evento	
<b>Trending</b>	<i>Boolean</i>	Se verdadeiro, o evento tem muito interesse para os utilizadores	
<b>UserId</b>	<i>UserIdentifier</i>	Identificador do Organizador	Chave externa para a entidade <i>User</i>
<b>StateId</b>	<i>StateIdentifier</i>	Identificador do estado	Chave externa para a entidade <i>State</i>
<b>RejectMessage</b>	<i>Text</i>	Mensagem de rejeição do evento	
<b>CreateDate</b>	<i>Date</i>	Data de criação do evento	
<b>LastUpdateDate</b>	<i>DateTime</i>	Data da última atualização do evento	
<b>IsPictureModified</b>	<i>Boolean</i>	Se verdadeiro, a imagem foi alterada desde da última sincronização com o servidor	

<b><i>IsFromServer</i></b>	<i>Boolean</i>	Identifica se o evento está no servidor	Atributo para sincronização com o servidor
<b><i>IsModified</i></b>	<i>Boolean</i>	Identifica se o evento foi modificado	Atributo para sincronização com o servidor
<b><i>IsActive</i></b>	<i>Boolean</i>	Identifica se o evento esta ativo	Atributo para sincronização com o servidor

### **Picture**

A tabela de imagens (**Figura 41**) guarda toda a informação referente a imagens principais, por exemplo:

- Foto de perfil de um ator;
- Imagem principal de um evento;
- Imagem principal de uma viagem.

São consideradas imagens principais todas aquelas que têm uma relação 1 para n, ou seja, a mesma imagem pode estar em n eventos, mas o evento só pode ter uma imagem principal.

**Tabela 41** - Tabela de imagens

<b>Campo</b>	<b>Tipo</b>	<b>Descrição</b>	<b>Observações</b>
<b><i>Id</i></b>	<i>Long Integer</i>	Identificador da imagem	Chave primária
<b><i>Filename</i></b>	<i>Text</i>	Nome do ficheiro	
<b><i>Picture</i></b>	<i>BinaryData</i>	Imagem em informação binária	
<b><i>IsFromServer</i></b>	<i>Boolean</i>	Se verdadeiro, a imagem existe no servidor	Atributo para sincronização com o servidor
<b><i>IsModified</i></b>	<i>Boolean</i>	Se verdadeiro, a imagem foi alterada localmente	Atributo para sincronização com o servidor
<b><i>IsActive</i></b>	<i>Boolean</i>	Se verdadeiro, a imagem não foi removida localmente	Atributo para sincronização com o servidor

### **Interest**

A **Tabela 42** apresenta a entidade *Interest*, que relaciona o evento com os utilizadores, atribuindo um estado a essa mesma relação. Os estados da relação podem ser:

- **Interested:** o utilizador tem interesse em ir;
- **Attending:** o utilizador vai ao evento;
- **Attended:** o utilizador já foi ao evento;
- **NotAttending:** o utilizador não tem qualquer interesse em ir ao evento.

Tabela 42 - Tabela de relação utilizador vs ponto de interesse

Campo	Tipo	Descrição	Observações
<b><i>Id</i></b>	<i>Long Integer</i>	Identificador da relação	Chave primária
<b><i>UserId</i></b>	<i>UserIdentifier</i>	Identificador do utilizador	Chave externa para a entidade <i>User</i>
<b><i>EventId</i></b>	<i>EventIdentifier</i>	Identificador do evento	Chave externa para a entidade <i>Event</i>
<b><i>StateId</i></b>	<i>StateIdentifier</i>	Identificador do estado	Chave externa para a entidade <i>State</i>
<b><i>IsFromServer</i></b>	<i>Boolean</i>	Se verdadeiro, a relação existe no servidor	Atributo para sincronização com o servidor
<b><i>IsModified</i></b>	<i>Boolean</i>	Se verdadeiro, a relação foi alterada localmente	Atributo para sincronização com o servidor
<b><i>IsActive</i></b>	<i>Boolean</i>	Se verdadeiro, a relação não foi removida localmente	Atributo para sincronização com o servidor

### Preferences

**Guia.me** calcula a preferência do utilizador baseada nas ações deste dentro da aplicação, sendo que para isso utiliza as categorias dos eventos. A entidade *preferences* resulta da relação entre categorias e utilizadores - **Tabela 43**.

Tabela 43 - Tabela de preferências

Campo	Tipo	Descrição	Observações
<b><i>Id</i></b>	<i>Long Integer</i>	Identificador da relação	Chave primária
<b><i>UserId</i></b>	<i>UserIdentifier</i>	Identificador do utilizador	Chave externa para a entidade <i>User</i>
<b><i>TagId</i></b>	<i>TagIdentifier</i>	Identificador da <i>Tag</i>	Chave externa para a entidade <i>Tag</i>
<b><i>Value</i></b>		Valor da categoria para o utilizador	

<b><i>IsFromServer</i></b>	<i>Boolean</i>	Se verdadeiro, a relação existe no servidor	Atributo para sincronização com o servidor
<b><i>IsModified</i></b>	<i>Boolean</i>	Se verdadeiro, a relação foi alterada localmente	Atributo para sincronização com o servidor
<b><i>IsActive</i></b>	<i>Boolean</i>	Se verdadeiro, a relação não foi removida localmente	Atributo para sincronização com o servidor

### 4.3.3. Tabelas de Dados de Referência

Tabelas de dados de referência ou tabelas estáticas (como denominadas pela OutSystems), consiste num conjunto de valores, como uma enumeração armazenada na base de dados. A estrutura é definida pelos atributos da entidade estática, enquanto os dados são os seus registos (Static Entities - OutSystems, 2017).

Todas as tabelas estáticas criadas no âmbito de **Guia.me** têm a mesma estrutura, diferindo apenas os dados nelas configurados - **Tabela 44**.

**Tabela 44** - Estrutura das tabelas de dados de referência

<b>Campo</b>	<b>Tipo</b>	<b>Descrição</b>	<b>Observações</b>
<b><i>Id</i></b>	<i>Long Integer</i>	Identificador da relação	Chave primária
<b><i>Label</i></b>		Identificador do evento	Legenda configurada
<b><i>Order</i></b>		Identificador da <i>Tag</i>	Numeração para descrever a ordenação
<b><i>IsActive</i></b>		Se verdadeiro, a relação existe no servidor	Boleano que define se está ativa ou não

São apresentados nas tabelas seguintes alguns dos exemplos configurados em quatro entidades de referência.

#### **ErrorMessage**

As mensagens de erro usadas na maioria dos casos de exceção do sistema são próprias da plataforma OutSystems. Houve a necessidade de criar até ao momento 4 novas mensagens, sendo elas descritas na **Tabela 45** e armazenadas na entidade *ErrorMessage*.

Tabela 45 - Mensagens de erro

Label	Descrição	Observações
<i>InputEmpty</i>	Caixa de edição de texto obrigatória encontra-se vazia	
<i>NoResults</i>	Não obteve resultados para o pedido efetuado	
<i>NoGPS</i>	Não existe conectividade GPS	
<i>Suspended</i>	O valor pretendido está suspenso	

**InterestState**

Quando o utilizador demonstra algum tipo de interesse sobre um evento, essa informação resulta numa relação entre as entidades *user* e *event*, que ficará registada na entidade *interest*. *InterestState* é a entidade que contém os valores de referência para classificar essa mesma relação - **Tabela 46**.

Tabela 46 - Dados de referencia, Evento utilizador

Label	Descrição	Observações
<i>Interested</i>	Utilizador tem interesse em ir ao evento	
<i>Attending</i>	Utilizador quer ir ao evento	
<i>Attended</i>	Utilizador já foi ao evento	
<i>NotAttending</i>	Utilizador não tem interesse no evento	

**RoleLevel**

Com a exceção do visitante, todos os restantes atores contêm uma função a eles associada. Os valores representativos dessas funções estão configurados na entidade *RoleLevel* - **Tabela 47**.

Tabela 47 - Dados de referencia, utilizador função

Label	Descrição	Observações
<i>User</i>	Função de utilizador	
<i>Organizer</i>	Função de organizador	
<i>Reviewer</i>	Função de revisor	

**State**

A entidade *State* (**Tabela 48**) armazena os estados configurados para a entidade *event*, ou seja, serve para definir um estado do evento, podendo ser utilizada para outras entidades, se necessário.

Estes valores são utilizados em muitas particularidades da aplicação, onde alguns exemplos são as listas de eventos dos diversos atores.

**Tabela 48** - Dados de referencia, estado

Label	Descrição	Observações
<i>Approved</i>	Aprovado	
<i>Pending</i>	Pendente	
<i>Rejected</i>	Rejeitado	
<i>InProgress</i>	Em progresso	
<i>All</i>	Todos	

**4.3.4. Diagramas de Sequência**

Os diagramas de sequência têm como objetivo clarificar o funcionamento de cada funcionalidade da aplicação, sendo possível visualizar a ordem de ações e as camadas que estão a ser consultadas. São também a última componente realizada para atualizar o modelo de domínio com o resultado final.

Os diagramas de sequência foram realizados para todos os casos de uso identificados anteriormente. Nesta seção serão detalhados os seguintes:

- Ver sugestões;
- Filtrar;
- Participar no evento;
- Ver viagens;
- Criar, editar publicar evento;
- Rejeitar.

Os restantes diagramas poderão ser consultados em anexo - II, Diagramas de Sequência.

Para a representação dos diagramas de sequência optou-se por agrupar algumas componentes, de modo a tornar menos complexo e mais agradável de visualizar, utilizando alguns termos como:

- **Camada de Negócio**

Todas as funções de lógica de negócio, invocações de consulta realizadas alto nível, ou todas as funções que atualmente já pertencem a esta camada na arquitetura da aplicação, foram agrupadas neste bloco.

- **Segmento “Entidade”**

Nas aplicações móveis, o acesso à base de dados local é praticamente instantâneo e, por isso, são realizados mais acessos do que uma aplicação *web*; uma simples consulta pode significar um acesso a pelo menos 25% das entidades de **Guia.me**. Optou-se, então, por agrupar as várias entidades num segmento, utilizando o “segmento eventos”, onde estão todas as entidades relacionadas aos eventos que determinado ecrã necessita.

### **Diagrama de Sequência Ver Sugestões**

A lista de eventos sugeridos pressupõe que o utilizador acede à aplicação **Guia.me** e esta desencadeia uma sequência de ações, sem haver ação do utilizador durante esse processo, que resultará numa lista de eventos relevantes apresentada ao utilizador - **Figura 66**.

Neste diagrama, os objetos a considerar para a camada de negócio são:

- *CheckUser*;
- *SuggestedEvents*;
- *RelevanceCalc*;
- *GetTagValues*;
- *GetEvents*;
- *GetLocation*;
- *GetUsersByEvent*;
- *GetDistanceByEvent*.

Os objetos a considerar para a base de dados (segmento eventos) são:

- *Tag*;
- *Preferences*;
- *Events*;
- *Interest*;
- *User*;
- *Picture*.

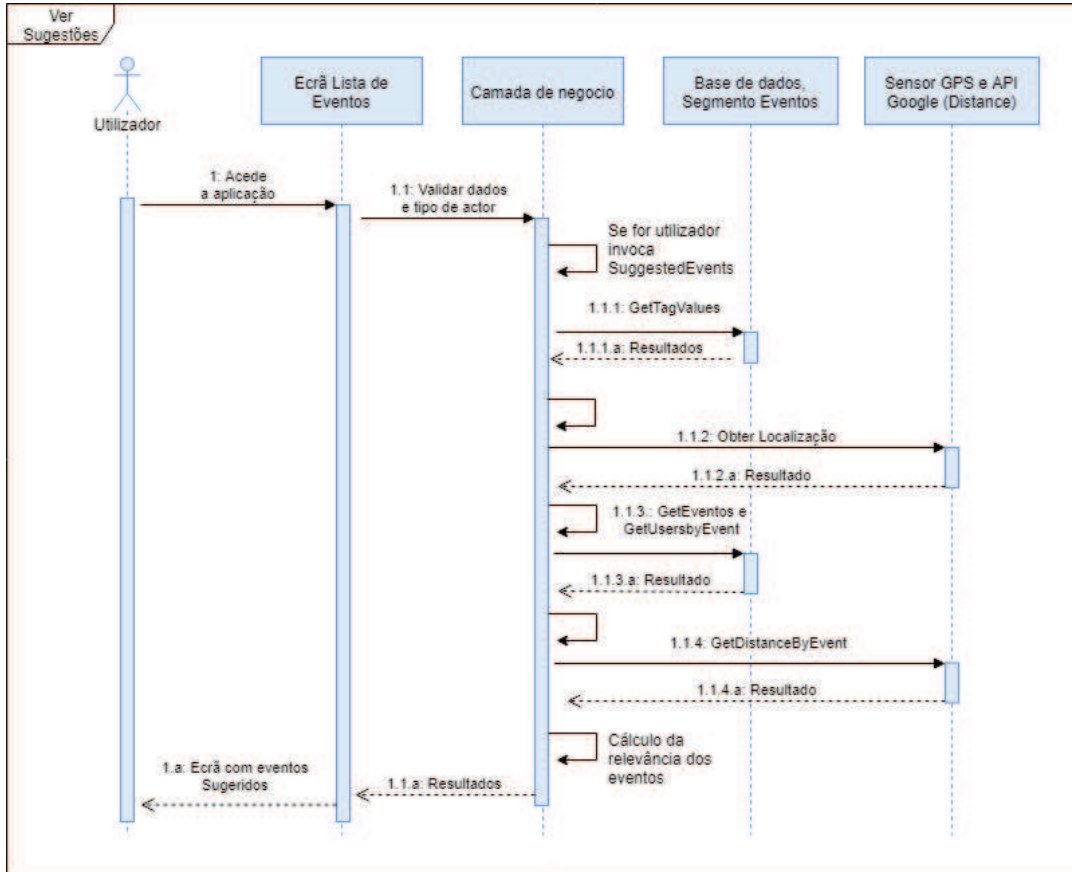


Figura 66 - Diagrama de sequência - Ver sugestões

### Diagrama de Sequência Filtrar

Todas as listas de eventos existentes na aplicação **Guia.me** contêm a funcionalidade **filtrar**, podendo ser as ações do utilizador diversas durante o processo (pode seleccionar vários tipos de filtros e com várias configurações) - **Figura 67**.

Neste diagrama não há acesso a base de dados. Quanto à camada de negócio, são usados os objetos *CheckUser* e *GetDistanceByEvent*.

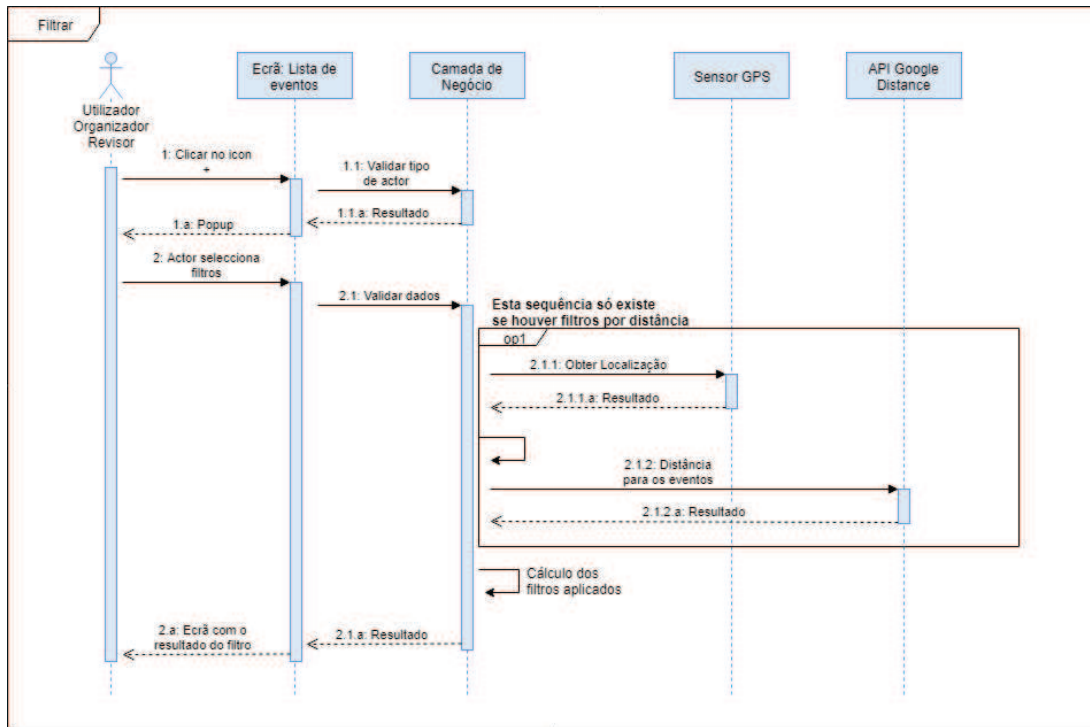


Figura 67 - Diagrama de sequência - Filtrar

### **Diagrama de Sequência Participar no Evento**

Esta funcionalidade só é realizada durante a execução de um plano por parte do utilizador, ou seja, o utilizador está a seguir o seu percurso disponibilizado pela aplicação **Guia.me** e a sequência de eventos da **Figura 68** colocam o utilizador “a participar no evento”.

Neste diagrama, os objetos da camada de negócio considerados são:

- *GetLocation;*
- *CheckEventsLocation.*

Os objetos da base de dados (segmento eventos e planos) considerados são:

- *Events;*
- *Preferences;*
- *Tag;*
- *User;*
- *Plans;*
- *Interest.*

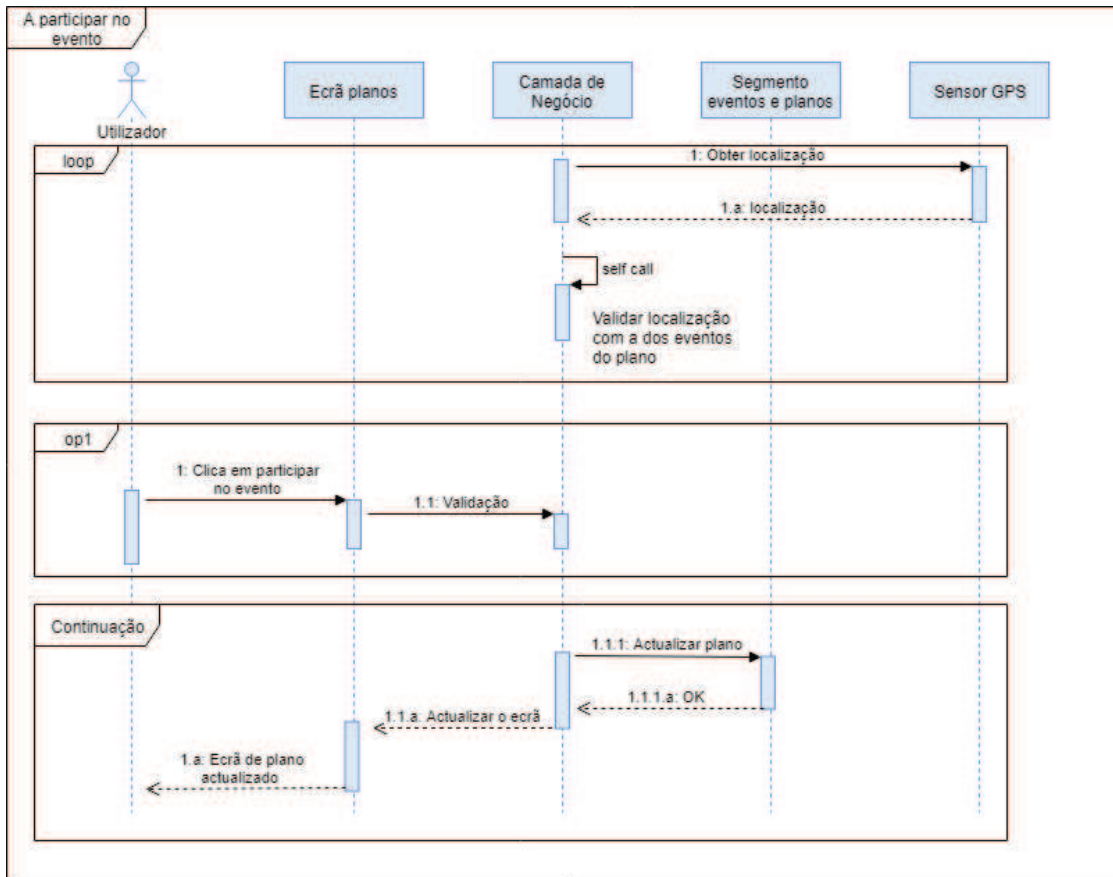


Figura 68 - Diagrama de sequência - A participar no evento

### **Diagrama de Sequência Ver Viagens**

O utilizador deseja ver as suas viagens e, para isso, utiliza a opção de viagens. A aplicação necessita de todas as viagens do utilizador, saber a sua localização atual e carregar toda a informação - **Figura 69**.

Neste diagrama, considerou-se na camada de negócios os seguintes objetos:

- *GetTrips;*
- *GetLocation;*
- *GetLocationInfo.*

Na base de dados (segmento de viagens), foram considerados os seguintes objetos:

- *Trips;*
- *Picture;*
- *User.*

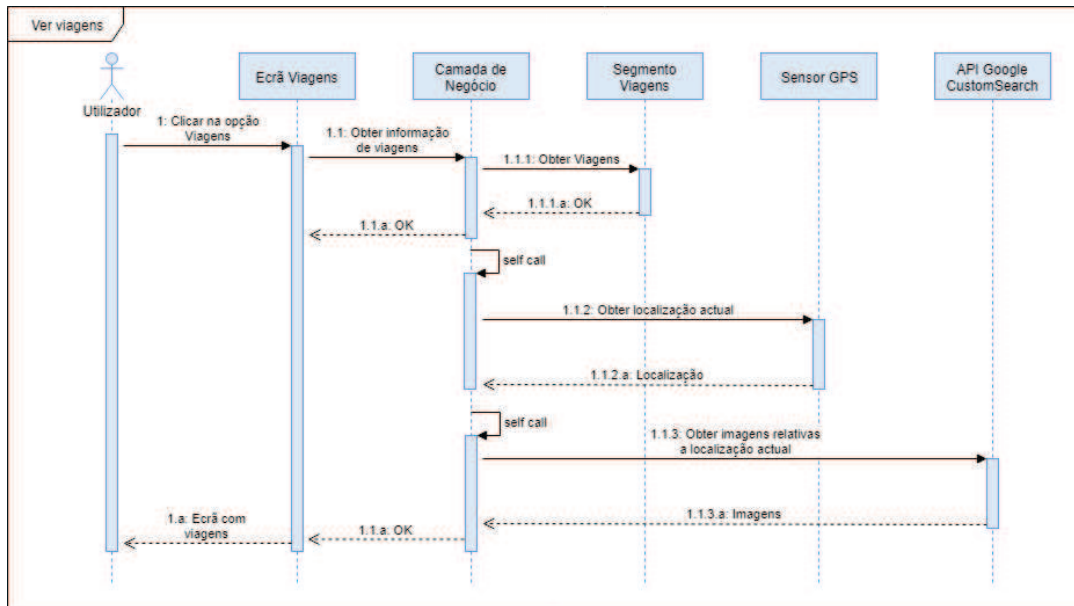


Figura 69 - Diagrama de sequência - Ver viagens

### **Diagrama de Sequência Criar, Editar e Publicar Evento**

Criar e editar um evento são funcionalidades do organizador, que terminam obrigatoriamente com o publicar do mesmo. Criar e editar têm um diagrama de sequência idêntico, diferenciando na parte técnica (um faz atualizações e o outro inserção). As duas funcionalidades apresentam um diagrama de sequência mais complexo devido às elevadas interações do organizador com o sistema e a complexidade da própria funcionalidade - **Figura 70**.

Neste diagrama, considerou-se os seguintes objetos na camada de negócio:

- *CheckUser;*
- *CreateEvent;*
- *GetLocation;*
- *GetLocationInfo;*
- *Publish.*

Na base de dados (segmento eventos), foram considerados os seguintes objetos:

- *Events;*
- *Picture;*
- *Tag;*
- *User.*

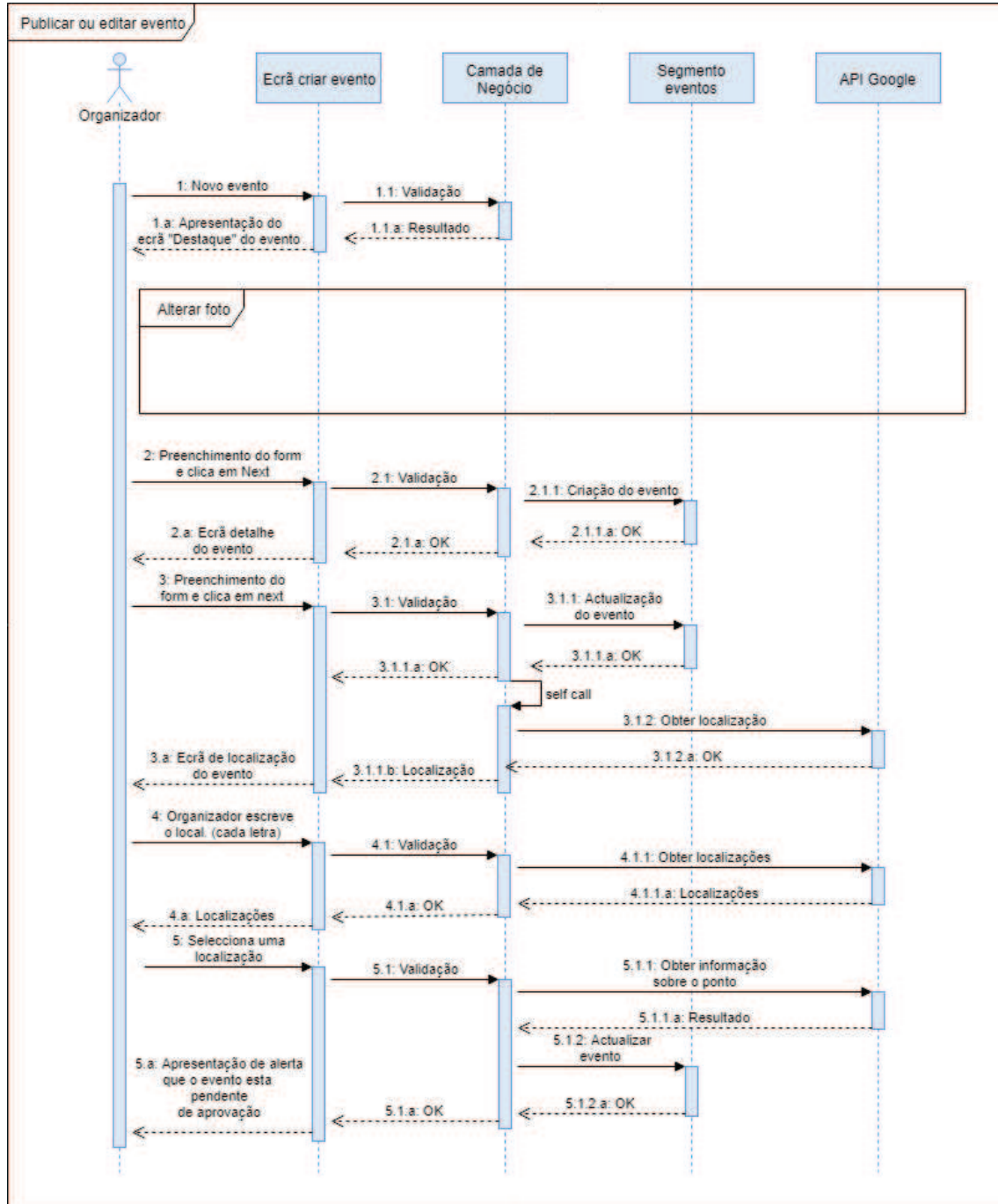


Figura 70 - Diagrama de sequência - Publicar ou editar evento

### Diagrama de Sequência Rejeitar

O seguinte diagrama de sequência exemplifica a sequência de ações e troca de mensagens entre o sistema para o caso de uso **rejeitar** - **Figura 71**.

Neste diagrama, considerou-se na camada de negócios os seguintes objetos:

- *SetEventPending;*
- *SendEventToUser.*

Na base de dados (segmento eventos), foram considerados os seguintes objetos:

- *Events;*
- *User.*

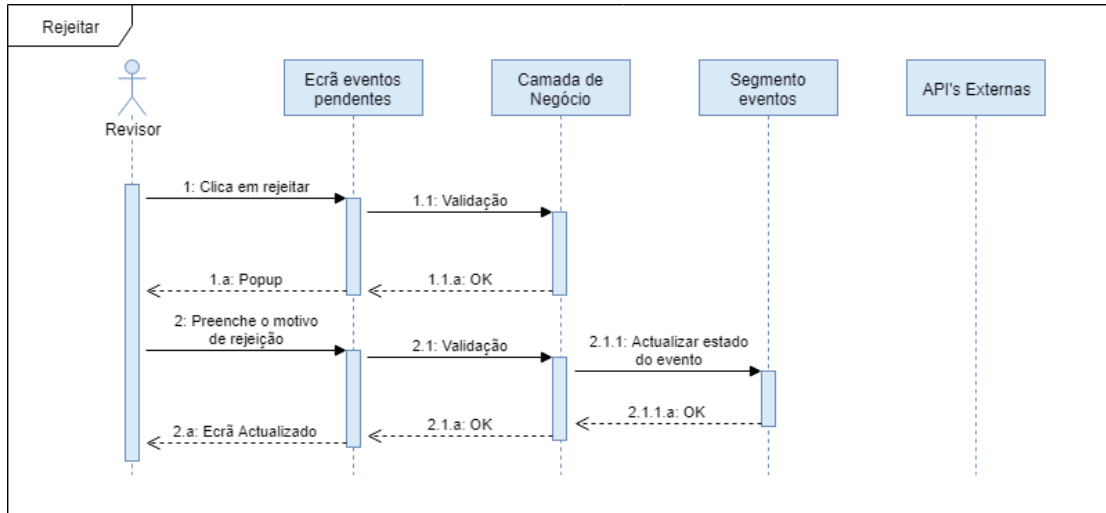


Figura 71 - Diagrama de sequência - Rejeitar

#### 4.3.5. Atualização do Modelo de Domínio - Diagrama de Classes

Após a elaboração da modelação de dados e elaboração dos diagramas de sequência, foi possível proceder a uma última atualização do modelo de domínio, que deu origem ao diagrama de classes representado na **Figura 72**.

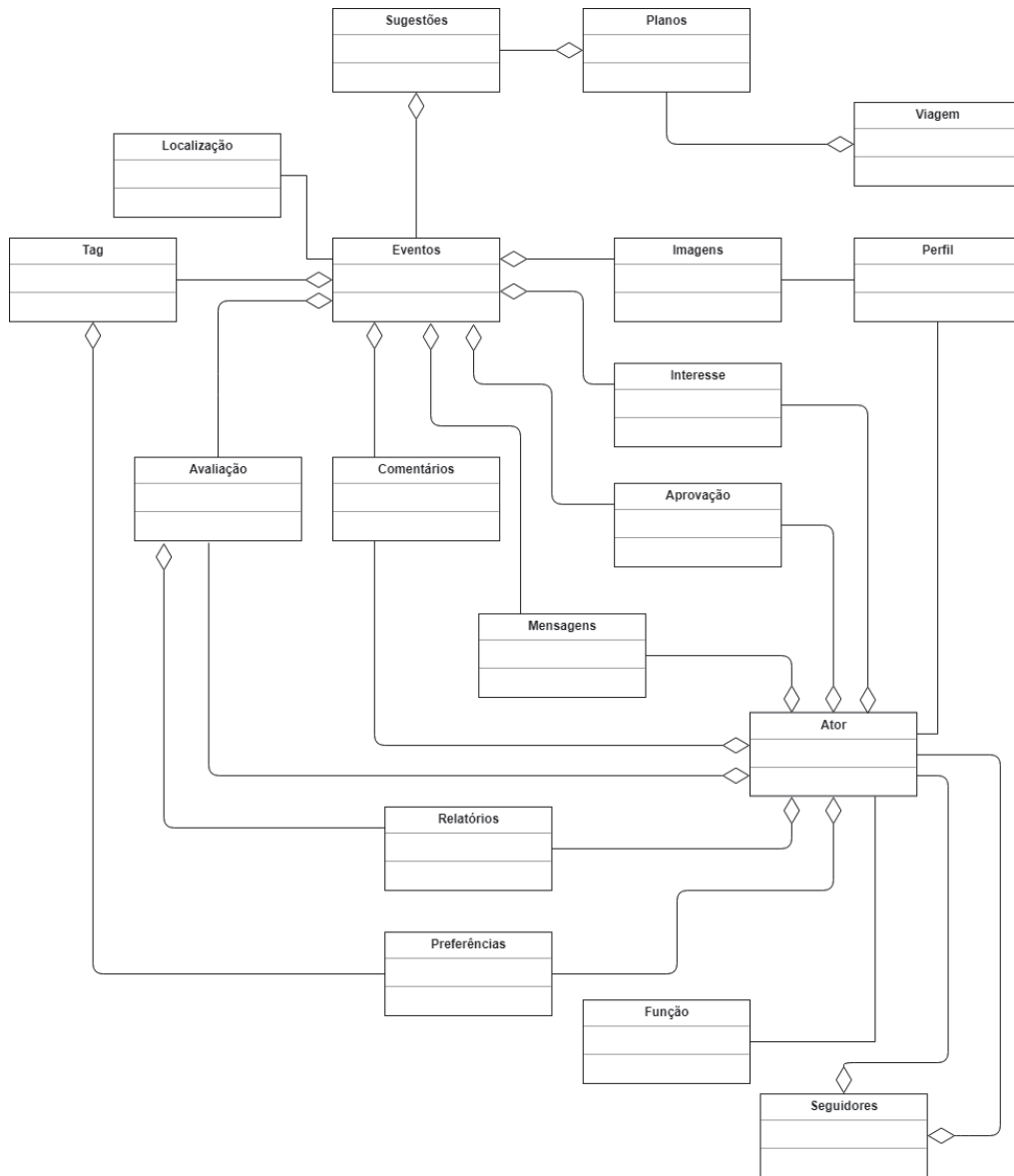


Figura 72 - Modelo de Domínio Final - Diagrama de Classes

## 5. Arquitetura e Tecnologias

Neste capítulo será apresentada a arquitetura escolhida e os motivos que levaram a essa escolha, assim como as tecnologias utilizadas no sistema.

### 5.1. Arquitetura

Um dos passos mais importantes no desenvolvimento do projeto é a escolha da arquitetura a usar. O uso de uma arquitetura adequada impulsiona o consenso, suporta o planejamento, reduz riscos, permite uma gestão menos complexa, traz maior facilidade em mudanças e reduz custos. (Menezes and OutSystems, 2012)

Muitas vezes, as arquiteturas falham devido a um investimento fraco e inconsistente na fase de desenho, falha no entendimento do contexto de negócio, falta de conhecimento das funcionalidades embudadas nas tecnologias envolvidas, basear-se em experiências passadas e não relacionadas.

Tendo em conta os pontos anteriores, os pontos chave para uma boa arquitetura são:

- **Bom alinhamento com o negócio:** suporte efetivo a funções e processos de negócio;
- **Promover a usabilidade:** fornecer recursos para melhorar a experiência do utilizador (UX);
- **Sustentável:** adaptável aos negócios e à mudança técnica da realidade;
- **Desempenho:** uso eficiente dos recursos disponíveis;
- **Escalável:** simples no dimensionamento e infraestrutura.

Com o objetivo de ter um código estruturado e organizado, para que o desenvolvimento fosse mais fácil, e sabendo quais os pontos que levam a uma boa arquitetura, optou-se por dividir a solução da aplicação em camadas distintas. Esta organização permite também que no futuro seja mais fácil a evolução da aplicação por outros programadores. A divisão escolhida foi de encontro à arquitetura também aconselhada pela OutSystems para o desenvolvimento de aplicações usando a sua tecnologia (CELLTT.COM, 2018). As 4 camadas escolhidas são:

- Camada de apresentação - GUI;
- Camada de lógica de negócio;
- Camada de acesso a dados;
- Camada de acesso ao servidor e entidades externas.

Quando se utiliza este modelo de 4 camadas no desenvolvimento em OutSystems, é de extrema importância que não haja referências cíclicas e no sentido de baixo para cima, pois causa dependências indesejadas entre módulos e uma grande perda de *performance* do sistema, tanto a nível do seu funcionamento, ou simples publicação desses mesmos módulos. Os módulos da camada 4 só devem ter referências entre eles ou para a camada exterior; já a camada 3 poderá ter entre os próprios módulos e a camada 4, e assim sucessivamente. Na **Figura 73** está representado um exemplo de um modelo de 4 camadas.

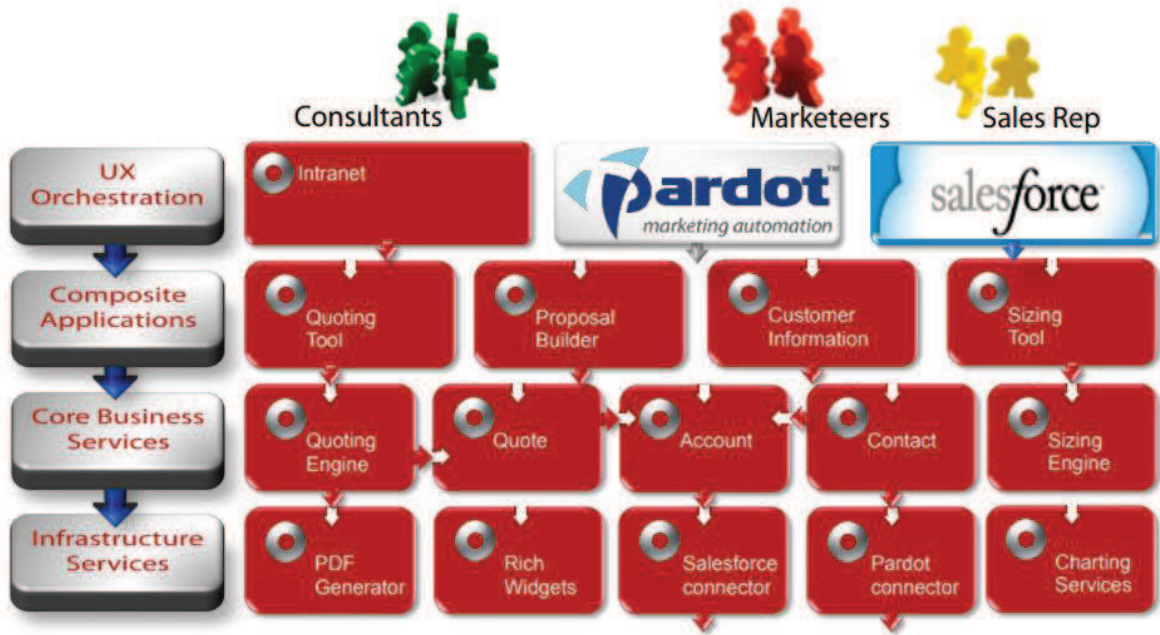


Figura 73 - Exemplo de um modelo de 4 camadas (Menezes and OutSystems, 2012)

Na **Figura 74** podemos ver a representação da arquitetura da aplicação. Estão representadas as 4 camadas e algumas características de cada uma. Seguidamente serão descritas em maior detalhe cada uma das 4 camadas.



Figura 74 - Arquitetura da aplicação

### 5.1.1. Camada de Apresentação - GUI

Nesta camada estão situadas todas as componentes de interface com o utilizador e a lógica destas. Todos os ecrãs com que os atores interagem na aplicação bem como a lógica de apresentação situam-se nesta camada.

### 5.1.2. Camada de Lógica de Negócio

A camada de lógica de negócio é aquela que tem mais lógica associada, pois suporta todas as funcionalidades core do negócio. Estas funcionalidades vão desde as mais básicas, como agrupar um evento numa categoria, até às mais complexas, como cálculos de relevância. Ainda dentro desta camada estão os *web block* que são como “fragmentos” de ecrã que podem ser reutilizados em diversos outros ecrãs.

### 5.1.3. Camada de Acesso a Dados

Nesta camada estão situadas todas as ações de acesso aos dados locais, todas as consultas ou alterações efetuadas em dados. Aqui é feito o acesso as tabelas e feita uma pequena triagem e tratamento dos resultados, de forma a que sejam mais facilmente consumidos pela camada de negócio.

### 5.1.4. Camada de Acesso ao Servidor e Entidades Externas

A camada de acesso ao servidor tem todas as ações onde é necessário interagir com o servidor. O objetivo é que a aplicação não dependa maioritariamente de conectividade a dados, por essa mesma razão são poucas as interações com o servidor.

Todas as alterações são feitas na base de dados local e posteriormente feita a sincronização, seja no *login* ou a pedido do ator ligado na aplicação. Poderá ou não ser uma sincronização massiva de toda a informação entre aquela instância da aplicação e o servidor.

Todas as ações de sincronização estão presentes nesta camada, podendo ser invocadas pelas camadas superiores.

### 5.1.5. Escalabilidade

Toda a aplicação foi desenvolvida em pequenas “peças” que se reutilizam e agrupam de forma a realizar as funcionalidades pretendidas. Por esta razão, as ações que invocam dados da base de dados local ficam numa camada completamente distinta e podem depois ser usadas pela camada superior (camada de negócio).

As ações da camada de dados devolvem os dados de uma forma genérica e o mais tratada possível. Assim, cada ação da camada superior pode utilizar as várias ações e criar lógica com os dados obtidos, de forma a construir uma funcionalidade única ou genérica, dependendo do pretendido em cada situação.

Em seguida, os ecrãs podem consumir essas funcionalidades e as ações da camada inferior reaproveitadas entre ecrãs. A integração de novos ecrãs poderá ter impactos reduzidos no sistema tendo em conta a arquitetura da aplicação.

## 5.2. Tecnologias usadas

Neste subcapítulo são descritas todas as tecnologias usadas para o desenvolvimento deste projeto, bem como o motivo pelo qual foi escolhida cada uma delas.

Na escolha da tecnologia a utilizar, é necessário avaliar qual a que mais se adequa ao nosso projeto e ao futuro do mesmo. Frequentemente surgem novas tecnologias que prometem ser revolucionárias na sua utilização mas nem sempre se adequam ao projeto em questão. A tecnologia a usar neste tipo de projetos deve ser uma tecnologia recente e promissora, mas ao mesmo tempo já deve ser robusta e ter muita documentação. Assim sendo, foi escolhida OutSystems como tecnologia principal para este projeto.

### 5.2.1. OutSystems

Foi escolhida a tecnologia OutSystems como a principal tecnologia a utilizar no desenvolvimento do projeto devido ao facto desta ser uma tecnologia com a qual o autor teve contato tanto a nível académico como profissional e a mesma ter um leque de ofertas abrangente para o pretendido. Usando apenas esta tecnologia é possível construir desde o *front-end* ao *back-end*, passando por toda a integração com APIs externas. É uma tecnologia recente e com muita procura, tal como demonstra o volume de mercado da mesma.

“A tecnológica tem um volume de negócios que ronda os 100 milhões de dólares (85 milhões de euros) e que, segundo a empresa, cresce mais de 70% anualmente.”(PEREIRA, 2018).

O crescimento da empresa OutSystems, que se afirma cada vez mais como uma das tecnologias de desenvolvimento futuro, mostra que é uma aposta de muitos consumidores no mundo do IT. Tendo em conta este ponto, considera-se que a tecnologia usada tem impacto no mercado atual e irá continuar a ser utilizada por mais tempo, não havendo risco de criar uma aplicação com uma tecnologia com validade.

“Este tipo de tecnologia, designada *low code* (e até há pouco tempo conhecida por *agile software*), permite aos programadores desenvolver aplicações de software de forma mais rápida e simples do que as tradicionais linguagens de programação e codificação, caracterizadas pela sua complexidade e elevado nível de abstração.” (PEREIRA, 2018)

As principais características da tecnologia OutSystems e que levaram à sua escolha para este projeto são:(Menezes and OutSystems, 2012)

- Uma infraestrutura escalável;
- Gestão de segurança e utilizadores;
- Uma *stack* independente;
- Multi linguagens;
- Integração fácil;

- *Multi-tenant*;
- Um IDE próprio de suporte a arquitetura;
- Uma comunidade grande;
- Disponibilização de uma componente de gestão da aplicação e de todo o ambiente do sistema.

Um outro ponto importante foi a quantidade de funcionalidades que a própria plataforma oferece e a sua comunidade, podendo facilmente integrar os mais diversos serviços e APIs com os de outros fornecedores, como é o caso da *Google*. Com isto, o objetivo passou por manter o foco no desenvolvimento das diversas funcionalidades da aplicação e não tanto na integração com aplicações externas.

### 5.2.2. APIs da *Google*

APIs são aplicações desenvolvidas com o objetivo de fornecer funcionalidades aos consumidores das mesmas sobre um determinado sistema, podendo abranger todo o tipo de negócio, desde banca, telecomunicações, etc (Vertigo, 2017). As APIs utilizadas neste projeto são essencialmente componentes para enriquecer e manipular os mapas, lugares e eventos, para fornecer mais informação aos utilizadores da aplicação. Como tal, foram escolhidas APIs da *Google*. Para a utilização destas APIs é necessário configurar uma conta de *developer* e a utilização de uma chave pessoal em todas as invocações realizadas a essas mesmas APIs. A *Google* fornece de seguida um conjunto de ferramentas e relatórios de forma a gerir todas as aplicações configuradas na conta.

Algumas das APIs a usar são:

- *Directions API* – Permite calcular direções entre dois pontos, utilizando vários meios de transporte (Directions Service - Google, 2018);
- *Distance Matrix API* – Permite calcular a distância e tempo entre múltiplos pontos, tendo em conta o meio de transporte escolhido (Distance Matrix Service - Google, 2018);
- *Geocoding API* – Permite, de uma forma direta, efetuar pedidos de conversão entre endereços e coordenadas ou vice-versa (Geocoding API - Google, 2018);
- *Geolocation API* – Fornece um local e o raio de precisão, com base nas informações sobre as estações móveis e os nós wifi (Geolocation API - Google, no date);
- *Maps Embed API* – Possibilidade de adicionar o mapa e todas as funcionalidades do mesmo para uma página (Maps Embed API - Google, 2018);
- *Maps JavaScript API* – Permite a customização de mapas com conteúdo próprio em paginas web ou aplicações móveis. Pode também ser utilizador *Maps Static API*, que

retorna uma imagem estática do local, sem ser necessário embutir código na página (Maps JavaScript API - Google, 2018);

- *Places Search* – Permite procurar locais em várias categorias, como estabelecimentos, pontos de interesse. Uma pesquisa por um local retorna todos os locais na proximidade (Place Search - Google, 2018);
- *Place Details* - Através do identificador do local resultante da *API Places Search*, é possível aceder a todo o detalhe de informação armazenado no *Google Places* para este local (Place Details - Google, 2018);
- *Places Photo* – Através do identificador do local resultante da *API Places Search*, é possível aceder a todas as imagens armazenadas no *Google Places* para este local (Place Photos - Google, no date);
- *Street View API* – Fornece visualizações panorâmicas 360º de estradas designadas em toda a área de cobertura, sendo a mesma do *Google Maps*. Também foi utilizada *Street View Static API* com as mesmas funcionalidades mas devolve uma imagem estática para que não seja necessário embutir código dinâmico ou qualquer outro tipo de desenvolvimento (Street View Service - Google, 2018);
- *CustomerSearch* – Permite uma pesquisa através dos parâmetros enviados (tal como uma pesquisa por imagens de uma bola no motor de pesquisa *Google*), sendo que a resposta desta API é em formato de *JSON* (Custom Search - Google, 2018).

Ao longo dos últimos 30 dias deste projeto, foram registadas as solicitações, erros e latência média a cada uma destas APIs. Através dos relatórios disponibilizados na conta de *developer* da *Google* foi possível extrair os seguintes valores para cada uma das componentes - **Tabela 49**.

**Tabela 49** - Solicitações, erros e latência média para cada API nos últimos 30 dias

API	Solicitações	Erros	Latência Média (ms)
<i>Directions API</i>	89	0	97
<i>Distance Matrix API</i>	72	0	322
<i>Geocoding API</i>	1258	4	281
<i>Geolocation API</i>	127	0	55
<i>Maps Embed API</i>	23	0	0
<i>Maps JavaScript API</i>	16	0	26
<i>Maps Static API</i>	0	0	0
<i>Places API for Web</i>	19	0	0
<i>Street View Static API</i>	13	0	284
<i>CustomerSearch</i>	924	3	205

## 6. Protótipo

Neste capítulo é feita uma descrição de todo o sistema, iniciada pela representação da interface gráfica da aplicação e descrição das principais funcionalidades de cada ator. São também apresentadas algumas componentes de lógica associada ao funcionamento básico da aplicação e, por fim, é apresentado o RoteiroWeb – Assistente para Administração.

### 6.1. Protótipo da Aplicação Móvel

Neste subcapítulo é apresentada a interface gráfica da aplicação recorrendo a *screenshots* da aplicação desenvolvida. Optou-se por organizar o subcapítulo tentando ter em consideração a navegabilidade realizada por cada um dos atores.

Houve a necessidade de adicionar ou reajustar componentes aos ecrãs durante esta fase, tendo em conta o que foi desenhado na secção 4.1.3 (Interface Gráfica), podendo então haver discrepâncias entre *mockups* e *screenshots*.

#### 6.1.1. Visitante

Ao aceder à aplicação e antes de qualquer registo ou *login* efetuado, qualquer ator é considerado **visitante**, tendo apenas as funcionalidades deste.

O visitante tem acesso à lista de eventos, podendo apenas consultar essa lista; não tem permissão para ver o detalhe dos eventos nem para efetuar qualquer tipo de ação sobre estes. Os eventos dessa lista são sugeridos apenas com base na distância a que o visitante se encontra destes. Para que seja possível o visitante tomar a função de outro ator, este necessita de se registar na aplicação *Guia.me*.

#### **Registar**

Na **Figura 75** está representado o ecrã de registo, através do qual o visitante pode efetuar o registo na aplicação e, assim, ter acesso às restantes funcionalidades da aplicação.

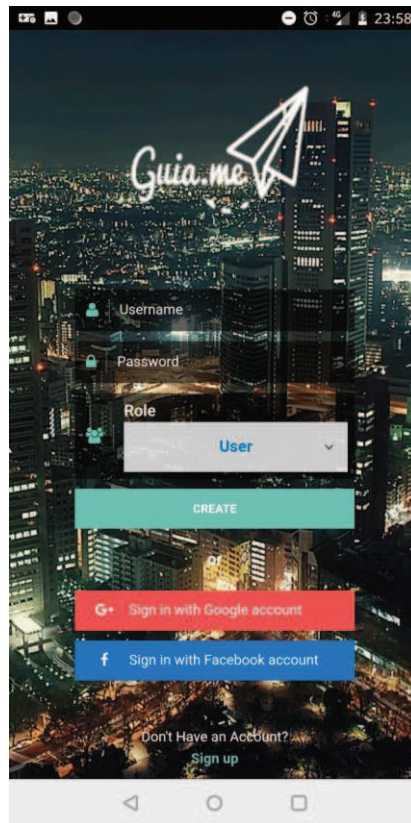


Figura 75 - Página de Registo

O visitante poderá registar-se de três formas: inserindo manualmente os seus dados e utilizando os registos por email, através da conta *Google* ou através do *Facebook*. Para proceder ao registo, deverá escolher o tipo de ator que pretende ser - **Figura 75**. Os pedidos para novos utilizadores são automaticamente aceites; ao contrário de pedidos para organizador e revisor, que carecem de aprovação por parte do administrador da aplicação. Optou-se por colocar o acesso a novos organizadores e revisores pendentes de aprovação por parte do administrador devido ao impacto que os mesmos têm no conteúdo da aplicação **Guia.me**. Contudo, esta componente poderá ser alterada no futuro, se houver justificação para tal.

### **Login**

Após registo efetuado, o visitante poderá efetuar *login* na aplicação, sem necessitar de selecionar a sua função (utilizador, organizador, revisor), pois esta é carregada automaticamente - **Figura 76**.

Após efetuar *login* na aplicação, o ator está habilitado para aceder ao **Guia.me**. É lhe apresentado um ecrã para escolha de meio de acesso à aplicação, se assim o desejar (poderá sempre aceder à aplicação sem qualquer tipo de validação): código pin ou utilização da impressão digital. Caso seja a sua primeira autenticação, antes de iniciar a utilização da aplicação, é apresentado através de uma sequência de imagens um pequeno resumo das funcionalidades da aplicação.

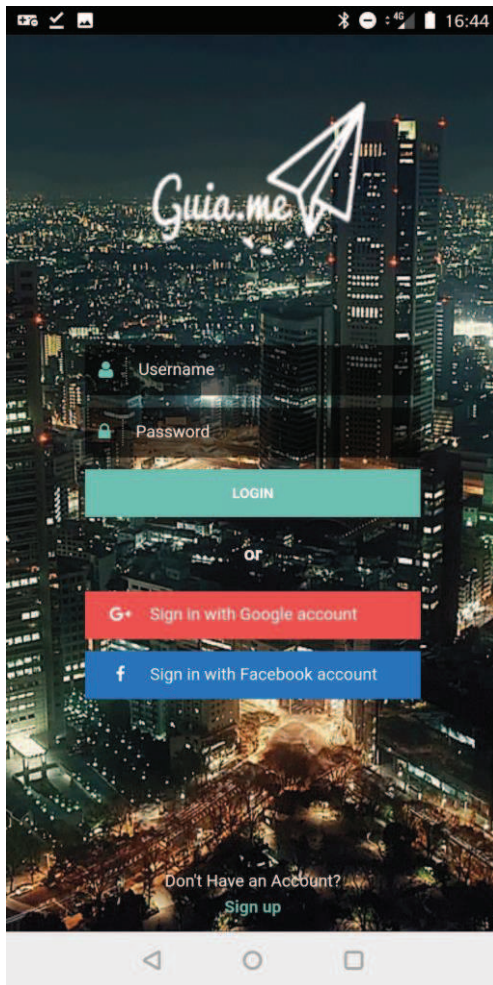


Figura 76 - Ecrã de Login

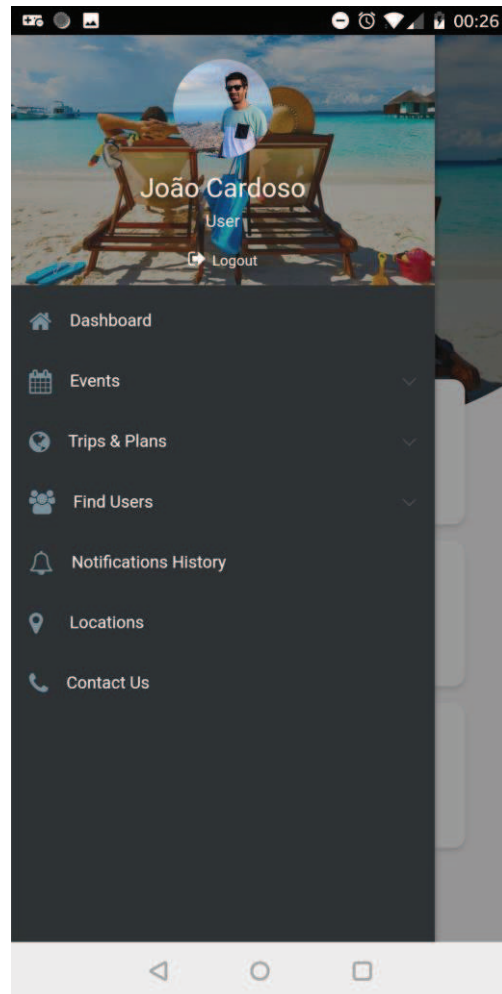


Figura 77 - Menu

### Recuperar Palavra Passe

Caso o ator pretenda recuperar a sua palavra-passe de acesso à aplicação, é possível pedir uma redefinição da mesma, através do preenchimento do formulário existente. Será feita a redefinição e enviada para o email existente na aplicação, podendo o ator ter novamente acesso à aplicação.

### 6.1.2. Utilizador

O **utilizador** é o ator que irá usufruir da oferta turística existente, ou seja, dos eventos disponibilizados na aplicação.

Após *login* efetuado na aplicação **Guia.me**, é apresentado um ecrã com a lista de eventos existentes sugeridos com base nos interesses do utilizador. Em qualquer momento, o utilizador (e qualquer outro ator) tem a possibilidade de aceder ao menu, permitindo navegar pela aplicação diretamente para as funcionalidades pretendidas - **Figura 77**.

### Ver Sugestões

Como referido anteriormente, ao aceder à aplicação **Guia.me** são carregados os eventos mais relevantes para o utilizador, calculados com base no descrito na secção 6.2.1 (Cálculo da Relevância do Evento). Estes eventos são apresentados no ecrã lista de eventos e cada um contém os elementos principais que o caracterizam e permitem evidenciar-se dos restantes eventos, ao olhar do utilizador - **Figura 78**.

Na lista, cada evento é representado por uma imagem que o caracterize (em plano de fundo), de forma a cativar rapidamente o interesse do utilizador, juntamente com informação básica sobre o mesmo. A informação presente em cada evento é a necessária para que o utilizador tenha uma visão geral do evento e consiga, rapidamente, perceber se tem interesse ou não no mesmo: título, localização, preço, motivo da relevância, categoria e outros utilizadores que vão ao evento.

É possível neste ecrã o utilizador expressar o seu interesse num determinado evento, através do ícone em formato de coração, presente no canto superior direito de cada evento. Esta funcionalidade pertence ao caso de uso **mostrar interesse**, que será detalhado posteriormente nesta secção.

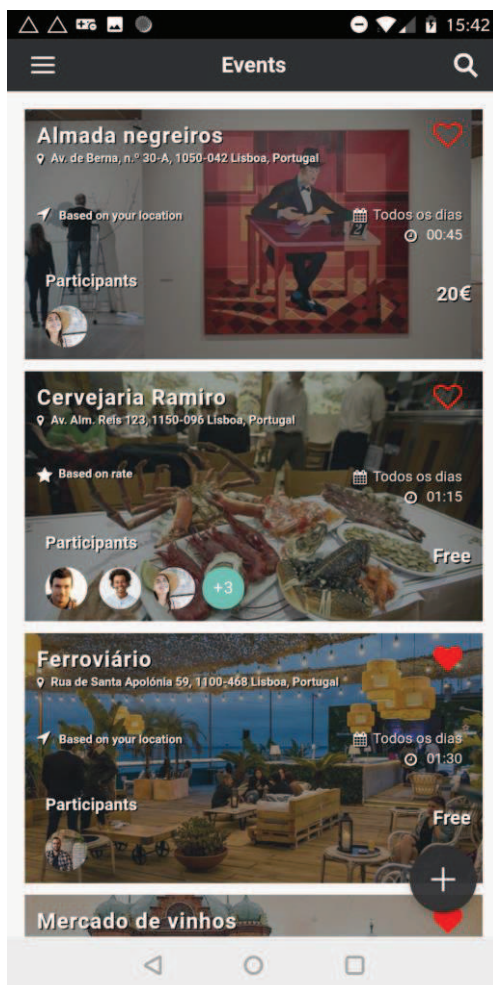


Figura 78 - Ecrã lista de eventos sugeridos

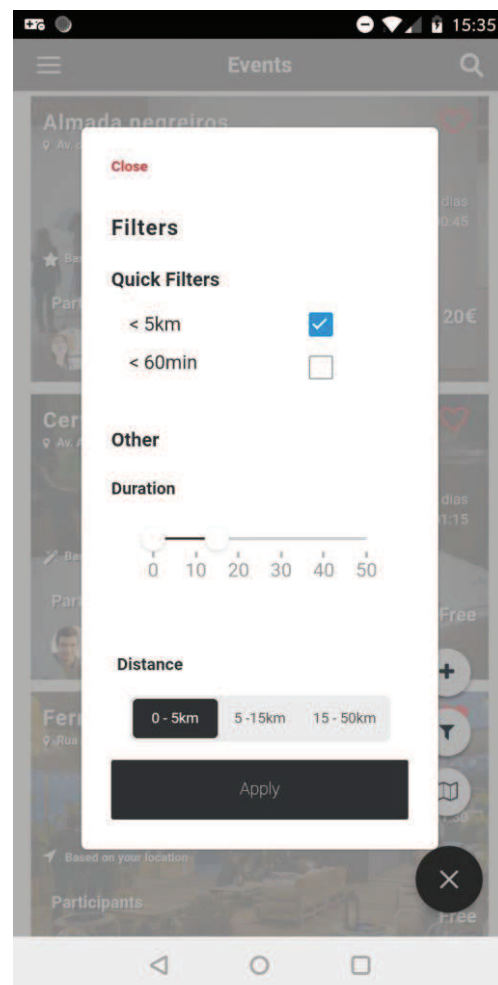


Figura 79 - Filtrar eventos

## Filtrar e Procurar Evento

Um dos objetivos do **Guia.me** é sugerir aos seus utilizadores uma lista de eventos o mais próximos possíveis dos seus interesses. Ainda assim, caso o utilizador pretenda refinar a lista de eventos fornecida pela aplicação, poderá adicionalmente filtrar os seus eventos tendo em conta a distância, categoria, classificação ou tipo - **Figura 79**. Quando aplicado o filtro de distância, é invocada a API *Distance* da *Google* e é apresentado em cada evento (por baixo da localização) a distância e tempo (a pé) a que o utilizador se encontra do evento - **Figura 80**. Esta funcionalidade está disponível para todos os atores.

Os filtros apresentados aos vários atores da aplicação estão divididos em – ver **Figura 79**.

- **Filtros rápidos** - filtros pré-configurados e disponibilizados no topo da página, para facilitar a escolha do utilizador. Estão pré-configurados os filtros de <5km e <60min pois um dos critérios mais usados pelos utilizadores na procura de eventos é a proximidade a que estes se encontram.
- **Outros** – aqui o utilizador pode configurar os mesmos filtros mas com maior rigor.

Para aceder ao filtro, o utilizador deverá selecionar a opção ‘+’ no canto inferior direito da página de eventos. Para aplicar o filtro, o utilizador apenas precisa de utilizar o botão de “aplicar”. Se pretender cancelar o filtro, basta utilizar o atalho na parte superior do *popup* para fechar o mesmo.

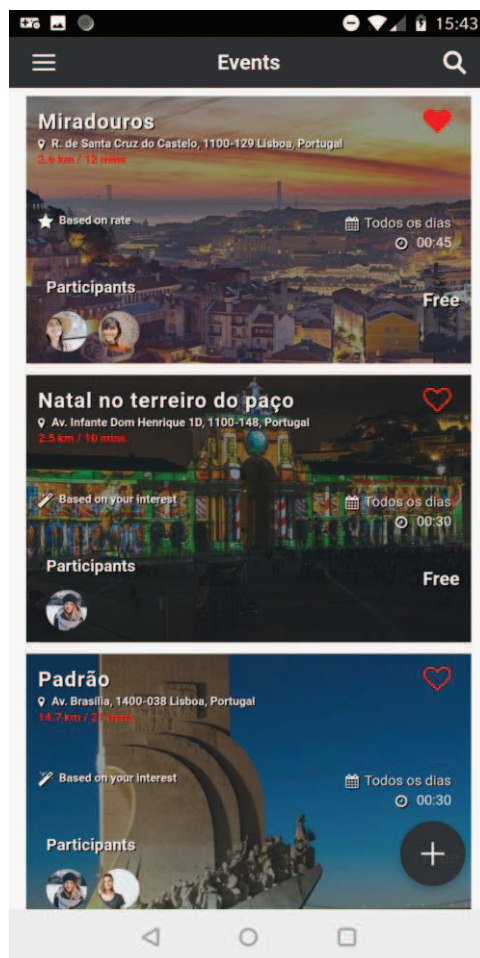


Figura 80 - Ecrã lista de eventos sugeridos (filtro)

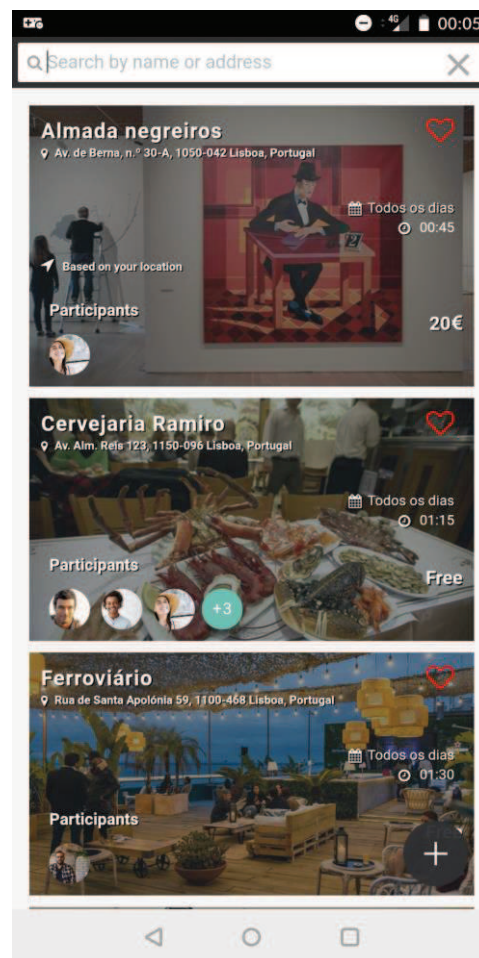


Figura 81 - Procurar evento

Em alternativa aos filtros e também disponível para todos os atores, é possível procurar por um evento em específico, através do seu nome ou localização. Após inserir a palavra a pesquisar e selecionada a tecla *enter* ou ícone de pesquisa no teclado do dispositivo móvel, a lista de eventos será atualizada. A opção de pesquisa foi colocada na parte superior de ecrã e é representada pelo ícone de lupa; este ícone é clicável e dá acesso à caixa de introdução de texto, onde deve ser inserido os parâmetros de pesquisa - **Figura 81**.

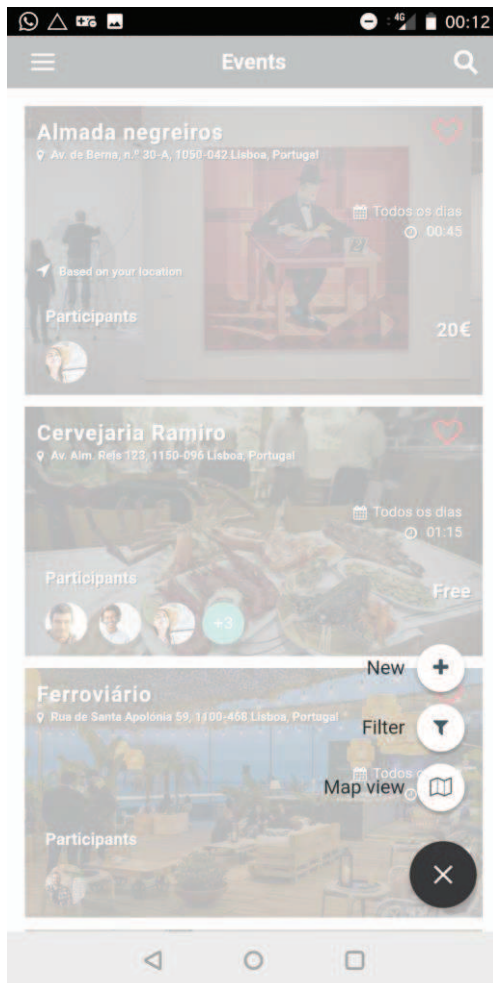


Figura 82 - Mais opções na lista de eventos

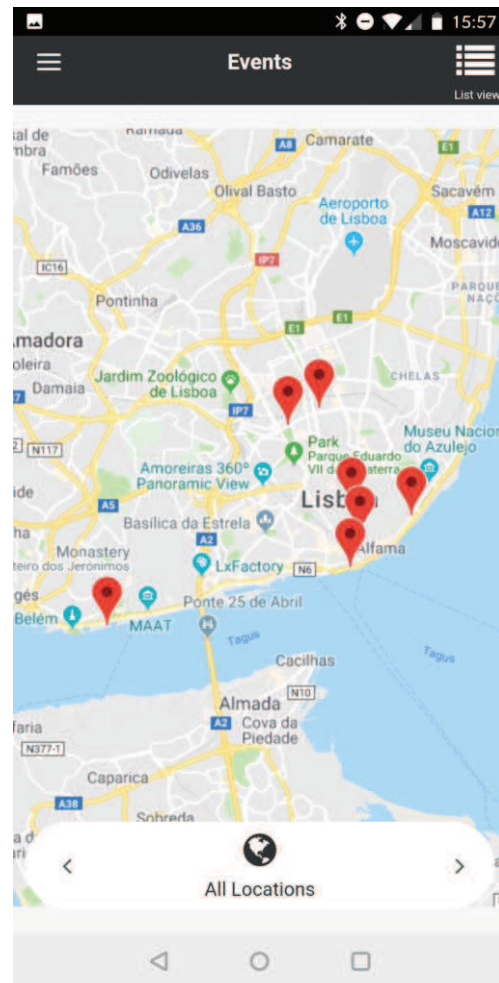


Figura 83 - Vista mapa de todos os eventos

Através da opção '+' no canto inferior direito da página de eventos (visível na **Figura 81**), para além de aceder aos filtros, é possível alterar a visão de lista de eventos para vista de mapa - **Figura 82**.

### Ver Localizações

A visão dos vários eventos sugeridos no mapa possibilita ao utilizador ter uma visibilidade da localização destes, sendo possível navegar por todos esses pontos nessa vista, ao utilizar as setas de navegação na parte inferior do ecrã - **Figura 83**.

Através desta funcionalidade é ainda possível aceder a muitas outras vantagens disponibilizadas pela *Google*, tal como a vista satélite ou camadas. Existe também a possibilidade de colocar o mapa em *full screen*, como é apresentado na **Figura 84**. Esta

funcionalidade faz parte do caso de uso **ver localização**. Selecionando um dos pontos é possível ver o trajeto da localização do utilizador até ao local, podendo depois usar uma aplicação externa como o *Google Maps* para navegar até esse ponto - **Figura 85**.

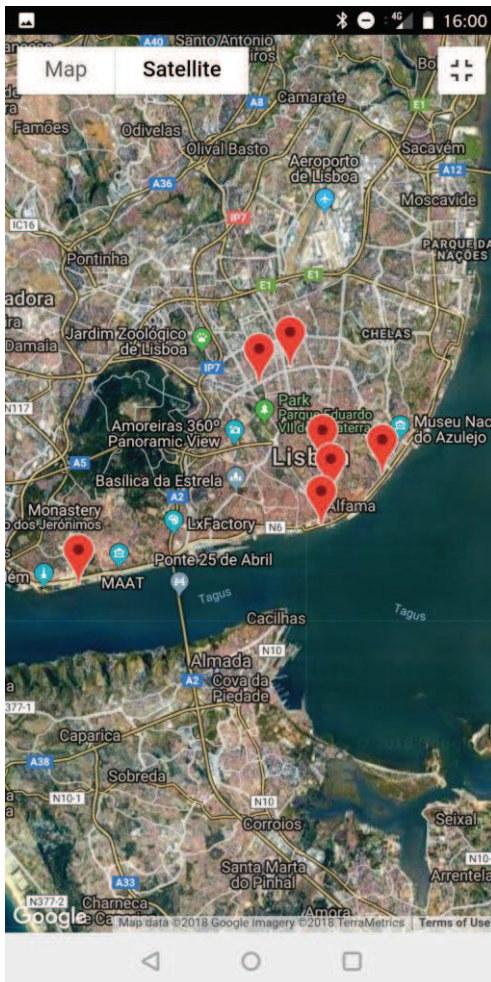


Figura 84 - Vista mapa satélite

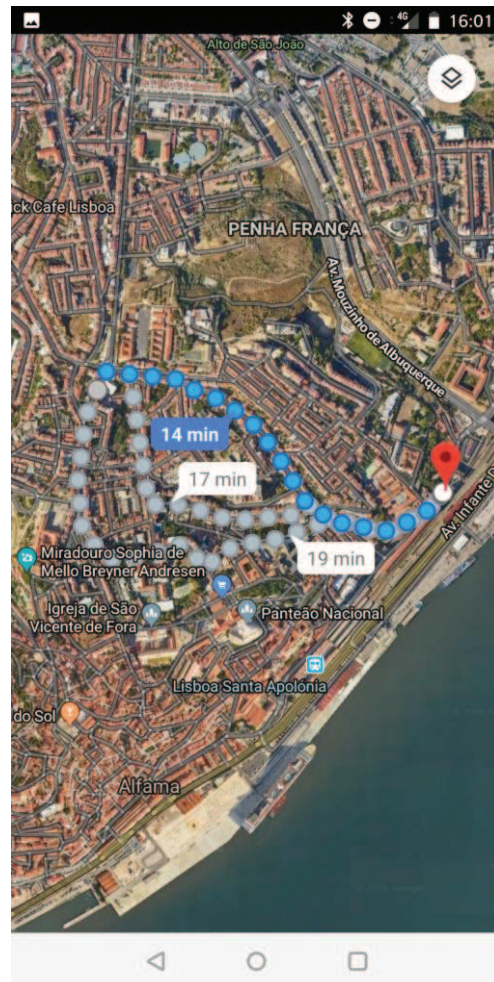


Figura 85 - Trajeto para um evento

### Ver Detalhe do Evento

Para aceder ao detalhe do evento, o ator (utilizador, organizador, revisor) deverá clicar no título do evento. No topo do ecrã de detalhe do evento é apresentada a imagem principal (a mesma apresentada no ecrã lista de eventos) e poderão ser apresentadas mais 4 imagens adicionadas pelo organizador. Será possível navegar através destas imagens deslizando para a direita ou esquerda - **Figura 86** e **Figura 87**.

No ecrã de detalhe do evento existem dois separadores: **Detalhes** e **Localização**. No separador de **Detalhes**, que será aquele que vem mostrado por defeito quando se acede aos detalhes de um evento, é visível a informação detalhada do evento: classificação, número de visitas, horários, localização, contato, fotos, comentários e categorias. No ecrã de detalhe do evento existe ainda a opção de enviar mensagem ou reportar um problema ao o organizador do evento (funcionalidade introduzida durante a fase de desenvolvimento final da aplicação e abordada posteriormente nesta secção).

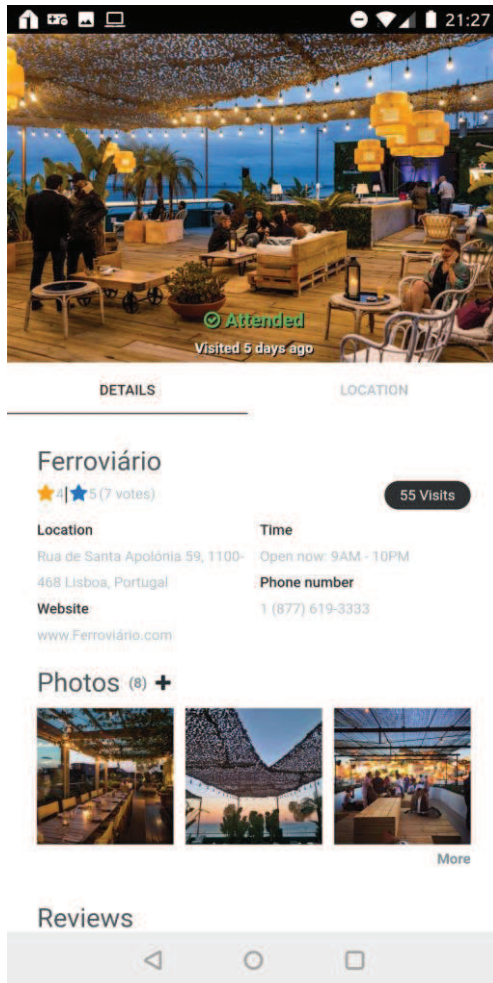


Figura 86 - Detalhe do evento

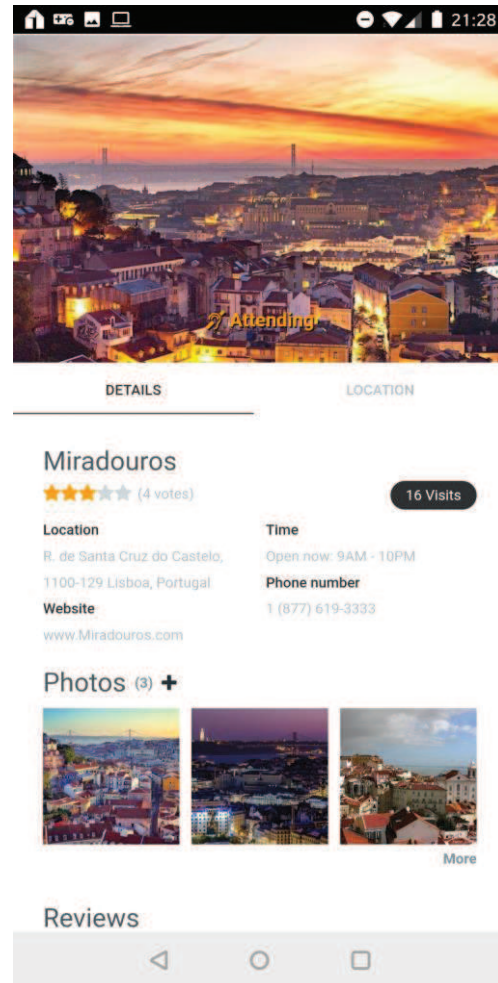


Figura 87 - Detalhe do evento

### Ver Localização

No separador **Localização** do detalhe do evento é visível o mapa com a localização do evento - **Figura 92**.

### Mostrar Interesse e Estive no Evento

Na parte superior do detalhe do evento (sobre a imagem principal) é visível o interesse do utilizador em relação ao respetivo evento, podendo assumir um de quatro valores:

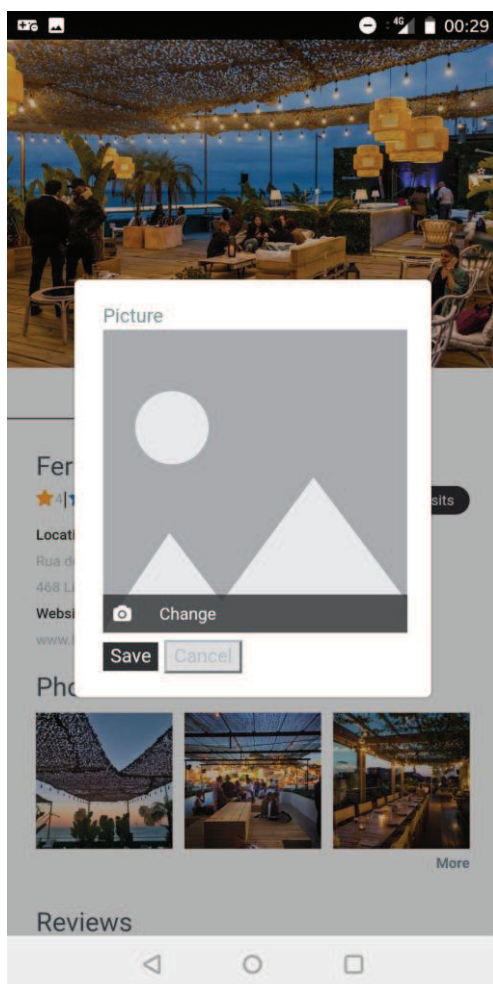
- **Interested** – O utilizador tem interesse em ir ao evento;
- **Attending** – O utilizador quer ir ao evento – **Figura 87**;
- **Attended** – O utilizador já esteve no evento. É apresentado o tempo que passou desde a sua presença no evento – **Figura 86**;
- **NotAttending** - O utilizador não tem interesse no evento.

### Avaliar e Ver Avaliações

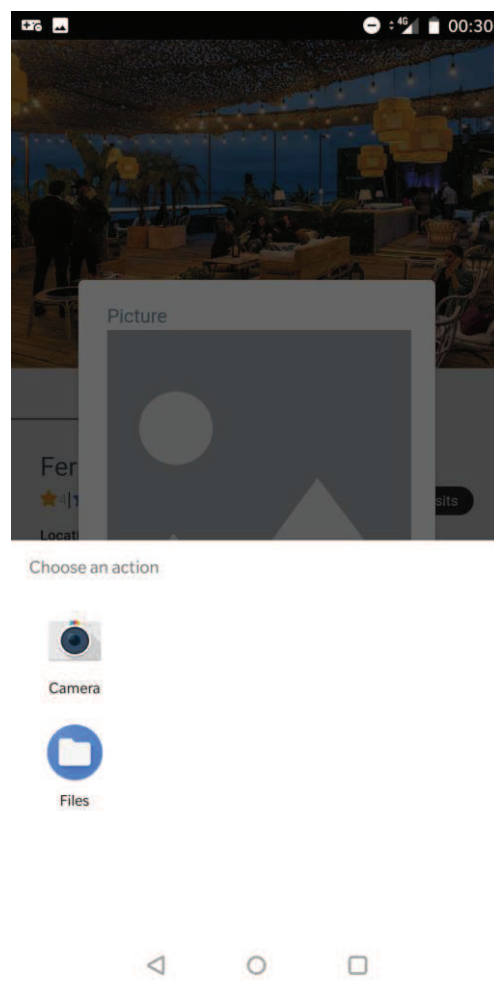
O utilizador poderá atribuir uma avaliação ao evento, com base na sua experiência. Para tal, deverá selecionar quantas estrelas desejar (de 1 a 5), sendo que maior número de estrelas representa melhor avaliação. A avaliação média da aplicação é recalculada e o valor obtido é representado pelo número de estrelas amarelas, abaixo do título do evento - **Figura 87**. A avaliação do utilizador é representada pela estrela azul - **Figura 86**.

### Ver e Adicionar Imagens

Ainda no ecrã de detalhe é possível visualizar as últimas três imagens que os utilizadores adicionaram ao evento. Se o utilizador pretender ver todas as imagens, poderá fazê-lo através da opção *more* - **Figura 90**. Para adicionar novas imagens ao evento, o utilizador deverá recorrer ao botão “+” (visível na **Figura 87**); ser-lhe-á apresentado um *popup* que permite adicionar fotos da galeria ou câmara - **Figura 88** e **Figura 89**.



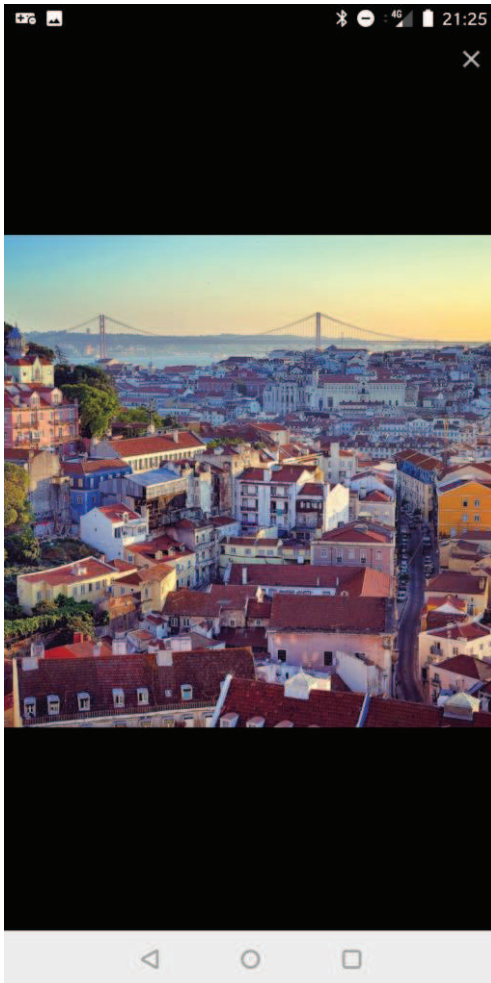
**Figura 88** - Popup nova imagem



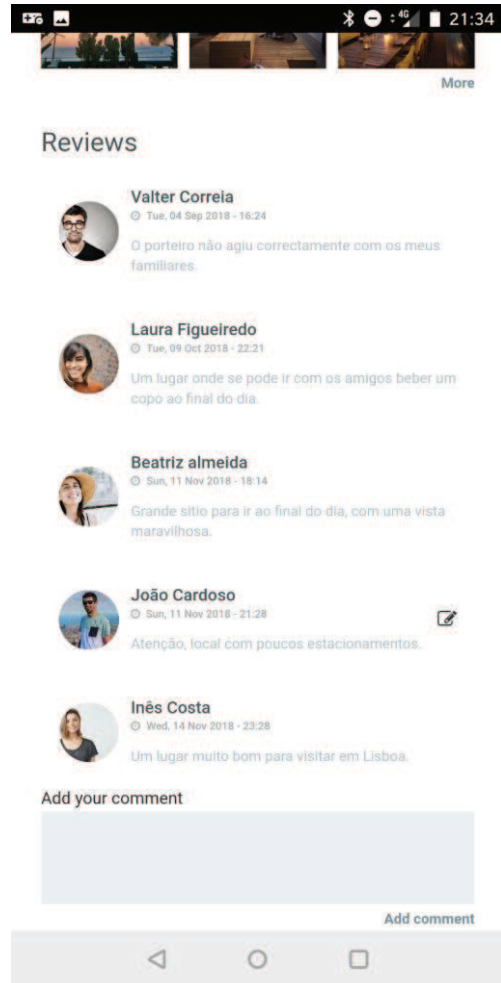
**Figura 89** - Opções de carregamento de nova imagem

### **Comentar e Ver Comentários**

O utilizador poderá deixar comentários sobre um evento, de forma a expressar a sua opinião sobre o mesmo. Os comentários são imediatamente adicionados, podendo ser removidos pelo organizador; neste caso, o administrador é notificado e será este quem tem a decisão final de remover ou não o comentário em questão. Os comentários são apresentados no detalhe do evento (após as imagens), sendo compostos pela fotografia e nome do utilizador e data e hora a que o comentário foi efetuado - **Figura 91**.



**Figura 90** - Visualização da imagem



**Figura 91** - Comentários do evento

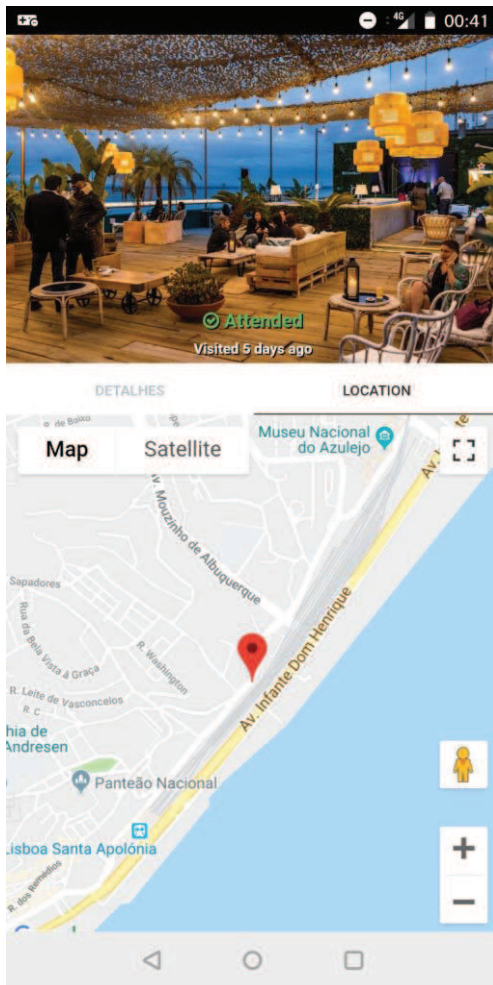


Figura 92 - Mapa evento único

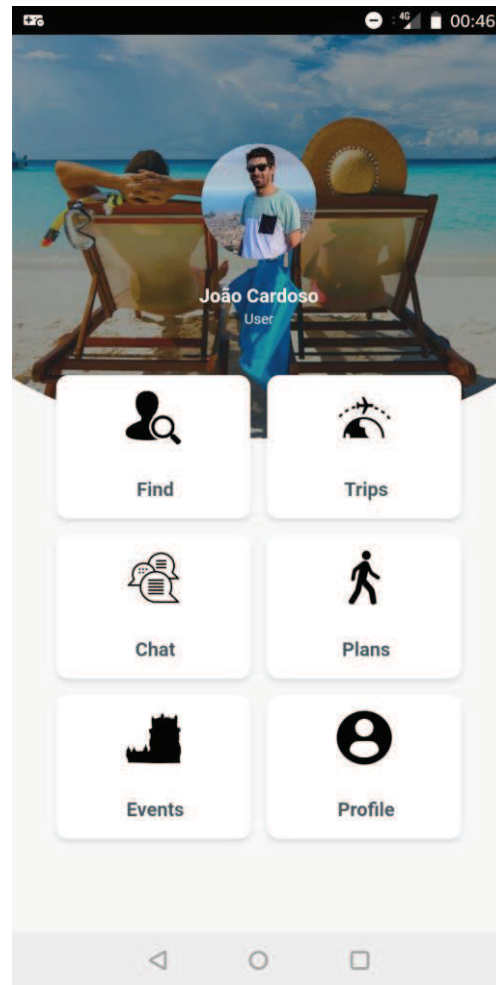


Figura 93 - Dashboard utilizador

### **Dashboard do Utilizador**

O *dashboard* do utilizador disponibiliza 6 atalhos para as suas funcionalidades principais - **Figura 93:**

- *Find* – Funcionalidade procurar;
- *Trips* – Funcionalidade de viagens;
- *Chat* – Funcionalidade de mensagens;
- *Plans* – Funcionalidade de planos;
- *Events* – Funcionalidade de ver eventos sugeridos;
- *Profile* – Funcionalidade ver perfil.

### **Procurar Utilizadores Seguidos e Utilizadores Com os Mesmos Interesses - Find**

Através do atalho “*find*” do *dashboard* é permitido ao utilizador procurar utilizadores por si seguidos ou utilizadores que vão aos mesmos eventos. O ecrã apresenta o número de resultados obtidos, ordenados por ordem alfabética e contendo a seguinte informação:

- **Utilizadores Seguidos**

Em cada utilizador apresenta a distância deste em relação a si, bem como a indicação se este está ou não *online* - **Figura 94**.

- **Utilizadores com Interesse nos Mesmos Eventos**

Em cada utilizador é apresentada a informação relativa ao número de eventos em comum entre os dois utilizadores, bem como a possibilidade de começar a seguir esse utilizador - **Figura 95**.

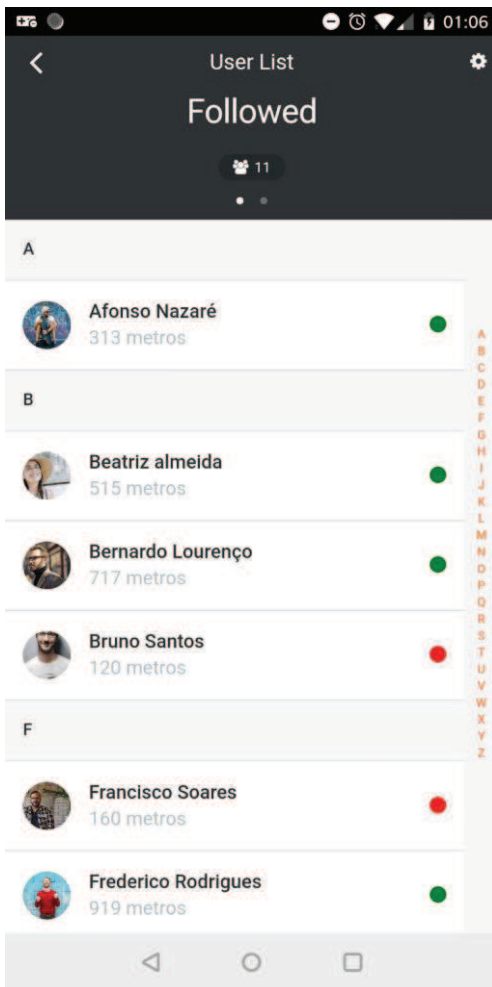


Figura 94 - Utilizadores seguidos

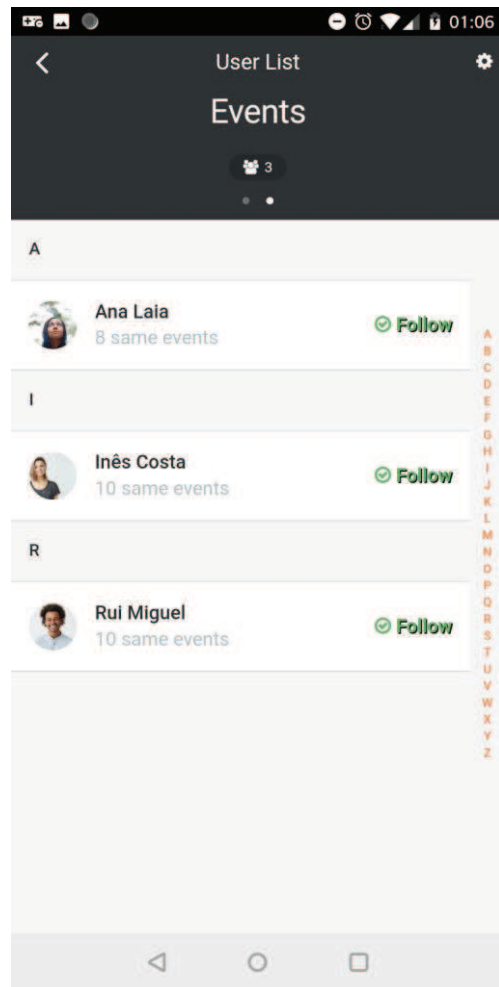


Figura 95 - Utilizadores com os mesmos interesses

**Ver Lista de Viagens e Consultar Viagem - Trips**

O utilizador tem a possibilidade de organizar os seus eventos, planos e informação adicional, de forma a criar uma “viagem”. As viagens ficam organizadas cronologicamente da mais recente para a mais antiga, sendo que na primeira posição é sempre apresentada a localização atual do utilizador, seguida, então, das suas viagens - **Figura 96**.

O utilizador poderá consultar cada uma das suas viagens e toda a informação a elas associada: planos, reservas, eventos, utilizadores seguidos que participem nos eventos da sua viagem - **Figura 97**.



Figura 96 - Viagens do utilizador

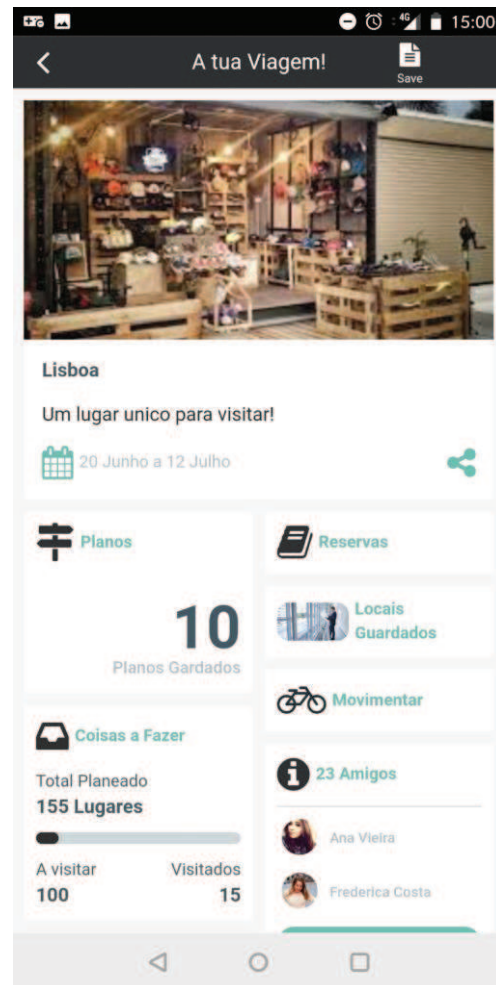


Figura 97 - Detalhe de viagem

### **Ver Mensagens e Conversar - Chat**

Na funcionalidade de mensagens ficam guardadas todas as interações entre os diferentes atores, podendo ser utilizadores, organizadores ou revisores. É possível armazenar todas as interações que o utilizador desejar pelo tempo máximo de 6 meses; **Guia.me** tem configurado um processo que apaga as mensagens anteriores aos últimos 6 meses.

As mensagens são agrupadas por utilizador, formando assim uma “conversa”. Cada uma é identificada pelo nome do utilizador, foto, texto, data e hora da última mensagem e número de mensagens por ler nessa conversa - **Figura 98**.

Uma nova conversa pode ser iniciada através do botão “+” (canto inferior direito). Para aceder a todas as mensagens trocadas numa conversa, basta seleccionar o nome do utilizador. No ecrã de uma conversa, as mensagens têm a hora a que foram enviadas ou recebidas e, no caso das mensagens enviadas, ainda têm informação se foram lidas. Para enviar uma nova mensagem numa conversa, o utilizador deve introduzir o texto na caixa de texto e clicar no ícone de enviar - **Figura 99**.

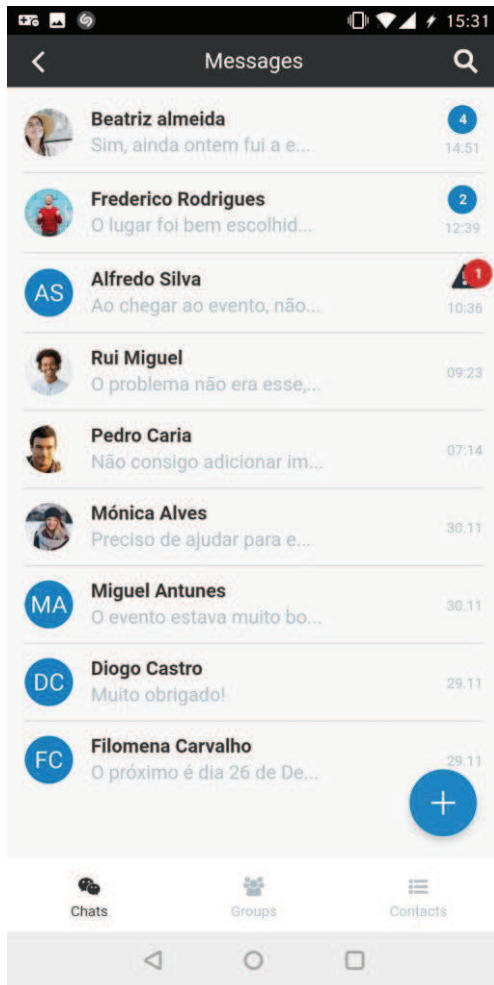


Figura 98 - Conversas do utilizador

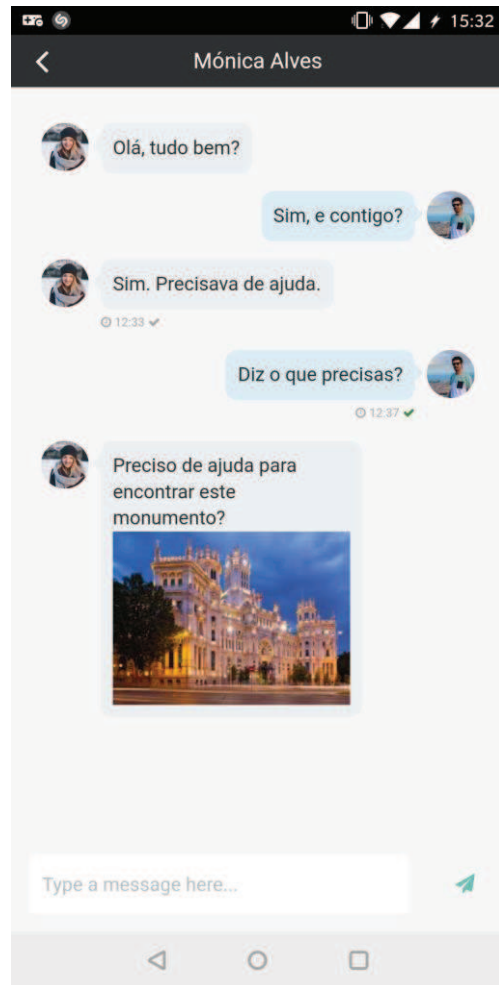


Figura 99 - Detalhe de uma conversa

### **Planear e Editar Plano - Plans**

O plano ou percurso é um conjunto de eventos disponibilizados em forma de linha temporal, tendo em conta as preferências do utilizador. Os planos podem ter diferentes durações: uma hora, várias horas ou dias. O plano é composto por uma linha temporal, onde está presente a seguinte informação - **Figura 100**:

- **Lado esquerdo da linha temporal:**
  - Hora a que deverá chegar ao evento;
  - Tempo, em minutos, entre dois eventos sequenciais.
- **Lado direito da linha temporal:**
  - Nome do evento;
  - Descrição do evento;
  - Tempo médio que demora a visitar;
  - Estado do evento em relação ao plano.

O utilizador pode gerar um plano automático com os seus eventos sugeridos, através da opção planos (“plans”). Durante a realização do plano, este vai sendo atualizado de duas formas possíveis:

- **Manualmente:** o utilizador clica no estado do evento para o qual se está a dirigir e esta ação irá desencadear que esse mesmo evento passe para o estado de “no evento”. Em seguida, o utilizador pode clicar novamente no estado do evento e este é alterado para “esteve no evento”, atualizando o evento seguinte para o estado “a caminho”, e assim sucessivamente durante toda a execução do plano;
- **Automaticamente:** a aplicação tem acesso à localização do dispositivo móvel e, no momento em que o utilizador chega ao evento, o estado do evento é alterado para “no evento”. O utilizador abandona o evento e o estado deste é atualizado para “esteve no evento”, atualizando também o evento seguinte para “a caminho”.

No plano é ainda possível ver os eventos que dele fazem parte em vista de mapa - **Figura 101**.

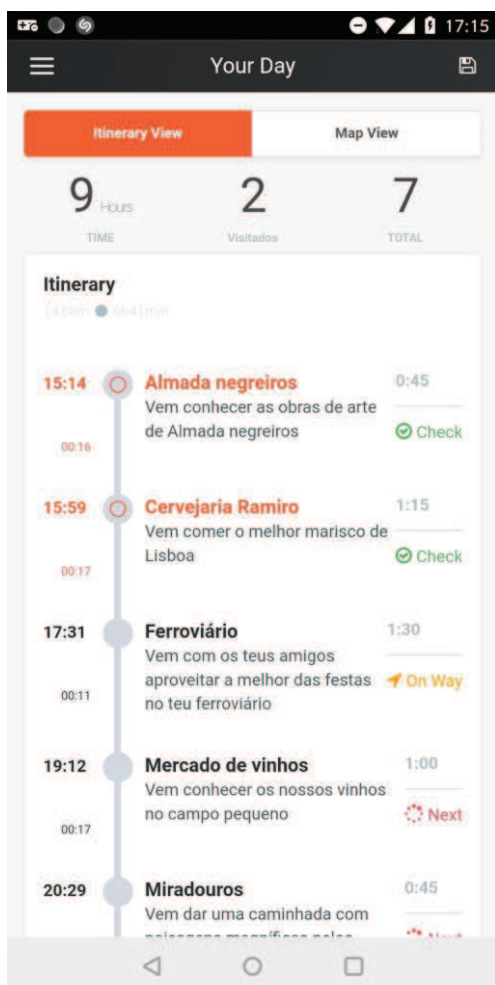


Figura 100 - Planos do utilizador



Figura 101 - Planos de utilizador (mapa)

### **Consultar Perfil, Alterar Dados e Alterar Foto - Profile**

O utilizador pode aceder ao seu perfil para consultar ou alterar dados pessoais, preferências, definições da aplicação, privacidade e pedidos, podendo também efetuar *logout* de *Guia.me* - **Figura 102**. Os utilizadores podem consultar também o perfil de outros, clicando nos seus nomes. Também o organizador e revisor têm acesso às funcionalidades referidas anteriormente.

Ao consultar o perfil, o utilizador poderá também alterar a sua foto (pressionando a foto atual), sendo apresentado um *popup* igual ao já mencionado anteriormente em situações de alterar ou adicionar imagens (**Figura 88**).

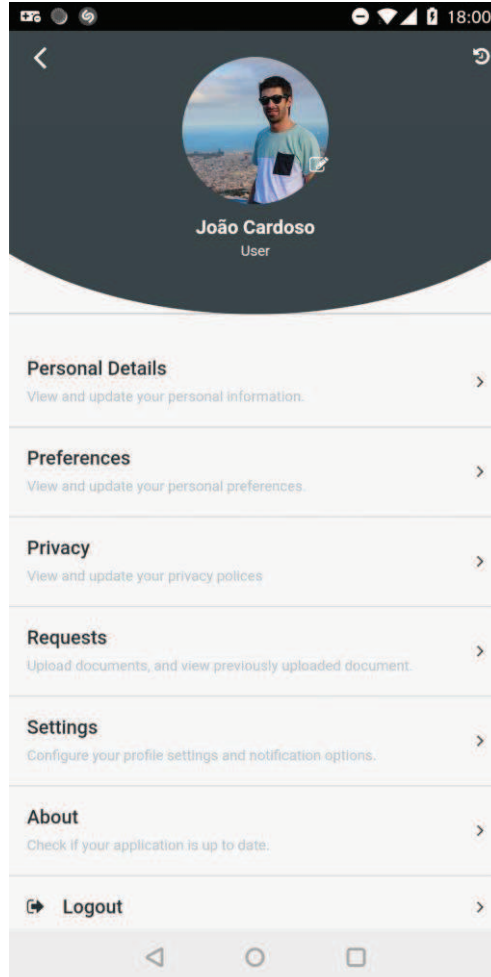


Figura 102 - Perfil

### 6.1.3. Organizador

O organizador é o ator que cria os eventos, responsável por todo o conteúdo neles adicionado.

#### **Ver Eventos Populares e Pendentes**

À semelhança do utilizador, também o organizador tem disponível uma lista de eventos ao iniciar a aplicação **Guia.me**, sendo estes os eventos que estão pendentes de ação sua. Se o organizador não tiver eventos pendentes, serão apresentados os seus eventos mais populares e dos quais é responsável. Ambas as listas são semelhantes à do utilizador, contendo assim as funcionalidades **procurar, filtrar e ver detalhes** - **Figura 78**.

## **Dashboard Organizador**

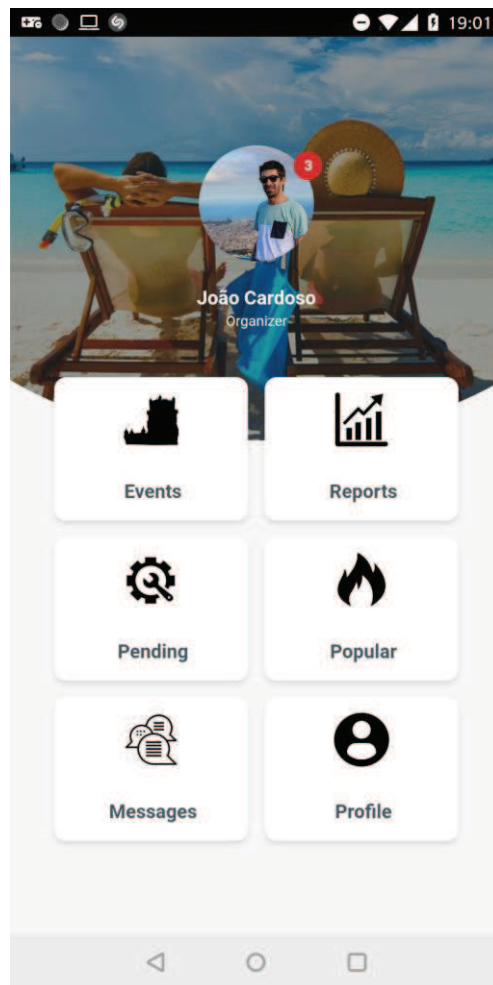


Figura 103 - Menu do Organizador

O organizador tem no seu *dashboard* as 6 funcionalidades que mais utiliza, sendo elas:

- *Events* - Lista dos seus eventos;
- *Reports* - Funcionalidade relatórios;
- *Pending* - Funcionalidade de eventos pendentes;
- *Popular* - Funcionalidade de eventos populares;
- *Messages* - Funcionalidade de mensagens;
- *Profile* - Funcionalidade de perfil.

### **Ver Eventos Populares - Popular**

Este atalho dá acesso aos eventos mais populares do organizador, para que o mesmo tenha a visibilidade sobre os seus eventos mais visitados e aqueles que recebem melhor avaliação. A lista de eventos contém as funcionalidades e características já referidas no início desta secção.

### **Ver Eventos Pendentes - Pending**

O organizador necessita de ter visibilidade sobre o estado dos eventos pelos quais é responsável. Para esse efeito, foi criada esta lista de eventos pendentes. Um evento é considerado pendente se:

- **Aguarda aprovação** – O evento publicado pelo organizador e, posteriormente atribuído a um revisor, ainda não foi aprovado;
- **Rejeitado pelo revisor** – O evento foi rejeitado pelo revisor porque não cumpria os requisitos, sendo o organizador notificado e ficando pendente de alteração sua;
- **Suspenso pelo revisor** – Após aprovação do evento, este fica disponível para os utilizadores poderem usufruir do mesmo. Se algo de errado se passar com o evento, o utilizador poderá notificar o organizador do problema, sendo que o revisor responsável pela aprovação também irá receber uma notificação e, se achar necessário, poderá suspender o evento. As notificações para ambos os atores são representadas na **Figura 103**. Adicionalmente, no ecrã de mensagens, será gerada uma conversa com o utilizador que reportou o problema, marcada como mensagem importante (triângulo vermelho) - **Figura 98**.

### **Criar e Publicar Evento - Events**

Através da opção *Events* do seu *dashboard*, o organizador poderá ter acesso à lista completa de todos os seus eventos. Para criar um novo evento, o organizador deverá seleccionar '+' e posteriormente clicar em "novo" (**Figura 82**). É apresentado ao organizador o ecrã destaque do evento (**Figura 104**) que deverá ser preenchido com o nome e uma breve descrição do evento. Através do botão "seguinte", passa ao próximo ecrã, ecrã de detalhe (**Figura 105**), onde deverão ser preenchidas as configurações do evento, como categorias, tempo médio de visita e horário. Por fim, o organizador pode aceder à configuração da localização, onde poderá pesquisar a localização do seu evento (**Figura 106**).

Após clicar em publicar, o evento será enviado para aprovação do revisor, que irá avaliar se o evento cumpre todos os requisitos. Caso o evento esteja correto, será aprovado e ficará disponível para os utilizadores.

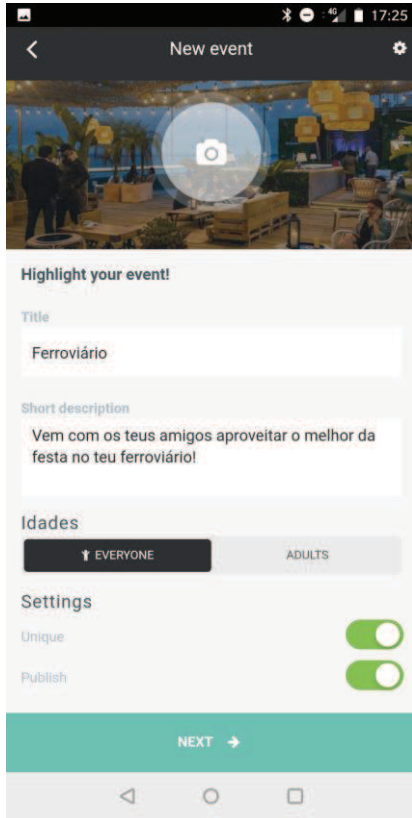


Figura 104 - Ecrã destaque do evento

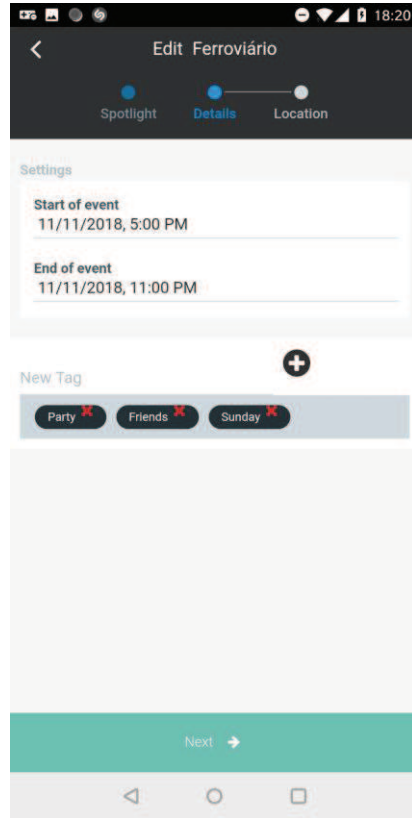


Figura 105 - Ecrã detalhes do evento

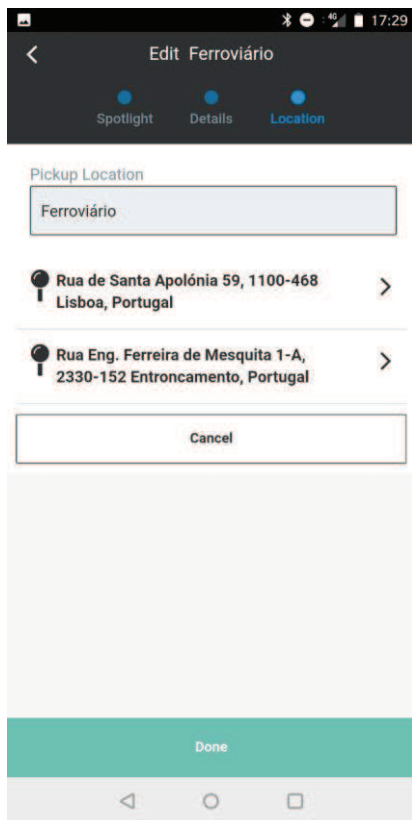


Figura 106 - Ecrã localização do evento

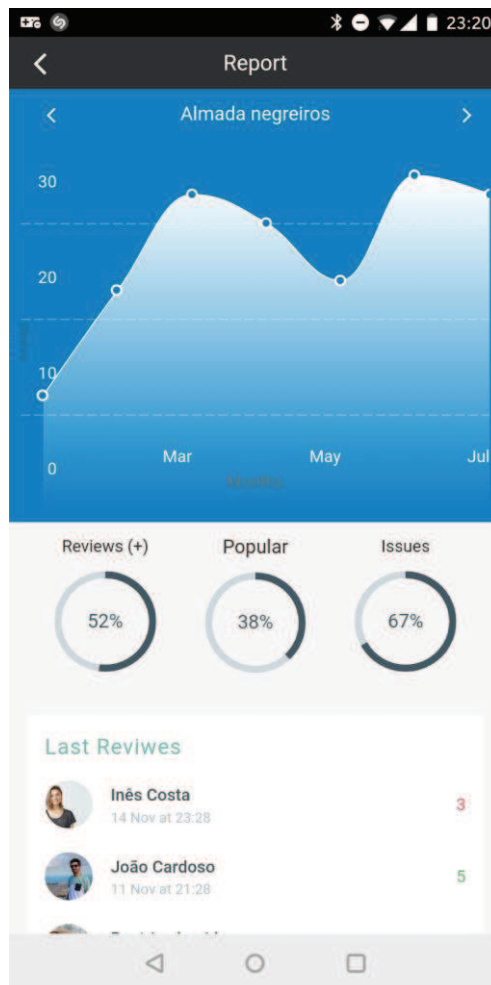


Figura 107 - Ecrã localização do evento

### **Ver Relatórios - Reports**

O organizador poderá consultar relatórios sobre os seus eventos, para que, de uma forma gráfica, seja possível visualizar o seu trabalho. Nos relatórios está disponível a seguinte informação - **Figura 108**:

- Últimas avaliações dos utilizadores aos eventos;
- Visitas mensais por evento;
- Percentagem de comentários entre um evento e o total dos seus eventos;
- Percentagem do tempo em que o evento esteve como popular desde que existe;
- Percentagem de problemas de um evento em relação aos restantes.



**Figura 108** - Relatório do Organizador

### **Ver Mensagens e Conversar - Messages**

Neste ecrã, o organizador tem acesso a todas as mensagens que troca com os restantes atores da aplicação. Esta funcionalidade é semelhante à funcionalidade *Chat* do utilizador, referida na secção 6.1.2. O organizador tem ainda acesso, neste ecrã, às notificações de problemas com os seus eventos - **Figura 98**.

### **Consultar Perfil, Alterar Dados e Alterar Foto - Profile**

A funcionalidade *Profile* permite ao organizador alterar todos os dados que lhe dizem respeito e que são configuráveis. Esta funcionalidade é semelhante à do utilizador, já referida anteriormente - **Figura 84** e **Figura 98**.

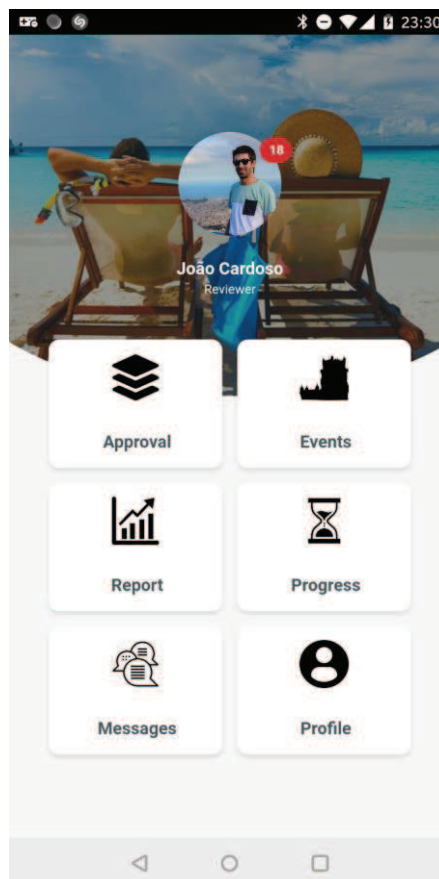
#### **6.1.4. Revisor**

À semelhança dos restantes atores da aplicação, também o revisor tem disponível uma lista de eventos no momento em que acede à aplicação, sendo esta lista composta pelos eventos pendentes de aprovação sua. Os eventos atribuídos a cada revisor são o mais próximos possível da sua área de residência, para o que o revisor tenha algum contexto do local.

### **Dashboard Revisor**

O revisor tem no seu *dashboard* as 6 funcionalidades que mais utiliza, sendo elas - **Figura 109**:

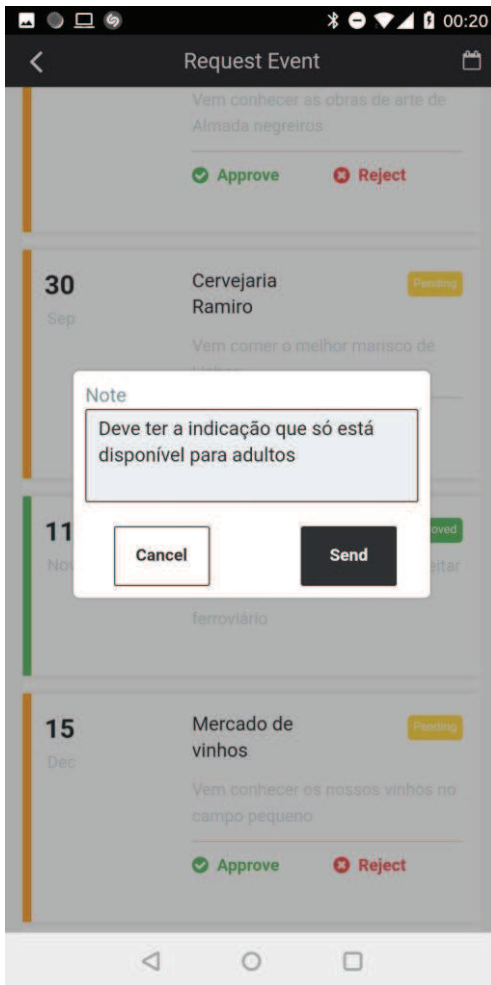
- *Approval* – Acesso aos eventos pendentes de aprovação sua;
- *Events* – Acesso a todos os eventos pendentes de todos os revisores;
- *Reports* – Acesso ao ecrã de relatórios;
- *Progress* – Acesso a todos os eventos aprovados por si;
- *Messages* – Acesso às conversas e mensagens;
- *Profile* – Acesso ao seu perfil.



**Figura 109** - Dashboard do Revisor

### **Aprovar e Rejeitar Evento - Approval**

Na funcionalidade *Approval*, o revisor tem acesso à lista de eventos pendentes da sua aprovação, que lhe são atribuídos com base na sua localização (definida no perfil). O revisor poderá aprovar um evento e este fica imediatamente disponível para todos os utilizadores de **Guia.me**. Se determinado evento não cumprir os requisitos para ser publicado, o revisor deverá rejeitá-lo e escrever uma mensagem para o organizador, informando sobre o motivo de tal rejeição - **Figura 110** e **Figura 111**.



**Figura 110** - Ecrã eventos pendentes (Reject Message)



**Figura 111** - Ecrã Approval

Através do seu *dashboard*, o revisor tem acesso a duas outras listas de eventos:

- *Events* – Lista de eventos pendentes de aprovação para todos os revisores;
- *Progress* – Lista de eventos já aprovados pelo revisor.

### **Consultar Relatório - Reports**

Através da funcionalidade *Reports*, no seu *dashboard*, o revisor pode consultar pequenos relatórios e extrair conclusões sobre o trabalho efetuado por si. Graficamente, os relatórios são semelhantes aos apresentados para o organizador (**Figura 108**).

### **Ver Mensagens e Conversar - Messages**

Como acontece com outros atores da aplicação, o revisor tem acesso à funcionalidade de mensagens, onde são visíveis todas as interações com os restantes atores. Muitas dessas interações são mensagens enviadas por utilizadores a reportar problemas em determinados eventos. O revisor pode aceder ao detalhe de uma conversa ou iniciar uma nova, tal como acontece com os restantes atores e já explicado anteriormente neste subcapítulo.

### **Consultar Perfil, Alterar Dados e Alterar Foto - Profile**

A opção *Profile* permite ao revisor alterar todos os seus dados configuráveis, à semelhança do comportamento já detalhado para o utilizador.

## **6.2. Lógica de negócio**

### **6.2.1. Cálculo da Relevância do Evento**

A aplicação *Guia.me* pretende otimizar o dia-a-dia de todos os utilizadores, através da sugestão de eventos de acordo com os seus interesses e preferências. Para obter a lista de eventos que mais se adequam ao utilizador, é feito um cálculo da relevância do evento com base nas três componentes que se seguem e atribuindo as diferentes percentagens a cada uma:

- **Categorias (60%):** A utilização da aplicação *Guia.me* pelos utilizadores fornece informação da navegação destes, através da soma de valores dessas ações para as categorias - **Tabela 50**.
- **Utilizadores (30%):** número de outros utilizadores seguidos (pelo utilizador em questão) que vão ao evento.
- **Distância (10%):** valor atribuído consoante a distância do evento ao utilizador, como configurado na **Tabela 51**.

#### **Categorias 60%**

As ações do utilizador dentro da aplicação são contabilizadas de forma a obter uma maior precisão dos eventos sugeridos ao utilizador. A tabela seguinte exemplifica a atribuição de pontos às diferentes categorias de um evento durante a navegação do utilizador na aplicação.

Tabela 50 - Pontuação atribuída a categorias

<b>Ações do utilizador</b>	<b>Pontos - Categorias Chave</b>	<b>Pontos - Outras Categorias</b>
<i>Abriu o evento</i>	3	1
<i>Deslizou fotos</i>		
<i>Uma foto</i>	3	1
<i>Mais de três fotos</i>	7	3
<i>Classificou</i>		
<i>Mais de 3 estrelas</i>	10	3
<i>Cinco estrelas</i>	15	5
<i>Comentou</i>	3	1
<i>Mostrou interesse</i>	5	2
<i>Vai</i>	7	3
<i>Percentagem do tempo médio gasto no evento</i>		
<i>&gt;10%</i>	10	1
<i>&gt;30%</i>	15	2
<i>&gt;50%</i>	20	5
<i>&gt;80%</i>	25	7
<i>&gt;100%</i>	30	10

Para que os pontos atribuídos durante o uso da aplicação não sejam elevados, todos os dias à 01:00 da madrugada é executado um timer que lança um processo, com o objetivo de nivelar os valores numa escala de 0 a 30. O valor 30 é atribuído à categoria com mais pontos e os pontos das restantes categorias são ajustados tendo em conta a relação entre valores dessas categorias (através de uma “regra de 3 simples”).

Os eventos têm obrigatoriamente de ter uma categoria principal e até 3 adicionais, que somando os pontos de todas contabiliza os 60% de relevância do evento para o utilizador.

#### **Utilizadores 30%**

Os utilizadores podem seguir outros utilizadores de forma a conhecer os eventos onde vão, mas estes também têm um peso importante para a relevância do evento, por cada utilizador (seguido pelo utilizador em questão), é somado 1 ponto para o total que no final corresponde a 30% da relevância.

#### **Distância 10%**

A distância do evento ao utilizador é a última componente utilizada, tendo apenas 10 % de peso nesta relevância. Os valores configurados são os apresentados na **Tabela 51**.

Tabela 51 - Relevância da distância ao ponto de interesse

<i>Distância do ponto de interesse ao utilizador (km)</i>	<b>Pontos</b>
<i>[0 - 1]</i>	30
<i>]1 - 2]</i>	25
<i>]2 - 3]</i>	20
<i>]3 - 4]</i>	15
<i>]4 - 5]</i>	10
<i>]5 - 7]</i>	7
<i>]7 - 10]</i>	5
<i>]10 - 15]</i>	4
<i>]15 - 20]</i>	3
<i>]20 - 30]</i>	2
<i>]30 - 50]</i>	1
<i>50 +</i>	0

No final os 100% correspondem à junção das três componentes e cada evento mediante do referido anteriormente terá uma relevância para o utilizador.

### 6.2.2. Sincronização Aplicação e Servidor

A sincronização entre a aplicação cliente e o servidor é uma componente importante e por vezes complicada de gerir e garantir.

Foram tidos em conta dois tipos de sincronização a usar nesta aplicação:

- **Consulta**, a sincronização realizada no sentido do cliente, ou seja, toda a informação é inserida no servidor e cada aplicação cliente que se liga faz o pedido ao servidor que retorna com a informação atualizada. Nesta situação as alterações são efetuadas sempre e apenas no servidor;
- **Consulta e edição**, aplicações que permitem a edição do conteúdo nos dispositivos móveis e posteriormente a sincronização para o servidor.

Neste sistema, optou-se pela manipulação de dados ser realizada maioritariamente do lado do cliente (dispositivos móveis), o servidor tem como objetivo armazenar e centralizar toda a informação atualizada vinda dos vários dispositivos. A manipulação de dados pode ser feita em diferentes dispositivos móveis em simultâneo, sem que esses mesmos dispositivos precisem de estar conectados à internet.

Todos os atores têm acesso ao sincronismo de dados, que é executado no momento de acesso à aplicação, mas que devido à complexidade e ao elevado volume de dados que pode ser transacionado, esta poderá ser feita através de uma sincronização denominada básica ou avançada. Na sincronização básica só é efetuada a sincronização de dados referentes aos

eventos que se encontrem na lista, sendo que se o utilizador abrir um evento este necessita de ser atualizado, ao invés da sincronização avançada já contém toda a informação sincronizada.

Sempre que são executadas algumas funcionalidades como andar pelas listas, abrir eventos, aceder as fotos, entre outras funcionalidades, são despoletadas ações que efetuam pedidos ao servidor para atualizar alguns dados da aplicação, todas estas atualizações são efetuadas sem que o utilizador se aperceba.

Quando a aplicação arranca é feita a sincronização mínima para a mesma funcionar e apresentando desde logo os eventos ao utilizador, enquanto continua a sincronizar a restante informação em *background*.

### **Implementação técnica**

Optou-se por resolver a questão da sincronização pela forma mais segura, com o objetivo de não ter muitos problemas, diminuindo o risco no curto prazo para desenvolver.

Se tivermos em consideração uma única entidade sem nenhuma ligação às restantes, a forma de sincronismo torna-se simples sendo utilizado um exemplo prático para esta representação:

- Entidade A, armazenada no servidor;
- Entidade B, armazenada no dispositivo móvel.

Ambas as entidades têm a mesma estrutura, com a exceção que a entidade B tem 3 atributos que apenas a ela lhe dizem respeito e não estão presentes no servidor.

- ***IsFromServer***, indica que o registo vem do servidor ou das aplicações cliente;
- ***IsModified***, indica que o registo foi modificado nas aplicações cliente;
- ***IsActive***, indica que o registo está ativo nas aplicações cliente.

Como referido anteriormente existem dois tipos de consulta, que serão detalhadas as suas implementações técnicas a seguir:

#### **ReadOnly – Consulta (Opção Analisada)**

O caso mais simples é o *read only*, onde o sincronismo é feito apenas do servidor para o cliente:

1. Seleciona os dados da respetiva entidade no servidor;
2. Todos os dados da entidade no cliente são apagados;
3. Insere todos os dados do servidor na entidade do cliente.

Quer isto dizer que sempre que acontece uma sincronização deste tipo nunca é possível modificar os dados do servidor, pois os pedidos são só de consulta.

#### **Read and Write – Consulta e Edição (Opção Escolhida)**

A situação de *read and write* é um pouco mais complexo, existem 3 consultas executadas na entidade local, sendo elas:

- **Registos com o atributo *IsFromServer* a falso**, dados da entidade local que não estão no servidor;
- **Registos com o atributo *IsModified* a verdadeiro**, dados da entidade local que foram modificados localmente desde de a ultima sincronização;

- **Registos com o atributo *IsActive* a false**, dados da entidade local que foram removidos no dispositivo móvel.

A aplicação cliente (dispositivo móvel) envia para o servidor os resultados das 3 consultas anteriormente referidas, e no servidor será então aplicada as alterações correspondentes a cada uma das consultas.

- **Registos com o atributo *IsFromServer* a falso**, são registos novos portanto são adicionados ao servidor sem validações adicionais;
- **Registos com o atributo *IsModified* a verdadeiro**, é realizado uma atualização dos dados no servidor que foram modificados;
- **Registos com o atributo *IsActive* a false**, os dados presentes no servidor são apagados.

Por último é efetuado o mesmo comportamento referido no *read only*, ou seja, é feita uma consulta a todos os registos existentes no servidor e enviados para o cliente móvel. Os dados do dispositivo móvel são apagados da entidade para depois inserir os novos dados vindos do servidor. Os dados inseridos no dispositivo móvel contemplam as 3 variáveis de sincronização já atualizadas:

- ***IsFromServer***, todos os dados têm este valor a verdadeiro pois todos eles estão no servidor;
- ***IsModified***, todos os dados têm este valor a falso, não sofrerem alterações pois vieram do servidor;
- ***IsActive***, todos os dados têm este valor a verdadeiro pois vieram do servidor.

Para garantir o comportamento correto, na aplicação é preciso manter os atributos referidos sempre atualizados dependendo das ações - **Tabela 52**.

Tabela 52 - Configurações para sincronismo

Ação	IsFromServer	IsModified	IsActive
<b>Novo registo</b>	falso	falso	verdadeiro
<b>Modificar registo</b>	Verdadeiro	Verdadeiro	verdadeiro
<b>Eliminar registo</b>	Verdadeiro	Falso	falso

Quando a entidade que vai ser sincronizada contém registos de outra entidade, a complexidade aumenta devido às chaves forasteiras e a integridade das relações. Nestes casos a lógica de sincronismo tem de ser adaptada a cada situação.

### 6.2.3. Cálculo do Plano Automático

Os planos automáticos são conjuntos de eventos disponibilizados cronologicamente em forma de percurso para os utilizadores, sendo gerados tendo em conta os eventos mais relevantes para o utilizador, os horários dos mesmos e o tempo que o utilizador tem disponível.

Para ter acesso a um plano automático o utilizador acede à aplicação e entra no ecrã de planos, escolhendo a opção de “gerar novo plano”. Posteriormente o utilizador coloca o tempo disponível e o tipo de evento que pretende, acedendo à lista de todos os eventos disponíveis no intervalo de tempo selecionado e ordenados por relevância (mais relevante para o menos relevante). O evento mais relevante é inserido no plano tendo em conta a hora em que começa (sendo que esta já se enquadra no intervalo de tempo selecionado pelo utilizador) e o tempo médio despendido no mesmo por outros utilizadores ou tempo necessário indicado pelo organizador. São colocados os eventos seguintes até completar o plano com o tempo disponível indicado pelo utilizador, sendo considerado o tempo entre eventos (deslocações) são também tidos em conta no plano. Para o cálculo da deslocação entre eventos, considera-se que o utilizador se deslocará a pé, sendo este método de deslocação configurável.

Os planos podem ser gerados de 3 formas:

- **Relevância** - o plano é construído utilizando os eventos mais relevantes, tal como foi descrito no exemplo a cima.
- **Relevância aleatória**, - o plano utiliza os eventos mais relevantes, mas de uma forma aleatória, ou seja, os eventos usados para o plano podem não ser exatamente os mais relevantes, mas sim, por exemplo, cinco eventos de entre os vinte primeiros mais relevantes (valor configurável pelo utilizador)
- **Aleatório** - os eventos usados para o plano são totalmente aleatórios não tendo em conta qualquer relevância. Para estes planos continua a entrar a componente de distância, a qual tem de estar dentro do raio definido pelo utilizador.

Quando o utilizador já tem um plano que lhe agrade, pode então dar-lhe início utilizando o ícone de *play* presente no ecrã. É apresentado o ecrã de navegação para o primeiro evento do plano ou, em alternativa, pode ser utilizado uma aplicação de navegação instalada no dispositivo móvel do utilizador.

Os planos são calculados com tempos médios de evento e tendo em conta o tempo gasto em deslocações entre eventos. Caso o utilizador se atrase, o plano é reajustado trocando o número de eventos ou eventos de maior ou menor duração.

#### 6.2.4. Processos e Timers

A plataforma da *OutSystems* permite projetar e gerir processos de negócio, integrando-os na aplicação. Normalmente, os processos são projetados a pensar no ciclo de vida de uma entidade, ou seja, como um fluxo de atividades a ser executadas em cada etapa do ciclo de vida. Um processo pode ser, por exemplo, a criação de um evento (*OutSystems - Processes*, 2017).

A possibilidade de executar lógica assíncrona ao funcionamento básico de *Guia.me* é feita através de *Timers*. Torna-se, então, possível executar um conjunto de tarefas, como por exemplo, o envio emails a uma determinada hora do dia. Os *timers* são geridos pelo *OutSystems Scheduler Service*, um serviço que verifica se há ou não *timers* prontos para executar e, em caso positivo, executa as suas ações (*OutSystems - Asynchronous*, 2016).

Os *timers* são executados quando a hora atual é maior ou igual que a hora configurada no *NextRun* do *timer*. Podem ser executados no máximo três *timers* em simultâneo (valor

configurado), devido ao elevado esforço que exige do processador. Quando existem mais do que três *timers* a ser executados ao mesmo tempo é utilizada a prioridade configurada. Quando o *timer* é executado e termina com êxito, a propriedade *NextRun* é atualizada para a próxima vez que este deve ser executado, sendo este tempo calculado de acordo com a propriedade do *shedule time*. Se o *timer* não for executado com sucesso, esta propriedade não é atualizada; significado isso que a próxima vez que o *OutSystems Scheduler Service* procurar por *timers*, esse será executado. A lógica assíncrona, como *Timers* e *Process Activities*, é executada numa sessão em separado: ao executar a ação associada ao *timer*, todas as variáveis de sessão têm o seu valor padrão (OutSystems Timers, 2017).

As aplicações móveis devem ser o mais fluidas e rápidas possível, pois nenhum utilizador vai querer esperar tempo entre as suas ações. Com esse objetivo, foram desenhados alguns processos:

- **Apagar Mensagens** - Processo que apaga todas as mensagens mais antigas que 6 meses;
- **Contador de visitas** - Processo que conta o número de novos registos na entidade *UserEvents* com o estado “já estive” no evento;
- **Cálculo de número e média de avaliações** - Processo que conta o número de avaliações para cada evento e faz a sua média;
- **Cálculo de relevância** - Cálculo que utiliza a atividade do utilizador para calcular a relevância.

Todos os *timers* estão atualmente programados para serem executados de dois em dois dias à 01h00.

Todos os processos anteriores são executados por *timers* de forma assíncrona ao funcionamento da aplicação, fazendo assim com que os processos mais demorados sejam executados de forma transparente para os utilizadores e em horas com menos probabilidade de estarem a usar a *Guia.me*.

### 6.3. RoteiroWeb - Assistente para Administração

Tal como mencionado no capítulo 4.1.1 Requisitos, na análise de requisitos inicial optou-se pela divisão da aplicação por vários perfis de utilizador/atores: visitante, utilizador, organizador e revisor. Foram estes os atores abordados durante a fase de modelação e para os quais foram elaborados o modelo de domínio, diagramas de sequência, casos de uso, diagramas de robustez, etc.

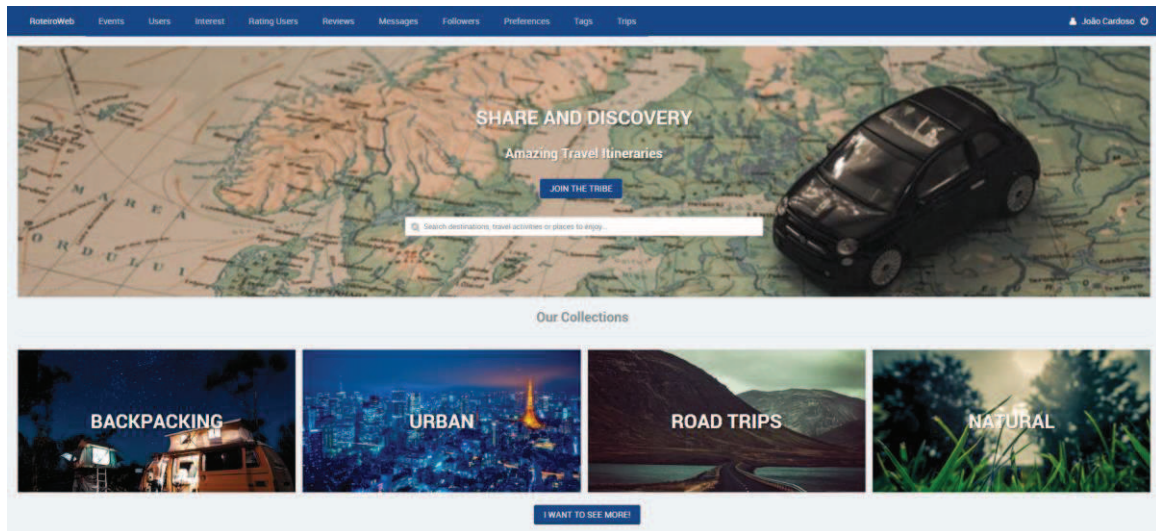
Contudo, na fase de implementação do projeto, surgiu a necessidade de ter uma plataforma de gestão para o sistema. Na fase de desenvolvimento prático do protótipo percebeu-se que seria mais rápido e fácil ter um *backoffice* numa página *web* que permitisse manipular dados mais facilmente para testes, de uma forma central a todo o sistema. Surgiu, então, o **RoteiroWeb** - ferramenta assistente de administração. Com esta ferramenta, surgiu também um novo ator - o administrador.

O administrador, através do **RoteiroWeb**, pode:

- Gerir os diferentes atores;
- Gerir categorias;

- Gerir comentários;
- Gerir avaliações;
- Gerir preferências;
- Gerir eventos;
- Gerir interesses.

A *homepage* do **RoteiroWeb** foi criada tendo por base os *templates* fornecidos pela própria plataforma OutSystems - **Figura 112**.



**Figura 112** - Homepage RoteiroWeb

**RoteiroWeb** disponibiliza dez menus na barra superior, dando acesso a dez listas com informação com a qual o administrador pode mais facilmente gerir a aplicação. As listas são:

- Eventos;
- Atores;
- Interesses;
- Avaliação;
- Comentários;
- Mensagens;
- Seguidores;
- Preferências;
- Viagens;
- Categorias.

A **lista de atores** é composta pela informação relevante de cada ator, como por exemplo: 1) nome, 2) email, 3) user, e 4) quatro indicadores do tipo de ator. O primeiro indicador diz respeito ao administrador e, como é visível na **Figura 113**, apenas um dos registos contém esta entrada como verdadeiro (o autor de **Guia.me**).

name	username	email	ADMIN	USER	HOST	REVIEWER
Beatriz Almeida	Beatriz	beatriz@almeida.com	✗	✓	✗	✗
Bernardo Lourenço	Bernardo	bernardo@lourenco.com	✗	✓	✗	✗
Bruno Santos	Bruno	bruno@santos.com	✗	✓	✗	✗
Francisco Soares	Francisco	francisco@soares.com	✗	✓	✗	✗
Frederico Rodrigues	Frederico	frederico@rodrigues.com	✗	✓	✗	✓
Inês Costa	Inês	ines@costa.com	✗	✓	✗	✗
João Cardoso	joaocardoso@gmail.com	joaocardoso@gmail.com	✓	✓	✓	✓
Laura Figueiredo	Laura	laura@figueiredo.com	✗	✓	✗	✗
Mónica Alves	Mónica	monica@alves.com	✗	✓	✗	✗
Pedro Carla	Pedro	pedrocarla@email.com	✗	✗	✓	✗
Rui Miguel	Rui	ruimiguel@email.com	✗	✗	✓	✗
Valter Correia	valter	valter@correia.com	✗	✓	✗	✗

Figura 113 - Lista de atores

Para consultar a informação mais detalhada de cada registo ou mesmo editar, basta clicar sobre o nome de utilizador, dando acesso a uma página com todos os atributos do ator, podendo os mesmos ser editados nessa página - **Figura 114**.

**João Cardoso - User**

Save image

Name: João Cardoso

Username: joaocardoso@gmail.com

Email: joaocardoso@gmail.com

Mobile Phone:

Creation Date: 2017-09-06 13:00:41

Last Login: 2018-12-09 13:42:29

Is Active:

User Type: User

Is Admin:

Figura 114 - Ecrã de detalhe e edição do ator

É possível criar um novo registo da entidade que está a ser consultada, através da opção idêntica à presente no canto superior direito da **Figura 113** (*Create New User*). É possível também importar e exportar dados para o servidor de forma massiva utilizando as funcionalidades de *import* e *export*, como visível no canto superior direito da **Figura 115**, utilizando como base um ficheiro excel.

A **Figura 115** representa uma **lista de eventos**, onde é possível identificar as principais características dos mesmos: imagens, nome, localização, classificação, etc.

PICTURE	PICTURE	ID	NAME	DESC	LOCATION ADDRESS	LOCATION LAT	LOCATION LONG	RATING COUNTER	RATING AVERAGE	VISITS COUNTER	DELETE
	79	103	Almada negreiros	Vem conhecer as obras de arte de Almada negreiros	Av. de Beira, n.º 30-A, 1050-042 Lisboa, Portugal	38.74	-9.15	1	3.00	0	
	80	104	Cervejaria Ramiro	Vem comer o melhor marisco de Lisboa	Av. Alm. Reis 123, 1150-096 Lisboa, Portugal	38.72	-9.14	4	3.50	1	
	64	105	Ferrovário	Vem com os seus amigos aproveitar a melhor das festas no teu ferroviário	Rua de Santa Apolónia 55, 1100-468 Lisboa, Portugal	38.72	-9.12	7	4.43	54	
	66	106	Mercado de vinhos	Vem conhecer os nossos vinhos no campo pequeno	Campo Pequeno 61, 1000-082 Lisboa, Portugal	38.74	-9.14	0	4.50	0	
	58	107	Miradouros	Vem dar uma caminhada com paisagens magnificas pelos miradouros de Lisboa	R. de Santa Cruz do Castelo, 1100-129 Lisboa, Portugal	38.73	-9.13	4	3.75	16	
	75	108	Natal no terreno do paço	Vem passar a noite de natal no terreno do paço	Av. Infante Dom Henrique 1D, 1100-148, Portugal	38.71	-9.14	2	5.00	0	
	51	109	Padrão	Padrão dos descobrimentos	Av. Brasília, 1400-038 Lisboa, Portugal	38.69	-9.21	50	4.50	89	

Figura 115 - Lista de eventos

Todas as restantes listas têm características e funcionalidades idênticas, todas elas com a hipótese de consultar e editar detalhes de cada registo ou criar um novo.

O **RoteiroWeb** consiste assim numa componente importante de gestão para o administrador da aplicação.

## 7. Conclusão

### 7.1. Considerações Finais

Com o crescimento do turismo em Portugal, houve um aumento exponencial de entidades a investir nesta área e, conseqüentemente, um aumento da oferta de eventos e métodos de divulgação dos mesmos. Um dos métodos mais utilizados pelas entidades para divulgação destes eventos são as aplicações móveis. Com os dispositivos móveis a fazerem parte da maioria das tarefas do cotidiano, acabam por ser também o meio preferencial dos turistas para obter informação sobre eventos, através de aplicações específicas.

Surgiu, assim, a oportunidade de desenvolver uma aplicação móvel com o objetivo de agilizar e rentabilizar o processo de gestão de tempo livre e escolha de eventos numa cidade ou região, bem como fomentar a comunicação entre utilizadores com os mesmo interesses, através de mensagens e notificações internas. Esta aplicação destina-se tanto a utilizadores interessados em consultar eventos e planear atividades, bem como a entidades que pretendam publicitar os seus eventos.

Para conseguir aferir sobre as características mais valorizadas neste tipo de aplicação, foram analisadas 5 aplicações semelhantes já existentes no mercado – *Google Trips*, *GuiaDeRodas*, *JiTT.travel*, *Minube* e *TripCase* - tendo em conta as principais funcionalidades, pontos fortes, pontos fracos, etc. Através desta análise, foi possível concluir que funcionalidades já existentes nestas aplicações são efetivamente úteis e que outras funcionalidade não existentes poderiam melhorar as mesmas. Assim sendo, foi possível elaborar um esboço das funcionalidades obrigatórias a implementar na aplicação a desenvolver e que poderão fazer com que se destaque das aplicações já existentes, sendo elas: 1) listagem de eventos sugeridos ao utilizador, com base nos seus interesses; 2) visão de mapa desses eventos; 3) detalhe de eventos; 4) comentários e avaliação aos eventos; 5) sugestão de percursos automáticos com base nos eventos sugeridos; 6) possibilidade de agrupar toda a informação sobre uma viagem; 7) interação entre utilizadores; 8) possibilidade de consultar conteúdos em modo *offline*, etc.

Após definir o objetivo da dissertação, foi escolhida a metodologia de trabalho a usar. Optou-se por seguir a metodologia ICONIX, por ser uma metodologia ágil e que permite avançar no projeto de forma iterativa e incremental - cada ciclo de desenvolvimento captura apenas um pequeno grupo de casos de uso, que direcionará o processo até se chegar à implementação.

Baseado na metodologia de trabalho adotada, em primeira instância foi efetuada a análise de requisitos, que consiste em: definição de requisitos; construção do modelo de domínio inicial; desenho da interface gráfica (recorrendo a *mockups* ilustrativos); navegabilidade entre ecrãs; e construção de casos de uso.

Nsta fase, foi tomada a decisão de dividir a aplicação por vários perfis de utilizador de forma a que cada um deles fosse um ator com um papel específico na mesma. O sistema deve permitir a interação entre os vários atores com as diversas funções, sendo que cada um desempenha um papel importante no sistema para que este funcione corretamente. Os atores do sistema são: 1) visitante; 2) utilizador; 3)organizador e 4) revisor.

Na segunda fase, efetuou-se a análise e desenho preliminar, que é composta pela descrição de cada um dos atores do sistema; descrição dos casos de uso apresentados na fase

anterior; construção de diagramas de robustez; e atualização do modelo de domínio, com base no trabalho desenvolvido nesta fase.

Na terceira fase – projeto – foi detalhada a componente mais técnica do sistema. Foi apresentado o modelo E/R e modelo relacional, bem como a descrição do modelo de dados, que permite identificar as entidades e o seu relacionamento. Foram construídos os vários diagramas de sequência, com base nos casos de uso, para que fosse representada a interação entre as diferentes camadas da aplicação. Por fim, foi atualizado novamente o modelo de domínio, com base nesta nova informação, dando origem ao Diagrama de Classes.

Antes de passar para a última fase da metodologia ICONIX (implementação) e para permitir ao leitor compreender melhor a descrição do protótipo implementado, foram apresentadas e detalhadas as tecnologias usadas e a arquitetura de software escolhida para esta aplicação,

Em termos de arquitetura, optou-se por dividir a solução da aplicação em 4 camadas distintas: 1) camada de apresentação – GUI; 2) camada de lógica de negócio; 3) camada de acesso a dados; 4) camada de acesso ao servidor e entidades externas. Esta organização permite que no futuro seja mais fácil a evolução da aplicação por outros programadores e vai de encontro à arquitetura também aconselhada pela OutSystems no desenvolvimento de aplicações usando a sua tecnologia. Foi escolhida a tecnologia *OutSystems* como a principal tecnologia a utilizar no desenvolvimento do projeto devido ao facto desta ser uma tecnologia com a qual o autor teve contacto tanto a nível académico como profissional e a mesma ter um leque de ofertas abrangente para o pretendido. Foram também abordadas neste capítulo as APIs da *Google*, componentes essenciais no enriquecimento e manipulação de mapas, lugares e eventos.

Na fase final, foi apresentado o protótipo do projeto implementado. Foram apresentados todos os ecrãs da aplicação e funcionalidades disponíveis para os 4 atores. Os ecrãs apresentados são *screenshots* da aplicação, variando em pequenos aspetos do desenhado na primeira fase (análise de requisitos).

Após implementação, conclui-se que a metodologia escolhida foi acertada, pois permitiu em cada fase ir incrementando novos cenários e corrigindo pequenas questões, de modo a tornar a aplicação cada vez mais próxima do pretendido. Concluiu-se também que a tecnologia e arquitectura escolhidas foram as mais adequadas ao projeto, por serem tecnologias robustas que libertam tempo na fase de implementação para que seja gasto no desenvolvimento de funcionalidades.

À data da conclusão desta dissertação já são conhecidas novas aplicações que se aproximam de algumas funcionalidade implementadas nesta aplicação; o que permite concluir que há uma evolução rápida e constante nesta área, para tentar oferecer ao utilizador aplicações cada vez mais completas e que vão de encontro às suas necessidades.

## 7.2. Trabalho Futuro

Devido ao crescimento tecnológico, há sempre espaço para evolução nesta área das aplicações móveis aplicadas ao turismo. Durante o desenvolvimento deste projeto foram surgindo ideias para aplicações futuras ou para otimizar as soluções atualmente existentes, nomeadamente:

- Inteligência artificial, por base de *machine learning*:

Com o avanço tecnológico, os computadores ganham cada vez mais capacidade de “concorrerem” com o ser humano. *Artificial intelligence* é um ramo da ciência que pretende criar máquinas com inteligência, sendo um sonho que se vem tornando cada vez mais real (Techopedia, 2018). Tudo isto está cada vez mais perto da realidade devido a conceitos como *Big Data* e *Machine Learning*. *Big Data* representa o crescimento exponencial de dados que atualmente ainda não são processados em tempo tolerável, originados pelos mais diversos sensores, dispositivos e aplicações (IBM Analytics, 2018). *Machine Learning* é a construção de algoritmos que capacitam as máquinas de aprender a fazer algo para o qual não foram expressamente programadas, através de um grande volume de dados (COPELAND, 2016). Assim, é importante numa aplicação como a **Guia.me** a implementação de inteligência artificial, com o objetivo de aproximar cada vez mais a sugestão de eventos aos interesses dos seus utilizadores. Esta funcionalidade seria um tema muito interessante e útil de ser abordado no futuro.

- Vários Idiomas

Apesar de o idioma escolhido para a aplicação ser o inglês, por ser considerado o idioma universal (COPELAND, 2016), considera-se importante que a mesma estivesse disponível no idioma de origem do utilizador.

- Navegação entre Eventos

Atualmente, a navegação é realizada usando uma aplicação paralela como o *Google Maps*, onde é possível navegar entre os eventos. Considera-se importante esta funcionalidade passar para a própria aplicação **Guia.me** para que a mesma não esteja dependente de outras aplicações instaladas no dispositivo móvel.

- Vários Utilizadores

A aplicação foi utilizada apenas em dois dispositivos em simultâneo. Considera-se importante avaliar a sua utilização com múltiplos utilizadores e grande quantidade de informação, de forma a validar as suas capacidades.

## 8. Bibliografia

Barata, L. M. S. S. de A. (2016) 'GESCORR – Sistema de Gestão Documental e Processos Empresariais'. Available at: [https://repositorio.ipcb.pt/bitstream/10400.11/6027/1/GesCORR - Gest3o de Processos Empresariais\\_A.pdf](https://repositorio.ipcb.pt/bitstream/10400.11/6027/1/GesCORR - Gest3o de Processos Empresariais_A.pdf) (Accessed: 1 November 2018).

Beck, K. and Andres, C. (2012) *Extreme Programming Explained : Embrace Change*. Second Edi. Edited by Addison-wesley professional. Paperback. Available at: <http://ptgmedia.pearsoncmg.com/images/9780321278654/samplepages/9780321278654.pdf>.

Carvalho, W. C. D. S. *et al.* (2012) 'A comparative analysis of the agile and traditional software development processes productivity', in *Proceedings - International Conference of the Chilean Computer Science Society, SCCS*. IEEE, pp. 74–82. doi: 10.1109/SCCS.2011.11.

CELLTT.COM (2018) *Magnificent Service Oriented Architecture Example With Architectures For OutSystems Applications | celltt.com*. Available at: <http://celltt.com/lovely-service-oriented-architecture-example/magnificent-service-oriented-architecture-example-with-architectures-for-OutSystems-applications/> (Accessed: 3 November 2018).

COPELAND, M. (2016) *The Difference Between AI, Machine Learning, and Deep Learning? | NVIDIA Blog*. Available at: <https://blogs.nvidia.com/blog/2016/07/29/whats-difference-artificial-intelligence-machine-learning-deep-learning-ai/> (Accessed: 11 December 2018).

Custom Search - Google (2018) *Using REST to Invoke the API | Custom Search | Google Developers*. Available at: [https://developers.google.com/custom-search/v1/using\\_rest](https://developers.google.com/custom-search/v1/using_rest) (Accessed: 6 December 2018).

DELFIM, S. M. (2017) *Diagrama de robustez - Portal Arquiteto*. Available at: <http://www.portalarquiteto.com.br/diagrama-de-robustez/> (Accessed: 6 October 2018).

Dewan, P. (2015) 'Words Versus Pictures: Leveraging the Research on Visual Communication', *Partnership: The Canadian Journal of Library and Information Practice and Research*, 10(1). doi: 10.21083/partnership.v10i1.3137.

Directions Service - Google (2018) *Directions Service | Maps JavaScript API | Google Developers*. Available at: <https://developers.google.com/maps/documentation/javascript/directions> (Accessed: 14 October 2018).

Distance Matrix Service - Google (2018) *Distance Matrix Service | Maps JavaScript API | Google Developers*. Available at: <https://developers.google.com/maps/documentation/javascript/distancematrix> (Accessed: 12 October 2018).

Gamma, E. and Salgado, L. A. M. (2000) *Padr3es de projeto : solu33es reutiliz3aveis de software orientado a objetos*. Grupo A - Bookman. Available at: [https://books.google.pt/books/about/Padr3es\\_de\\_Projetos\\_Solu33es\\_Reutiliz.html?id=U91CYCqTCgkC&source=kp\\_cover&redir\\_esc=y](https://books.google.pt/books/about/Padr3es_de_Projetos_Solu33es_Reutiliz.html?id=U91CYCqTCgkC&source=kp_cover&redir_esc=y) (Accessed: 10 November 2018).

Geocoding API - Google (2018) *Developer Guide | Geocoding API | Google Developers*. Available at: <https://developers.google.com/maps/documentation/geocoding/intro> (Accessed: 14 October 2018).

Geolocation API - Google (no date) *Developer Guide | Geolocation API | Google Developers*. Available at: <https://developers.google.com/maps/documentation/geolocation/intro> (Accessed: 17 October 2018).

Google (2017) *Google Trips*. Available at: <https://get.google.com/trips/> (Accessed: 28 February 2017).

guiaderodas (2017) *guiaderodas*. Available at: <http://www.guiaderodas.com/> (Accessed: 11 February 2017).

IBM Analytics (2018) *Big Data Analytics | IBM Analytics*. Available at: <https://www.ibm.com/analytics/hadoop/big-data-analytics> (Accessed: 16 December 2018).

Jacobson, I. (1992) *Object-oriented software engineering: a use case driven approach*. ACM Press. Available at: [https://books.google.pt/books/about/Object\\_oriented\\_Software\\_Engineering.html?id=A6lQAAAAMAAJ&source=kp\\_cover&redir\\_esc=y](https://books.google.pt/books/about/Object_oriented_Software_Engineering.html?id=A6lQAAAAMAAJ&source=kp_cover&redir_esc=y) (Accessed: 9 November 2018).

*JiTT.travel* (2017). Available at: <https://jitt.travel/pt/lisbon/> (Accessed: 29 January 2017).

Kruchten, P. (2004) *The rational unified process: an introduction*. Addison-Wesley. Available at: [https://books.google.pt/books?id=RYCMx6o47pMC&printsec=frontcover&hl=pt-PT&source=gbs\\_ge\\_summary\\_r&cad=0#v=onepage&q&f=false](https://books.google.pt/books?id=RYCMx6o47pMC&printsec=frontcover&hl=pt-PT&source=gbs_ge_summary_r&cad=0#v=onepage&q&f=false) (Accessed: 17 August 2018).

Maps Embed API - Google (2018) *Developer Guide | Maps Embed API | Google Developers*. Available at: <https://developers.google.com/maps/documentation/embed/guide> (Accessed: 8 September 2018).

Maps JavaScript API - Google (2018) *Overview | Maps JavaScript API | Google Developers*. Available at: <https://developers.google.com/maps/documentation/javascript/tutorial> (Accessed: 9 October 2018).

Menezes, F. and OutSystems (2012) 'The art of designing great Architectures in OutSystems'. Available at: <https://vdocuments.mx/OutSystems-the-art-of-designing-OutSystems-architectures-nextstep-2012.html> (Accessed: 3 November 2018).

Minube (2017) *Minube - Viagens e Turismo - Aplicações Android no Google Play*. Available at: <https://play.google.com/store/apps/details?id=com.minube.app> (Accessed: 12 February 2017).

Observador (2017) *Portugal eleito o "Melhor Destino do Mundo" nos World Travel Awards*. Available at: <https://observador.pt/2017/12/10/portugal-eleito-o-melhor-destino-do-mundo-nos-world-travel-awards/> (Accessed: 10 March 2018).

OutSystems - Asynchronous (2016) *OutSystems Platform Timers and Asynchronous Processes - OutSystems*. Available at: [https://success.OutSystems.com/Support/Enterprise\\_Customers/Maintenance\\_and\\_Operations/OutSystems\\_Platform\\_Timers\\_and\\_Asynchronous\\_Processes](https://success.OutSystems.com/Support/Enterprise_Customers/Maintenance_and_Operations/OutSystems_Platform_Timers_and_Asynchronous_Processes) (Accessed: 13 December 2018).

OutSystems - Processes (2017) *Use Processes - OutSystems*. Available at: [https://success.OutSystems.com/Documentation/10/Developing\\_an\\_Application/Use\\_Processes](https://success.OutSystems.com/Documentation/10/Developing_an_Application/Use_Processes) (Accessed: 13 December 2018).

OutSystems Timers (2017) *Use Timers - OutSystems*. Available at: [https://success.OutSystems.com/Documentation/10/Developing\\_an\\_Application/Use\\_Timers](https://success.OutSystems.com/Documentation/10/Developing_an_Application/Use_Timers) (Accessed: 13 December 2018).

PEREIRA, J. M. (2018) *Expresso | OutSystems é o novo 'unicórnio' português*. Available at: <https://expresso.sapo.pt/economia/2018-06-05-OutSystems-e-o-novo-unicornio-portugues#gs.LWa9ngo> (Accessed: 17 September 2018).

Place Details - Google (2018) *Place Details | Places API | Google Developers*. Available at: <https://developers.google.com/places/web-service/details> (Accessed: 11 October 2018).

Place Photos - Google (no date) *Place Photos | Places API | Google Developers*. Available at: <https://developers.google.com/places/web-service/photos> (Accessed: 9 October 2018).

Place Search - Google (2018) *Place Search | Places API | Google Developers*. Available at: <https://developers.google.com/places/web-service/search> (Accessed: 9 October 2018).

Publituris (2014) *A função da tecnologia de informação na competitividade para o turismo*. Available at: <https://www.publituris.pt/2014/01/13/a-funcao-da-tecnologia-de-informacao-na-competitividade->

para-o-turismo/ (Accessed: 16 September 2018).

Rosenberg, D. and Stephens, M. (2007) *Use case driven object modeling with UML : theory and practice*. Apress. Available at: <https://epdf.tips/use-case-driven-object-modeling-with-uml-theory-and-practice6236166b7d7e73a9629fe1a488e5aa8991670.html> (Accessed: 9 November 2018).

ROSENBERG, D., STEPHENS, M. and COLLINS-COPE, M. (2005) *Agile Development with ICONIX Process—People, Process, and Pragmatism*. Available at: <http://users.nagafix.co.uk/~mark/000.html> (Accessed: 18 May 2018).

Static Entities - OutSystems (2017) *Static Entities - OutSystems*. Available at: [https://success.OutSystems.com/Documentation/10/Developing\\_an\\_Application/Use\\_Data/Data\\_Modeling/Static\\_Entities](https://success.OutSystems.com/Documentation/10/Developing_an_Application/Use_Data/Data_Modeling/Static_Entities) (Accessed: 17 October 2018).

Street View Service - Google (2018) *Street View Service | Maps JavaScript API | Google Developers*. Available at: <https://developers.google.com/maps/documentation/javascript/streetview> (Accessed: 9 October 2018).

Techopedia (2018) *What is Artificial Intelligence (AI)? - Definition*. Available at: <https://www.techopedia.com/definition/190/artificial-intelligence-ai> (Accessed: 16 November 2018).

Tomás, M. R. S. (2009) 'Métodos ágeis: características, pontos fortes e fracos e possibilidades de aplicação'. Available at: [https://run.unl.pt/bitstream/10362/2003/1/WPSeries\\_09\\_2009Tomas.pdf](https://run.unl.pt/bitstream/10362/2003/1/WPSeries_09_2009Tomas.pdf) (Accessed: 1 November 2018).

tripcase (2017) *Using TripCase App: How to Manage Your Itineraries*. Available at: [http://br.tripcase.com/using-tripcase/?utm\\_medium=web&utm\\_source=footer&utm\\_campaign=about&utm\\_content=br](http://br.tripcase.com/using-tripcase/?utm_medium=web&utm_source=footer&utm_campaign=about&utm_content=br) (Accessed: 12 February 2017).

Turismo de Portugal (2018) *TravelBI by Turismo de Portugal*. Available at: [http://www.turismodeportugal.pt/pt/Turismo\\_Portugal/visao\\_geral/Paginas/default.aspx](http://www.turismodeportugal.pt/pt/Turismo_Portugal/visao_geral/Paginas/default.aspx) (Accessed: 16 September 2018).

Vertigo (2017) *O que é API? Entenda de uma maneira simples*. Available at: <https://vertigo.com.br/o-que-e-api-entenda-de-uma-maneira-simples/> (Accessed: 17 December 2018).

## 9. Anexos

### I. Tabelas usadas

#### User

Campo	Tipo	Descrição	Observações
<b><i>Id</i></b>	<i>Long Integer</i>	Identificador único do <i>user</i>	Chave primária
<b><i>Name</i></b>	<i>Text</i>	Nome	
<b><i>Username</i></b>	<i>Text</i>	Nome de utilizador	
<b><i>Email</i></b>	<i>Email</i>	<i>Email</i>	
<b><i>PinCode</i></b>	<i>Text</i>	Código pin de acesso a aplicação	
<b><i>IsActive</i></b>	<i>Boolean</i>	Se verdadeiro, o utilizador está ativo	

#### ExternalUser

Campo	Tipo	Descrição	Observações
<b><i>Id</i></b>	<i>Long Integer</i>	Identificador do utilizador externo	Chave primária
<b><i>UserId</i></b>	<i>UserIdentifier</i>	Identificador do utilizador	Chave externa para a entidade <i>User</i>
<b><i>PictureId</i></b>	<i>PictureIdentifier</i>	Identificador da imagem	Chave externa para a entidade <i>Picture</i>
<b><i>RoleLevelId</i></b>	<i>RoleIdentifier</i>	Identificador da função	Chave externa para a entidade <i>RoleLevel</i>
<b><i>IsFromServer</i></b>	<i>Boolean</i>	Se verdadeiro, o utilizador existe no servidor	Atributo para sincronização com o servidor
<b><i>IsModified</i></b>	<i>Boolean</i>	Se verdadeiro, o utilizador foi alterado localmente	Atributo para sincronização com o servidor
<b><i>IsActive</i></b>	<i>Boolean</i>	Se verdadeiro, o utilizador está ativo	Atributo para sincronização com o servidor

#### Trip

Campo	Tipo	Descrição	Observações
<b><i>Id</i></b>	<i>Long Integer</i>	Identificador o da viagem	Chave primária
<b><i>PictureId</i></b>	<i>PictureIdentifier</i>	Identificador da imagem	Chave externa para a entidade <i>Picture</i>
<b><i>Description</i></b>	<i>Text</i>	Descrição do local da viagem	
<b><i>StartDate</i></b>	<i>Date</i>	Data de início da viagem	

<b>EndDate</b>	<i>Date</i>	Data de fim da viagem	
<b>StatusId</b>	<i>StateIdentifier</i>	Identificador do estado	Chave externa para a entidade <i>State</i>
<b>Name</b>	<i>Text</i>	Nome da viagem	
<b>IsFromServer</b>	<i>Boolean</i>	Se verdadeiro, a viagem existe no servidor	Atributo para sincronização com o servidor
<b>IsModified</b>	<i>Boolean</i>	Se verdadeiro, a viagem foi alterada localmente	Atributo para sincronização com o servidor
<b>IsActive</b>	<i>Boolean</i>	Se verdadeiro, a viagem não foi removida localmente	Atributo para sincronização com o servidor

**Plans**

<b>Campo</b>	<b>Tipo</b>	<b>Descrição</b>	<b>Observações</b>
<b>Id</b>	<i>Long Integer</i>	Identificador da relação	Chave primária
<b>UserId</b>	<i>UserIdentifier</i>	Identificador do utilizador	Chave externa para a entidade <i>User</i>
<b>EventId</b>	<i>EventIdentifier</i>	Identificador do evento	Chave externa para a entidade <i>Event</i>
<b>IsFromServer</b>	<i>Boolean</i>	Se verdadeiro, a relação existe no servidor	Atributo para sincronização com o servidor
<b>IsModified</b>	<i>Boolean</i>	Se verdadeiro, a relação foi alterada localmente	Atributo para sincronização com o servidor
<b>IsActive</b>	<i>Boolean</i>	Se verdadeiro, a relação não foi removida localmente	Atributo para sincronização com o servidor

**TripPlans**

<b>Campo</b>	<b>Tipo</b>	<b>Descrição</b>	<b>Observações</b>
<b>Id</b>	<i>Long Integer</i>	Identificador da relação	Chave primária
<b>TripId</b>	<i>TripIdentifier</i>	Identificador da viagem	Chave externa para a entidade <i>Trip</i>
<b>PlanId</b>	<i>PlanIdentifier</i>	Identificador do plano	Chave externa para a entidade <i>Plan</i>

<b><i>IsFromServer</i></b>	<i>Boolean</i>	Se verdadeiro, a relação existe no servidor	Atributo para sincronização com o servidor
<b><i>IsModified</i></b>	<i>Boolean</i>	Se verdadeiro, a relação foi alterada localmente	Atributo para sincronização com o servidor
<b><i>IsActive</i></b>	<i>Boolean</i>	Se verdadeiro, a relação não foi removida localmente	Atributo para sincronização com o servidor

**EventTrip**

<b>Campo</b>	<b>Tipo</b>	<b>Descrição</b>	<b>Observações</b>
<b><i>Id</i></b>	<i>Long Integer</i>	Identificador da relação	Chave primária
<b><i>TripId</i></b>	<i>TripIdentifier</i>	Identificador da viagem	Chave externa para a entidade <i>Trip</i>
<b><i>EventId</i></b>	<i>EventIdentifier</i>	Identificador do evento	Chave externa para a entidade <i>Event</i>
<b><i>IsFromServer</i></b>	<i>Boolean</i>	Se verdadeiro, a relação existe no servidor	Atributo para sincronização com o servidor
<b><i>IsModified</i></b>	<i>Boolean</i>	Se verdadeiro, a relação foi alterada localmente	Atributo para sincronização com o servidor
<b><i>IsActive</i></b>	<i>Boolean</i>	Se verdadeiro, a relação não foi removida localmente	Atributo para sincronização com o servidor

**Rating**

<b>Campo</b>	<b>Tipo</b>	<b>Descrição</b>	<b>Observações</b>
<b><i>Id</i></b>	<i>Long Integer</i>	Identificador da avaliação	Chave primária
<b><i>UserId</i></b>	<i>UserIdentifier</i>	Identificador do utilizador	Chave externa para a entidade <i>User</i>
<b><i>EventId</i></b>	<i>EventIdentifier</i>	Identificador do evento	Chave externa para a entidade <i>Event</i>
<b><i>Value</i></b>		Valor da avaliação	
<b><i>IsFromServer</i></b>	<i>Boolean</i>	Se verdadeiro, a avaliação existe no servidor	Atributo para sincronização com o servidor

<b><i>IsModified</i></b>	<i>Boolean</i>	Se verdadeiro, a avaliação foi alterada localmente	Atributo para sincronização com o servidor
<b><i>IsActive</i></b>	<i>Boolean</i>	Se verdadeiro, a avaliação não foi removida localmente	Atributo para sincronização com o servidor

**Reviews**

<b>Campo</b>	<b>Tipo</b>	<b>Descrição</b>	<b>Observações</b>
<b><i>Id</i></b>	<i>Long Integer</i>	Identificador do percurso	Chave primária
<b><i>UserId</i></b>	<i>UserIdentifier</i>	Identificador do utilizador	Chave externa para a entidade <i>User</i>
<b><i>EventId</i></b>	<i>EventIdentifier</i>	Identificador do evento	Chave externa para a entidade <i>Event</i>
<b><i>Comment</i></b>	<i>Text</i>	Texto inserido pelo utilizador no evento	
<b><i>IsFromServer</i></b>	<i>Boolean</i>	Se verdadeiro, a relação existe no servidor	Atributo para sincronização com o servidor
<b><i>IsModified</i></b>	<i>Boolean</i>	Se verdadeiro, a relação foi alterada localmente	Atributo para sincronização com o servidor
<b><i>IsActive</i></b>	<i>Boolean</i>	Se verdadeiro, a relação não foi removida localmente	Atributo para sincronização com o servidor

**Tag**

<b>Campo</b>	<b>Tipo</b>	<b>Descrição</b>	<b>Observações</b>
<b><i>Id</i></b>	<i>Long Integer</i>	Identificador da <i>tag</i>	Chave primária
<b><i>Label</i></b>	<i>Text</i>	Label da <i>tag</i>	
<b><i>Description</i></b>	<i>Text</i>	Descrição da <i>tag</i>	
<b><i>IsFromServer</i></b>	<i>Boolean</i>	Se verdadeiro, a <i>tag</i> existe no servidor	Atributo para sincronização com o servidor
<b><i>IsModified</i></b>	<i>Boolean</i>	Se verdadeiro, a <i>tag</i> foi alterada localmente	Atributo para sincronização com o servidor
<b><i>IsActive</i></b>	<i>Boolean</i>	Se verdadeiro, a <i>tag</i> não foi	Atributo para

		removida localmente	sincronização com o servidor
--	--	---------------------	------------------------------

**EventTag**

Campo	Tipo	Descrição	Observações
<b><i>Id</i></b>	<i>Long Integer</i>	Identificador da relação	Chave primária
<b><i>EventId</i></b>	<i>EventIdentifier</i>	Identificador do evento	Chave externa para a entidade <i>Event</i>
<b><i>TagId</i></b>	<i>TagIdentifier</i>	Identificador da <i>Tag</i>	Chave externa para a entidade <i>Tag</i>
<b><i>IsMain</i></b>	<i>Boolean</i>	Se verdadeiro, significa que é a categoria chave do evento	
<b><i>IsFromServer</i></b>	<i>Boolean</i>	Se verdadeiro, a relação existe no servidor	Atributo para sincronização com o servidor
<b><i>IsModified</i></b>	<i>Boolean</i>	Se verdadeiro, a relação foi alterada localmente	Atributo para sincronização com o servidor
<b><i>IsActive</i></b>	<i>Boolean</i>	Se verdadeiro, a relação não foi removida localmente	Atributo para sincronização com o servidor

**EventPicture**

Campo	Tipo	Descrição	Observações
<b><i>Id</i></b>	<i>Long Integer</i>	Identificador da imagem	Chave primária
<b><i>EventId</i></b>	<i>EventIdentifier</i>	Identificador do evento	Chave externa para a entidade <i>Event</i>
<b><i>Filename</i></b>	<i>Text</i>	Nome do ficheiro	
<b><i>Picture</i></b>	<i>BinaryData</i>	Imagem	
<b><i>EntryDateTime</i></b>	<i>DateTime</i>	Data de adição da imagem	
<b><i>IsFromServer</i></b>	<i>Boolean</i>	Se verdadeiro, a imagem existe no servidor	Atributo para sincronização com o servidor
<b><i>IsModified</i></b>	<i>Boolean</i>	Se verdadeiro, a imagem foi alterada localmente	Atributo para sincronização com o servidor

<b><i>IsActive</i></b>	<i>Boolean</i>	Se verdadeiro, a imagem não foi removida localmente	Atributo para sincronização com o servidor
------------------------	----------------	---	--

**Followers**

<b>Campo</b>	<b>Tipo</b>	<b>Descrição</b>	<b>Observações</b>
<b><i>Id</i></b>	<i>Long Integer</i>	Identificador da relação	Chave primária
<b><i>Follower</i></b>	<i>UserIdentifier</i>	Identificador do utilizador	Chave externa para a entidade <i>User</i>
<b><i>Followed</i></b>	<i>UserIdentifier</i>	Identificador do utilizador	Chave externa para a entidade <i>User</i>
<b><i>EntryDate</i></b>	<i>DateTime</i>	Data e hora da relação	
<b><i>IsFromServer</i></b>	<i>Boolean</i>	Se verdadeiro, a relação existe no servidor	Atributo para sincronização com o servidor
<b><i>IsModified</i></b>	<i>Boolean</i>	Se verdadeiro, a relação foi alterada localmente	Atributo para sincronização com o servidor
<b><i>IsActive</i></b>	<i>Boolean</i>	Se verdadeiro, a relação não foi removida localmente	Atributo para sincronização com o servidor

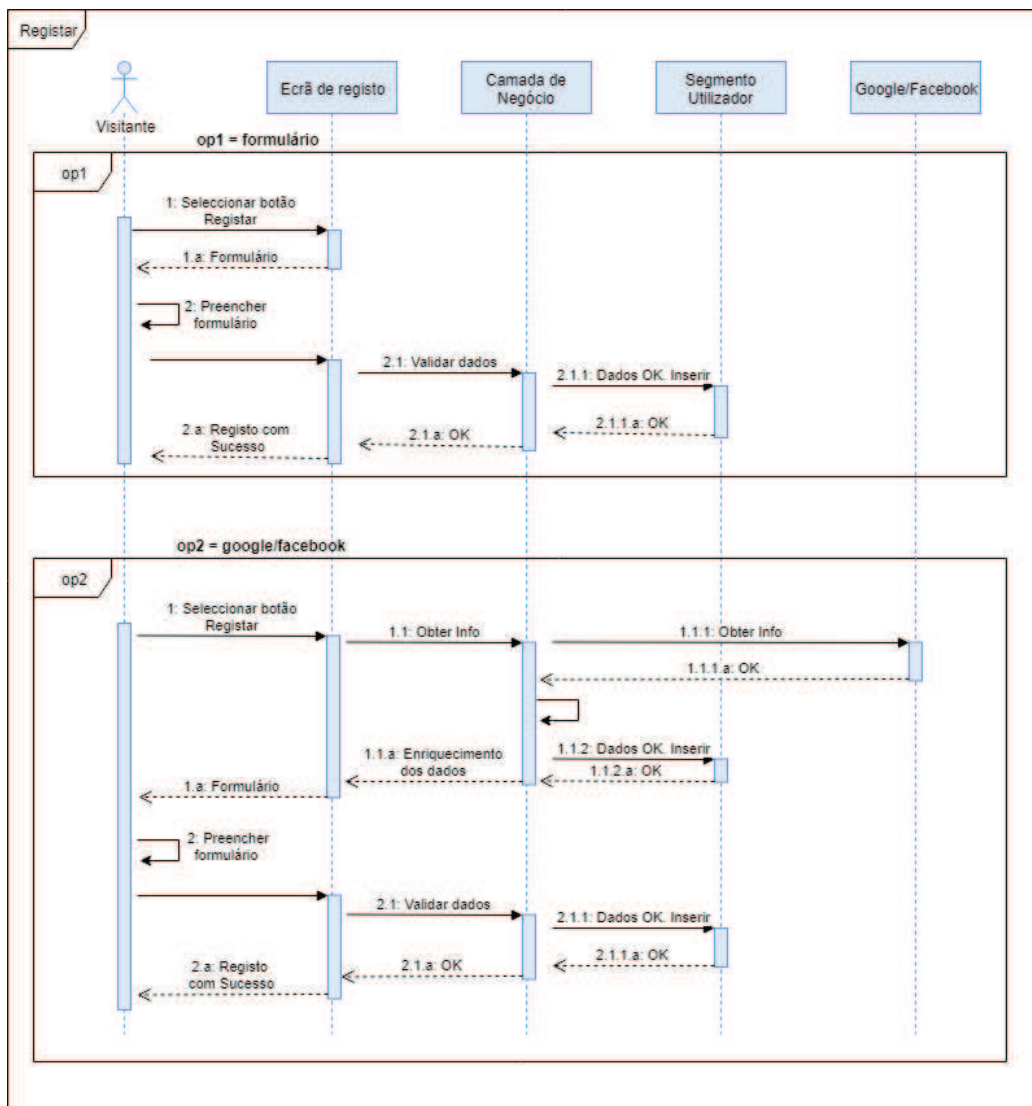
**Messages**

<b>Campo</b>	<b>Tipo</b>	<b>Descrição</b>	<b>Observações</b>
<b><i>Id</i></b>	<i>Long Integer</i>	Identificador da relação	Chave primária
<b><i>UserSent</i></b>	<i>UserIdentifier</i>	Identificador do utilizador	Chave externa para a entidade <i>User</i>
<b><i>UserReceived</i></b>	<i>UserIdentifier</i>	Identificador do utilizador	Chave externa para a entidade <i>User</i>
<b><i>EventId</i></b>	<i>EventIdentifier</i>	Identificador do evento	Chave externa para a entidade <i>Event</i>
<b><i>Message</i></b>	<i>Text</i>	Texto da mensagem	
<b><i>EntryDateTime</i></b>	<i>DateTime</i>	Data e hora do envio da mensagem	
<b><i>IsFromServer</i></b>	<i>Boolean</i>	Se verdadeiro, a relação existe no servidor	Atributo para sincronização com o servidor

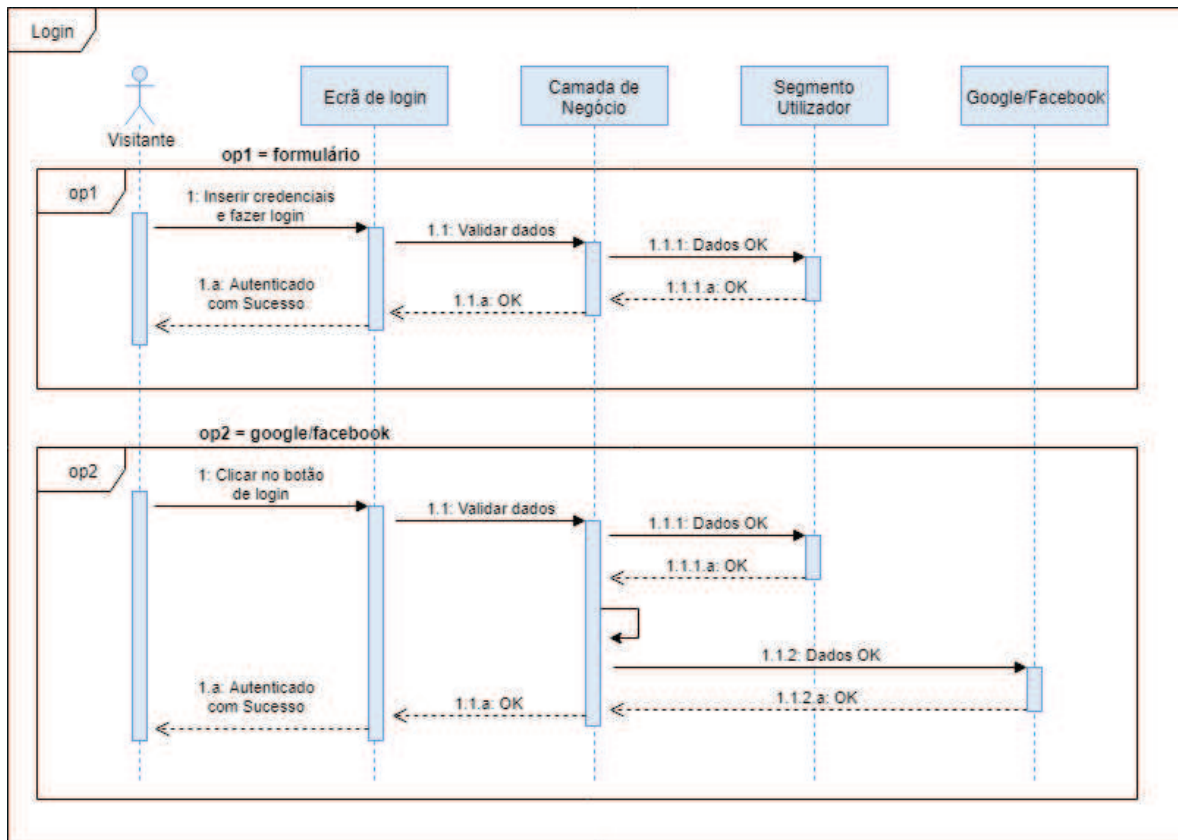
<b><i>IsModified</i></b>	<i>Boolean</i>	Se verdadeiro, a relação foi alterada localmente	Atributo para sincronização com o servidor
<b><i>IsActive</i></b>	<i>Boolean</i>	Se verdadeiro, a relação não foi removida localmente	Atributo para sincronização com o servidor

## II. Diagramas de Sequência

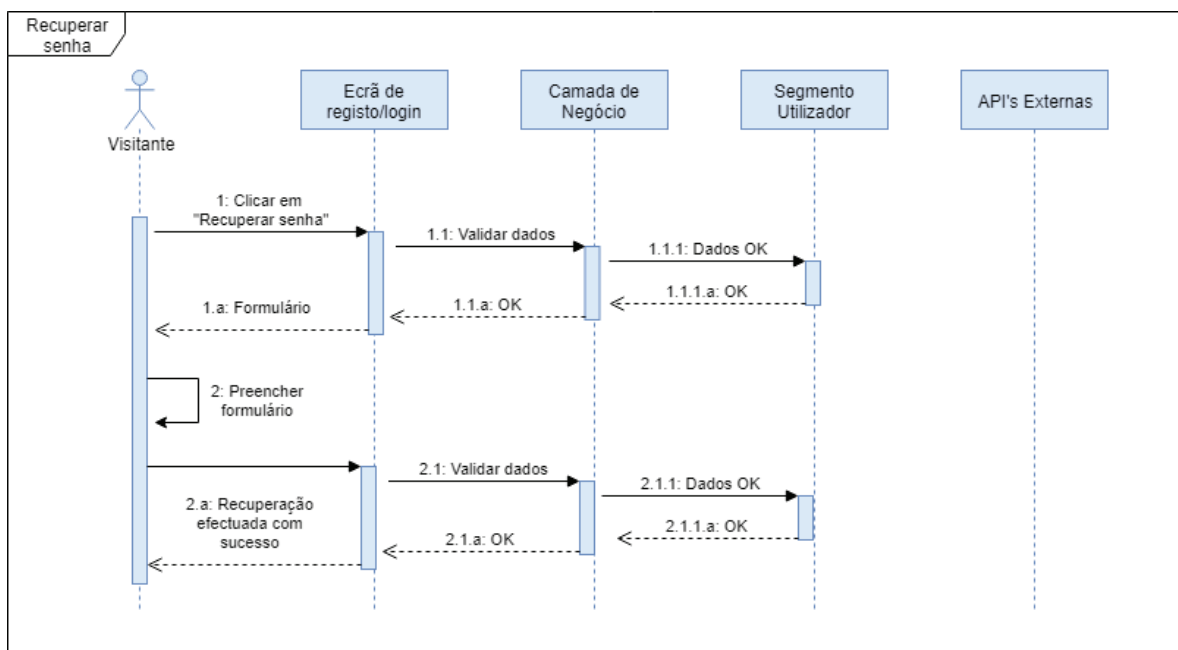
### Diagrama de Sequência Registrar



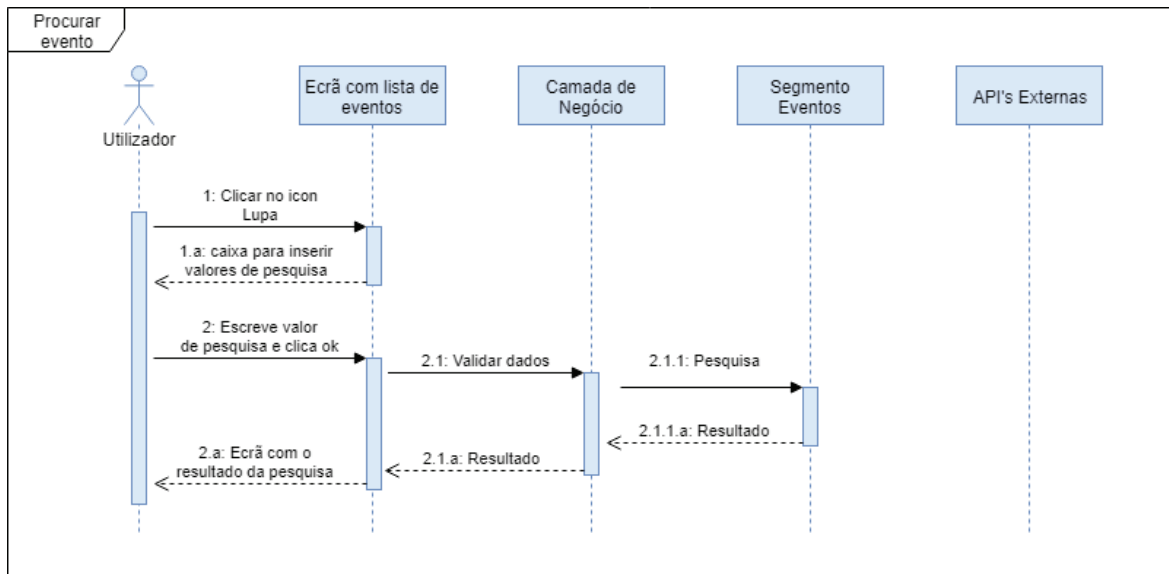
### Diagrama de Sequência Login



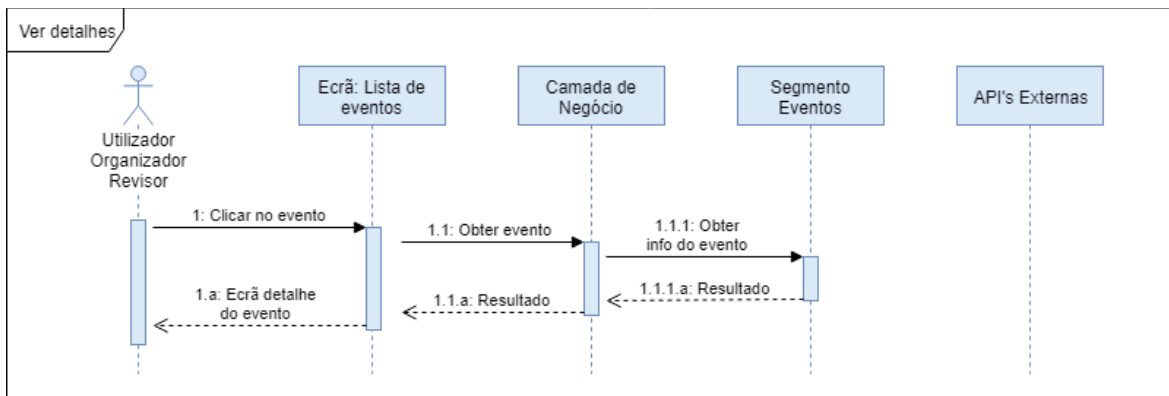
### Diagrama de Sequência Recuperar Palavra-Passe



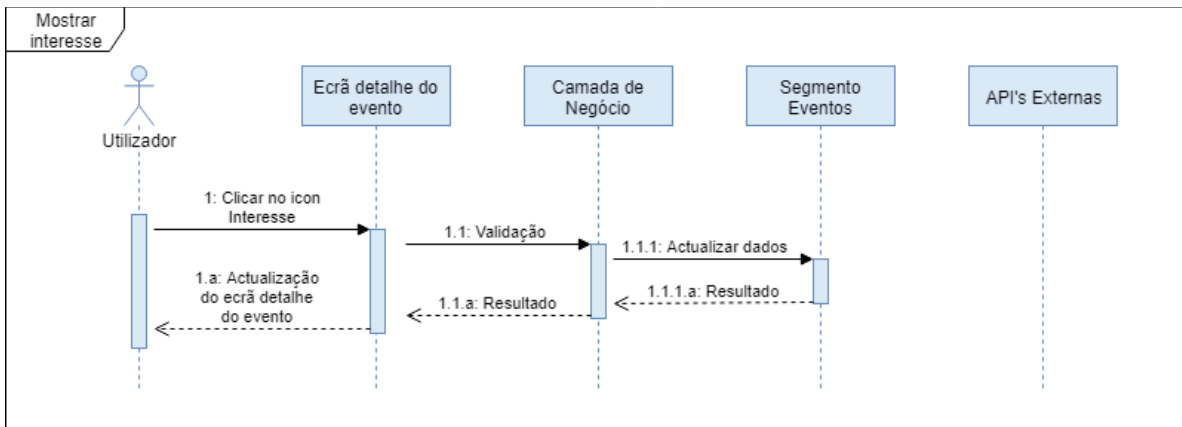
## Diagrama de Sequência Procurar Evento



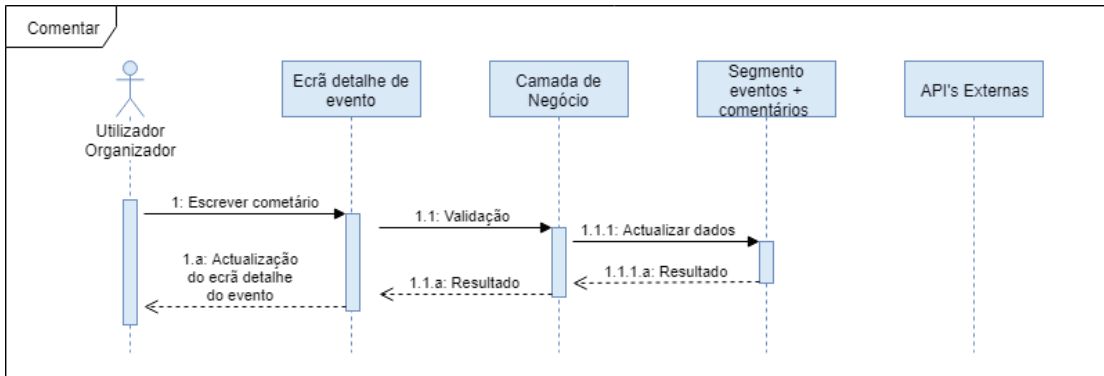
## Diagrama de Sequência Ver Detalhes



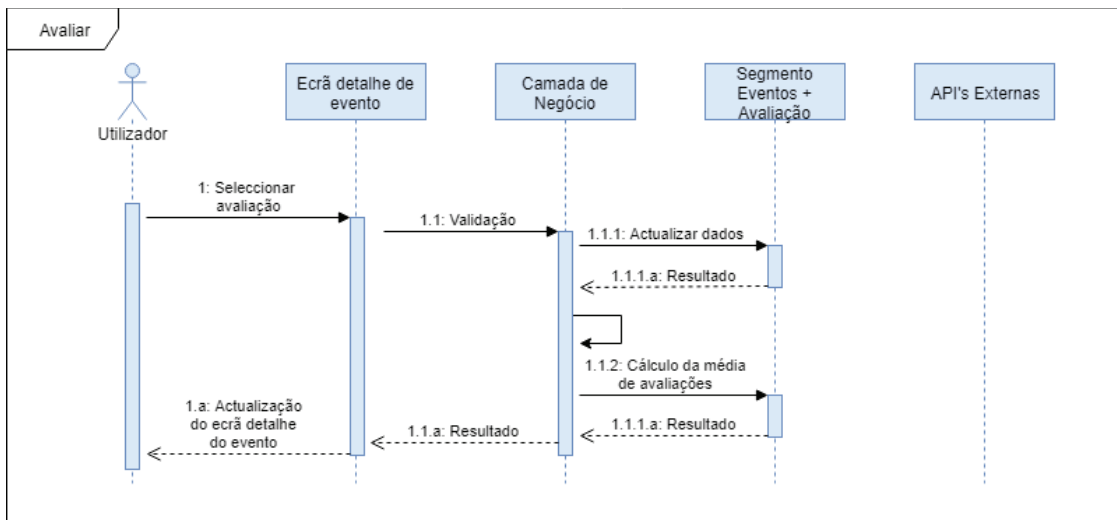
## Diagrama de Sequência Mostrar Interesse



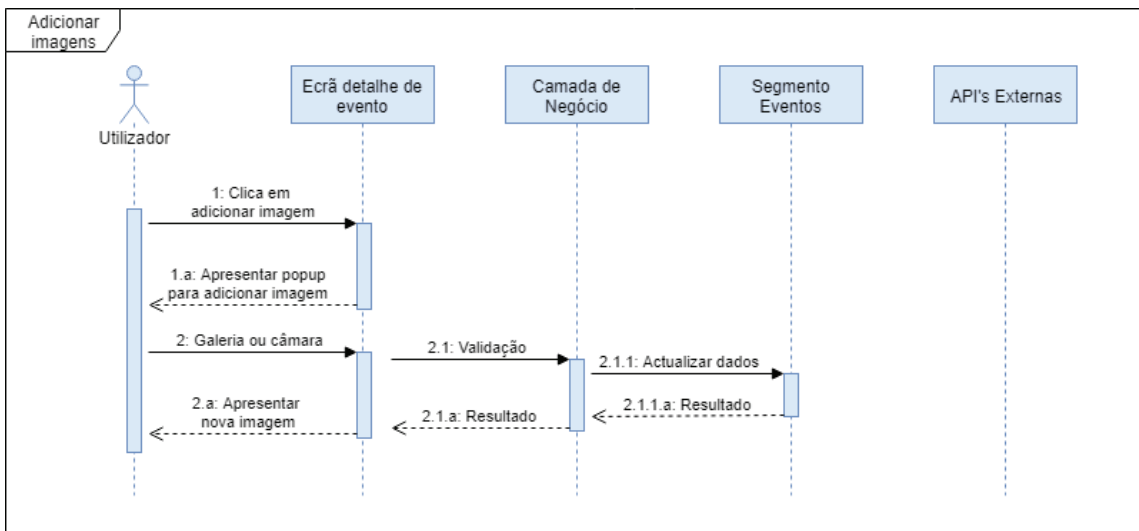
### Diagrama de Sequência Comentar



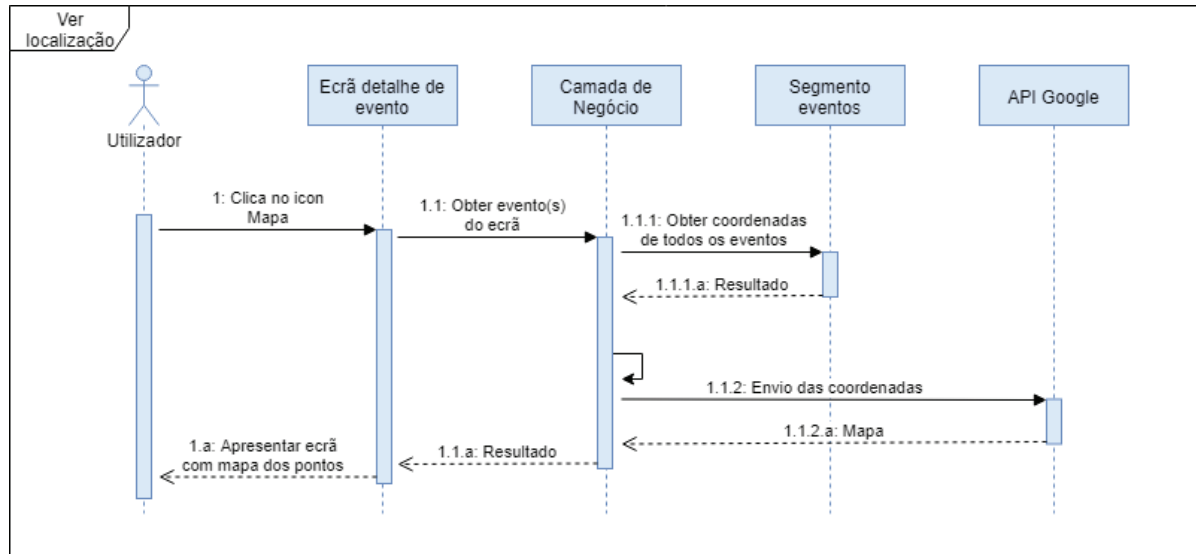
### Diagrama de Sequência Avaliar



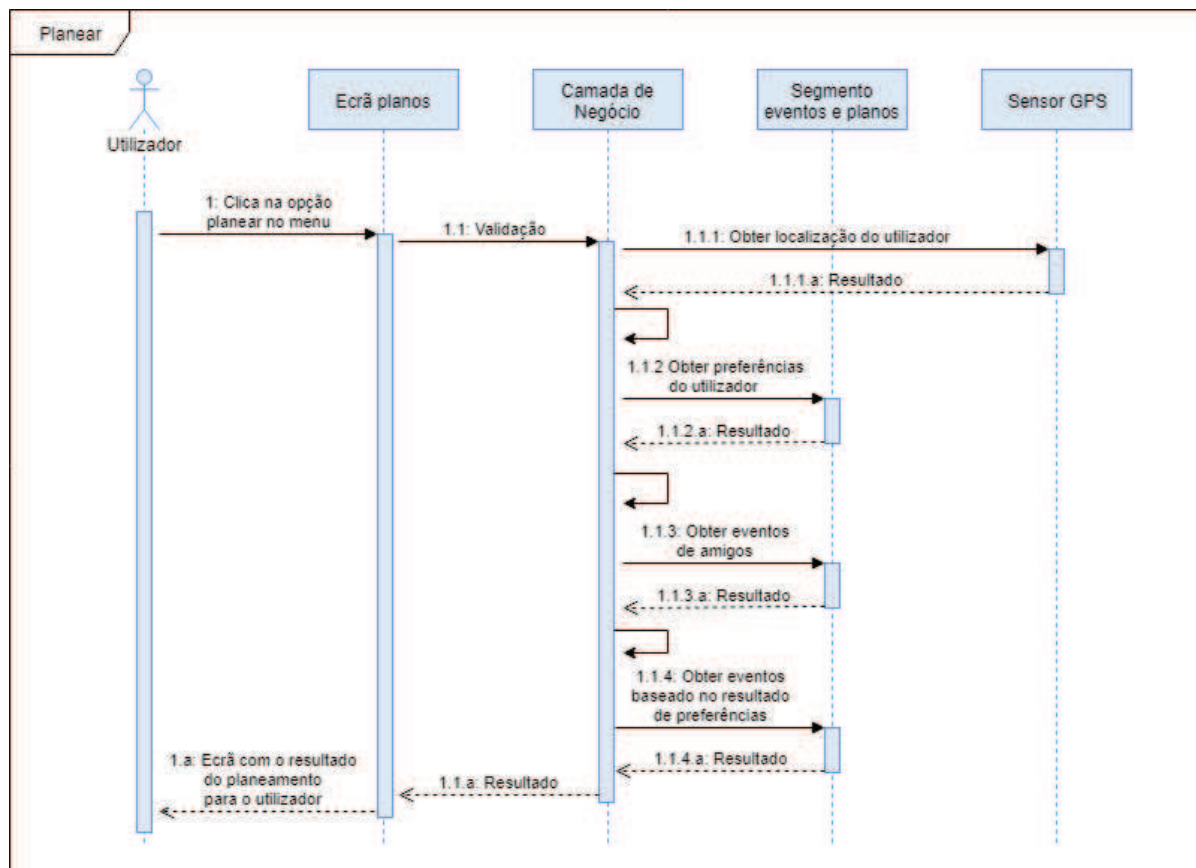
### Diagrama de Sequência Adicionar Imagens



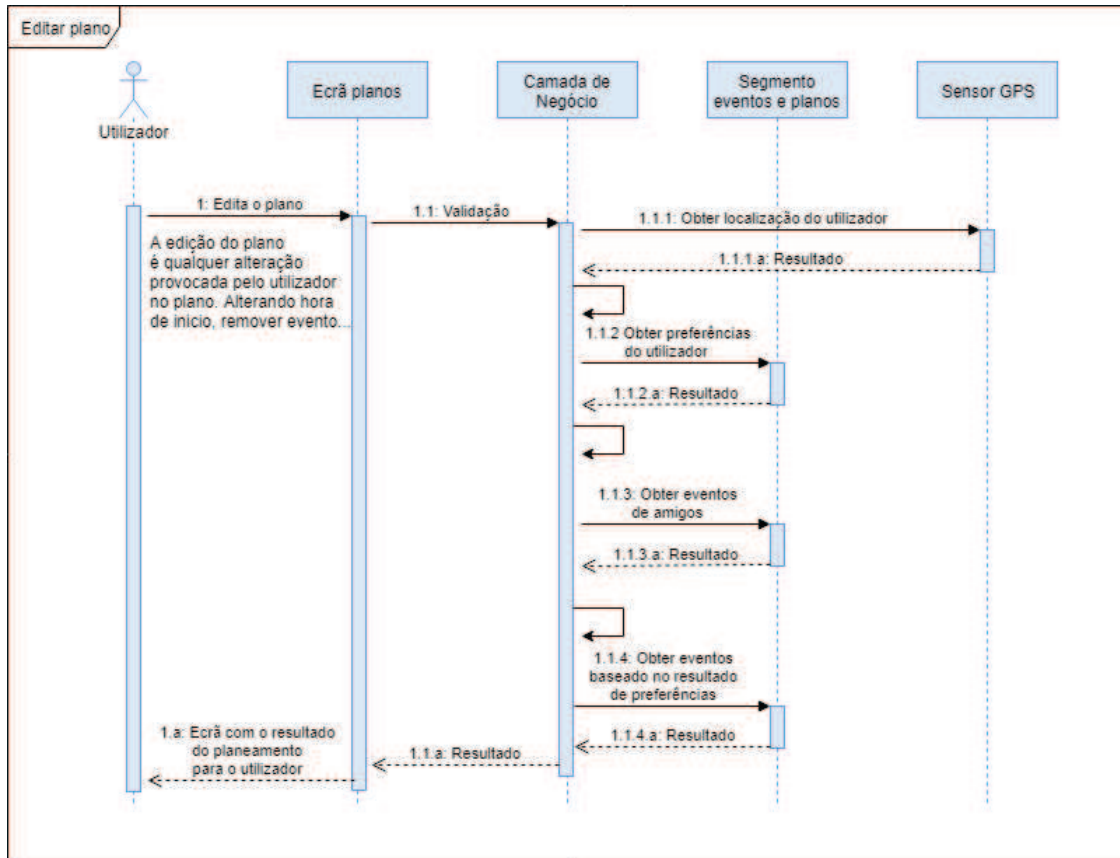
### Diagrama de Sequência Ver Localização



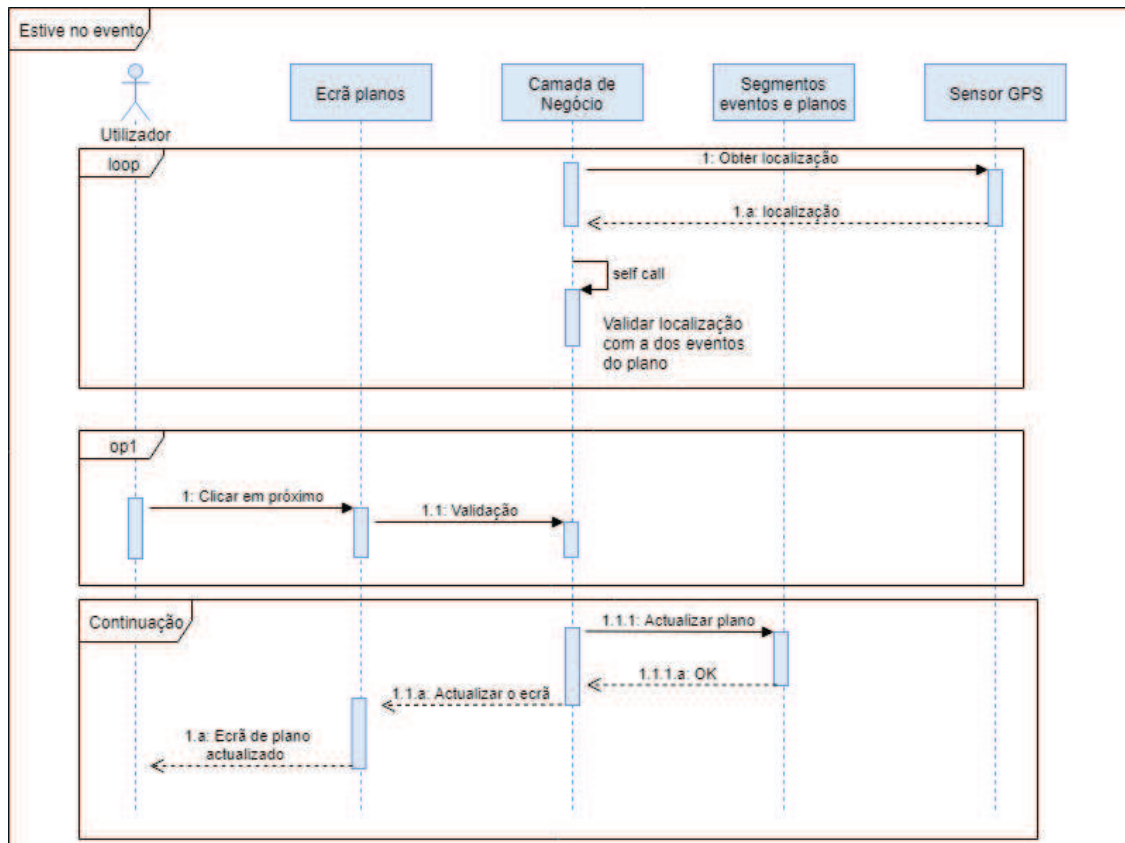
### Diagrama de Sequência Planear



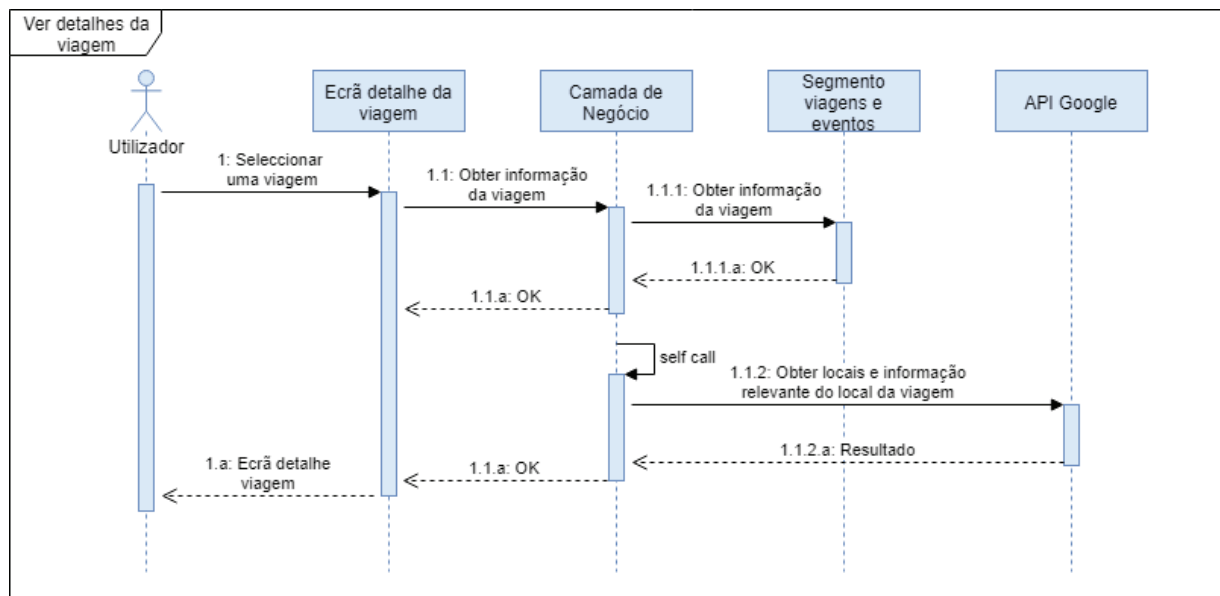
### Diagrama de Sequência Editar Plano



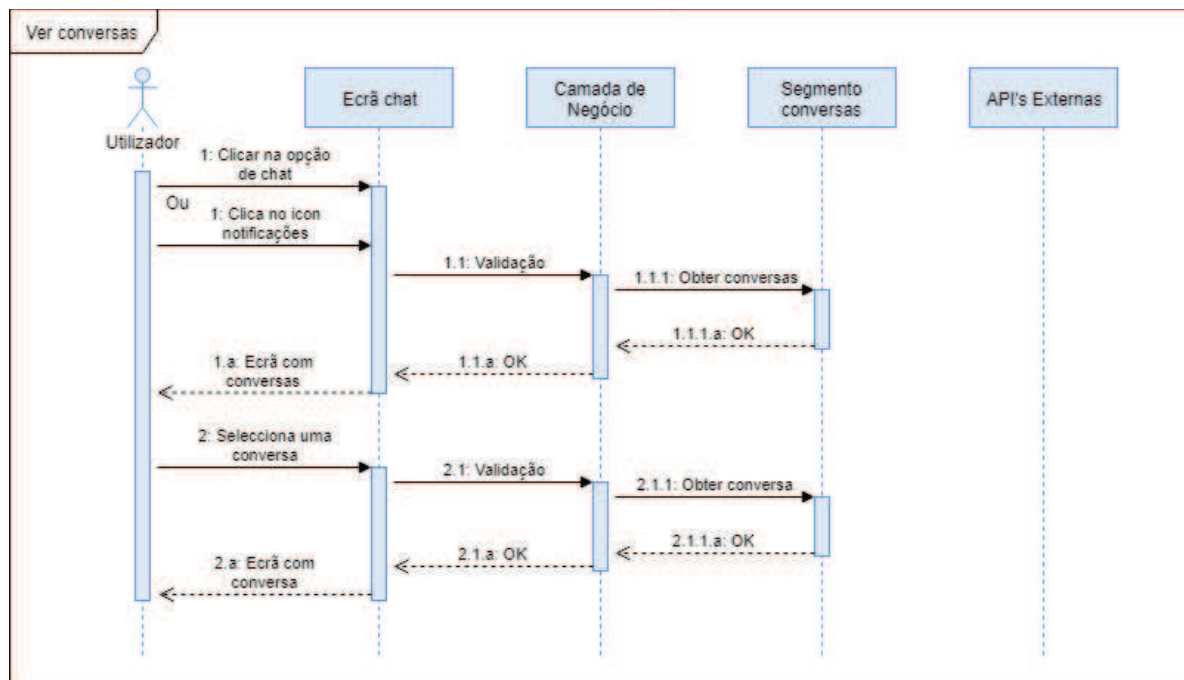
### Diagrama de Sequência Estive no Plano



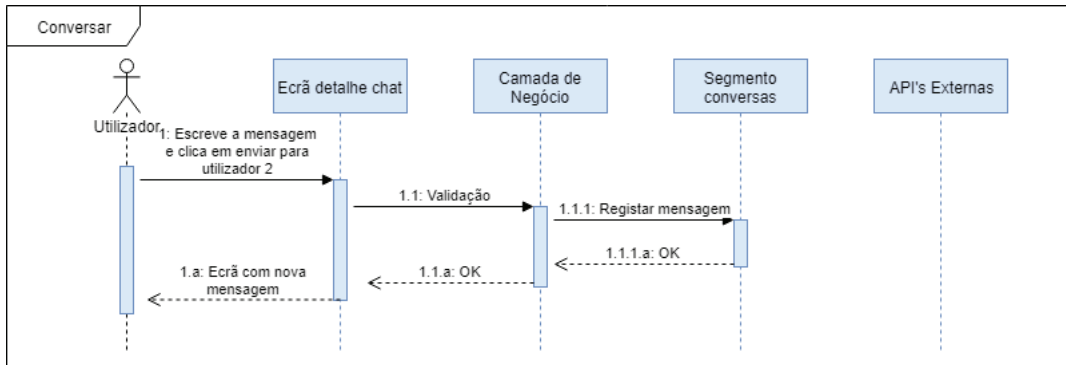
### Diagrama de Sequência Ver Detalhes da Viagem



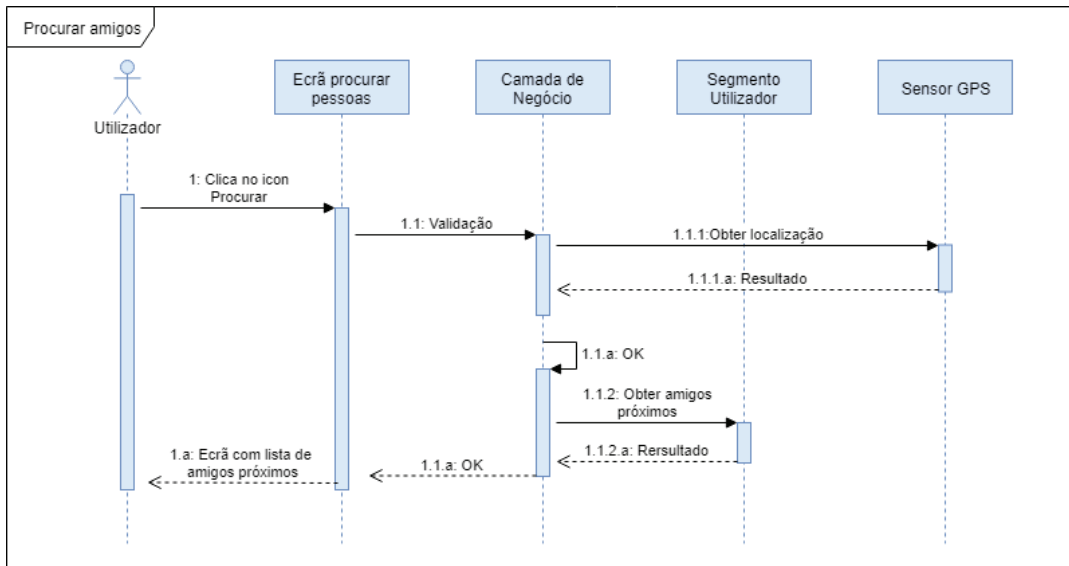
### Diagrama de Sequência Ver Conversas



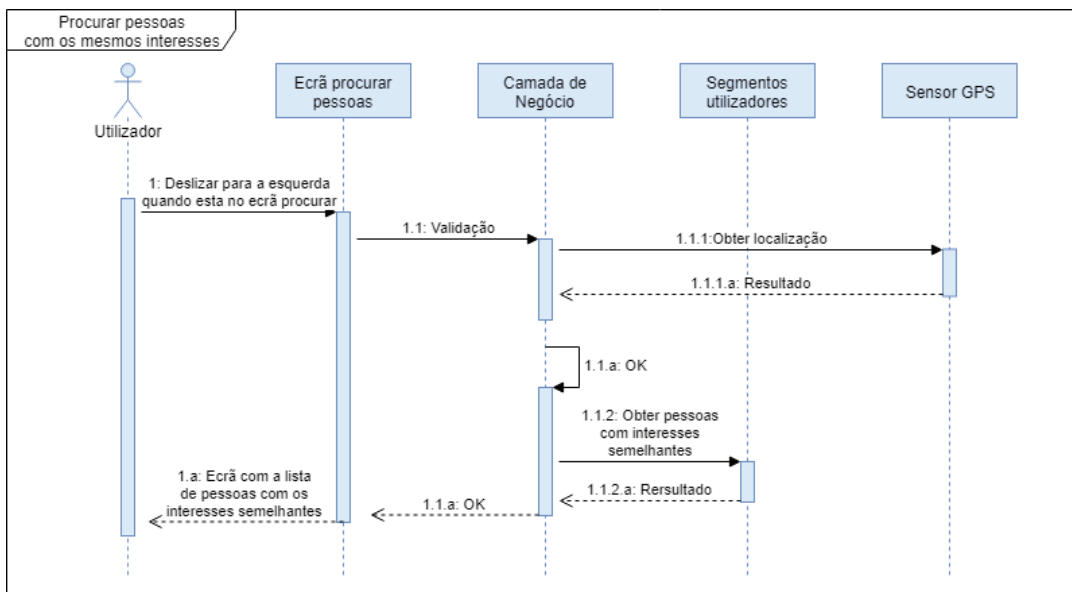
### Diagrama de Sequência Conversar



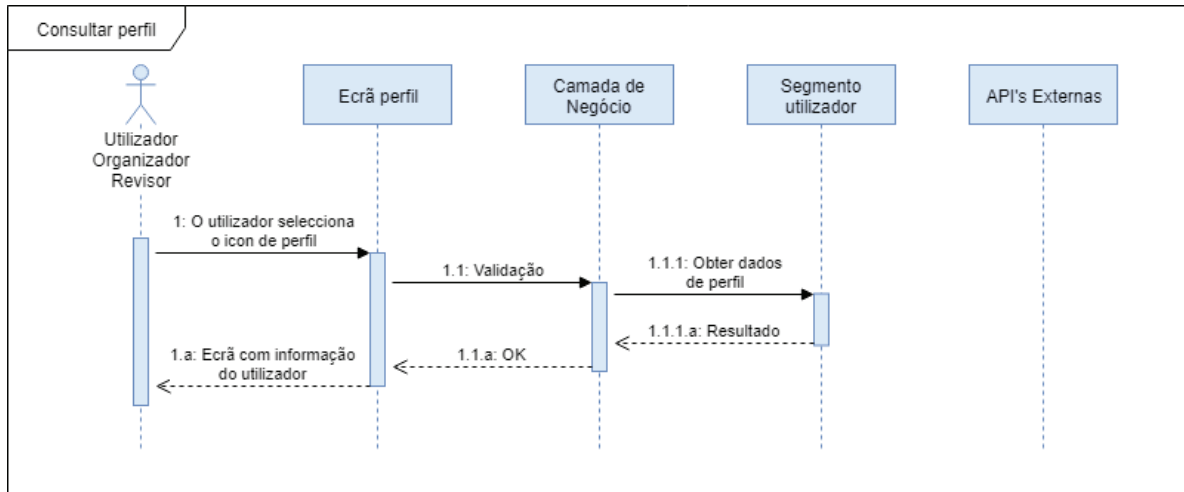
### Diagrama de Sequência Procurar Utilizadores Seguidos



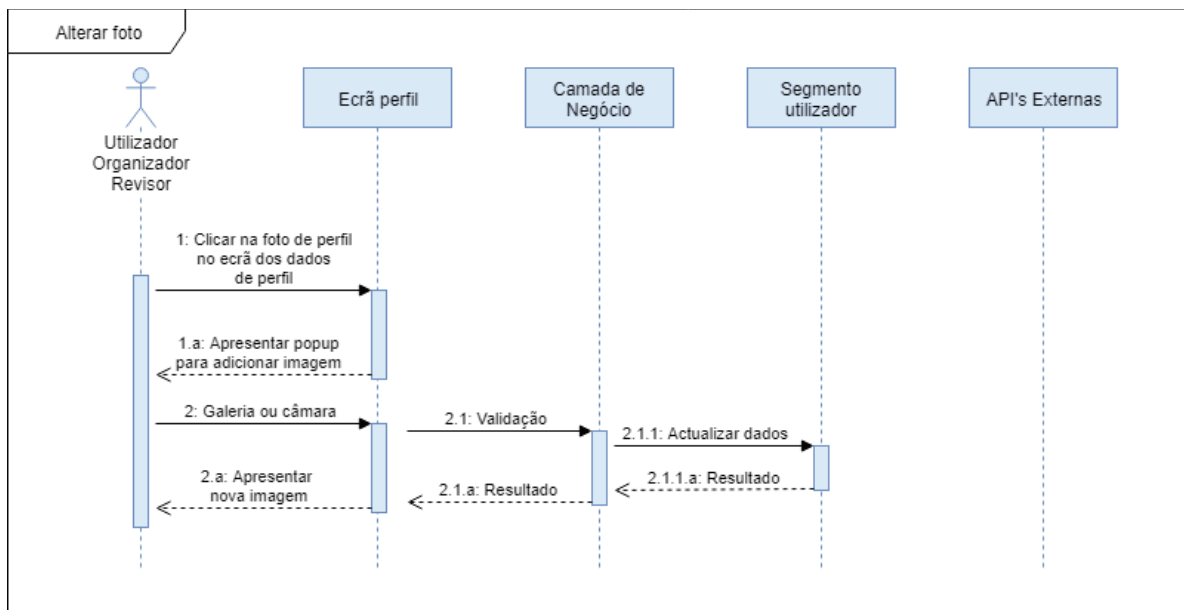
### Diagrama de Sequência Procurar Utilizadores com os Mesmos Interesses



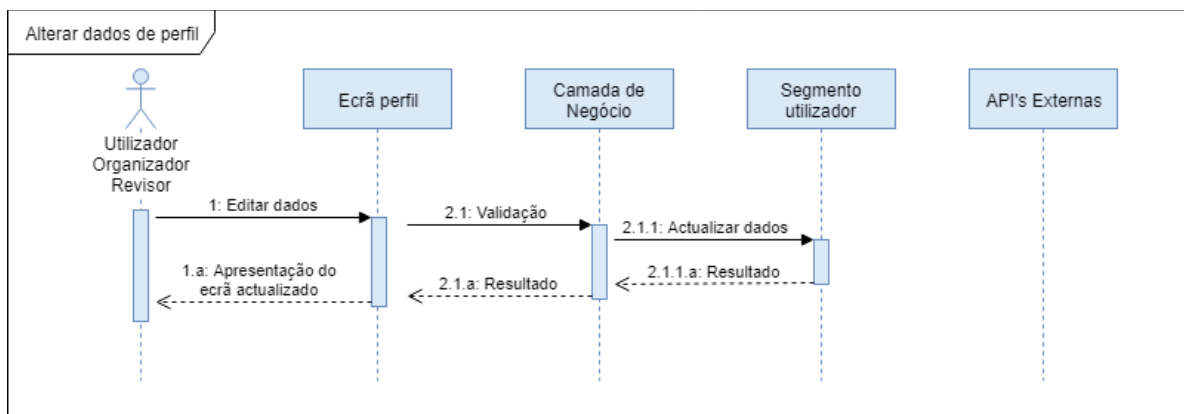
### Diagrama de Sequência Consultar Perfil



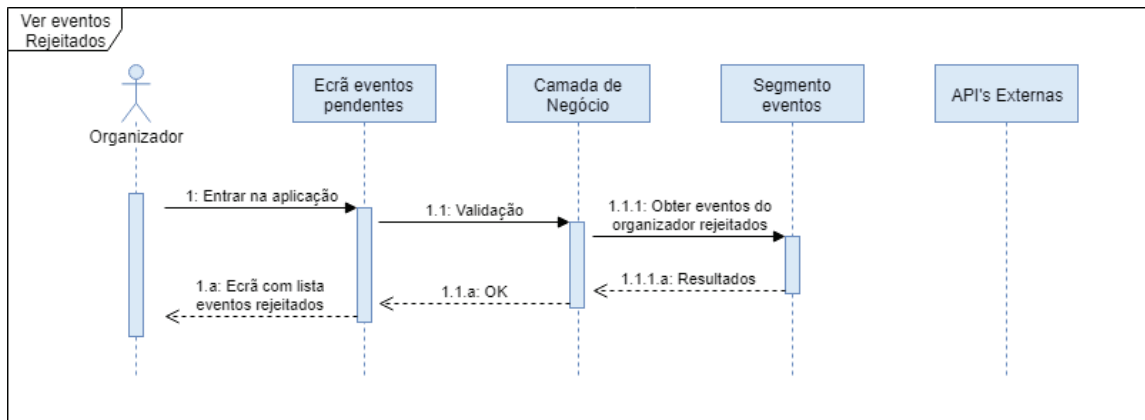
### Diagrama de Sequência Alterar Foto



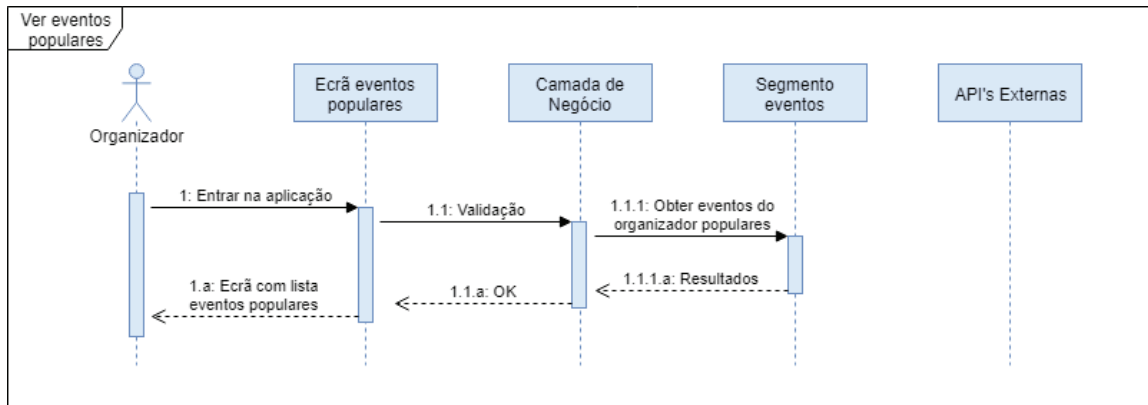
### Diagrama de Sequência Alterar Dados de Perfil



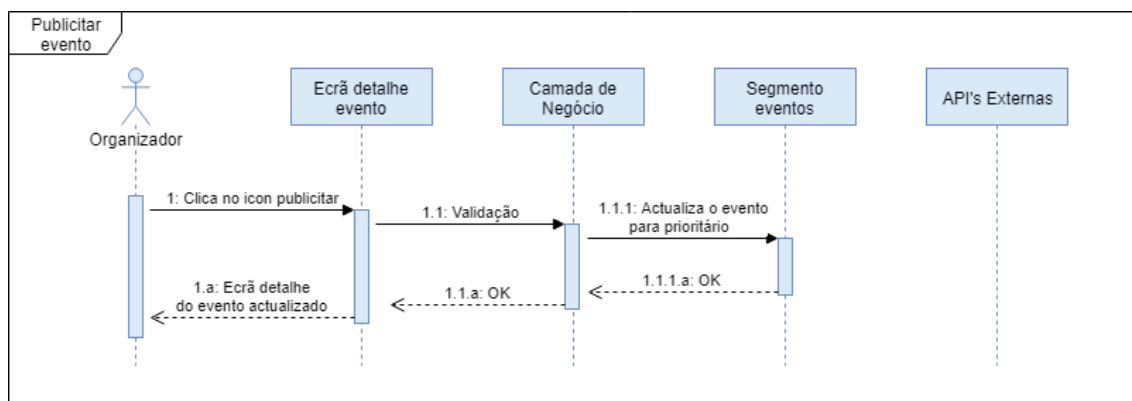
### Diagrama de Sequência Ver Eventos Pendentes



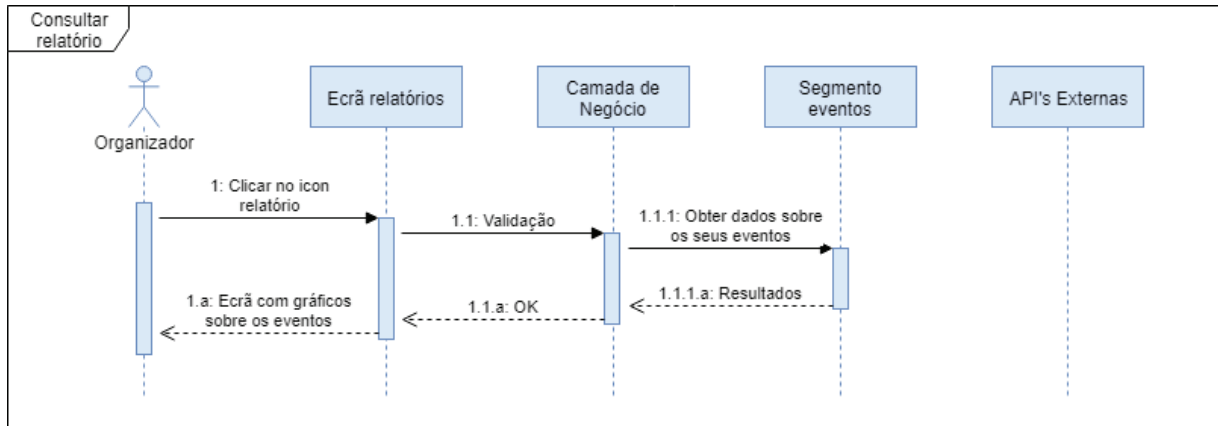
### Diagrama de Sequência Ver Eventos Populares



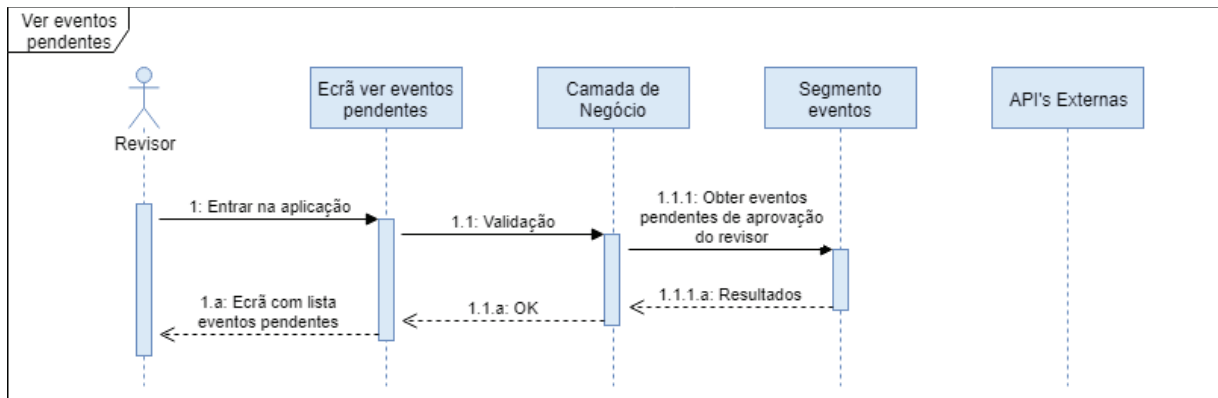
### Diagrama de Sequência Publicitar Evento



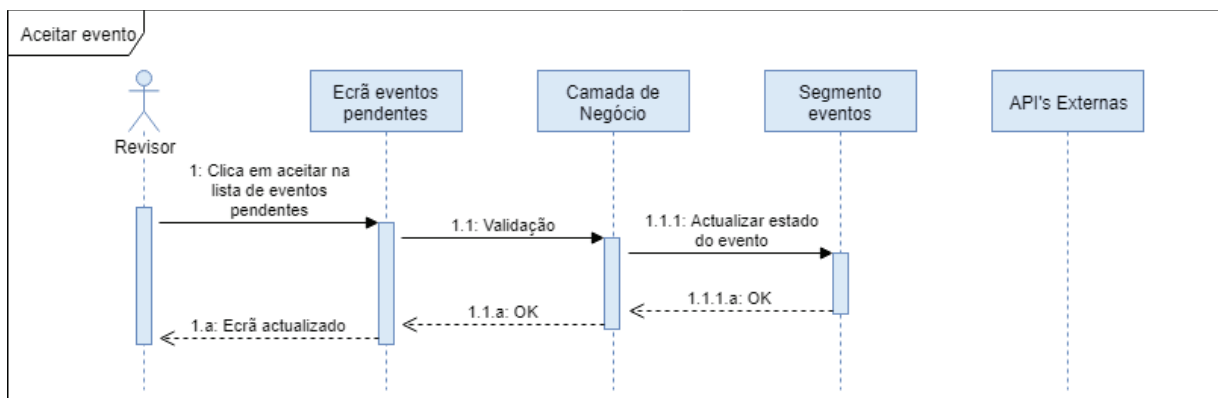
### Diagrama de Sequência Consultar Relatório



### Diagrama de Sequência Ver Eventos Pendentes de Aprovação

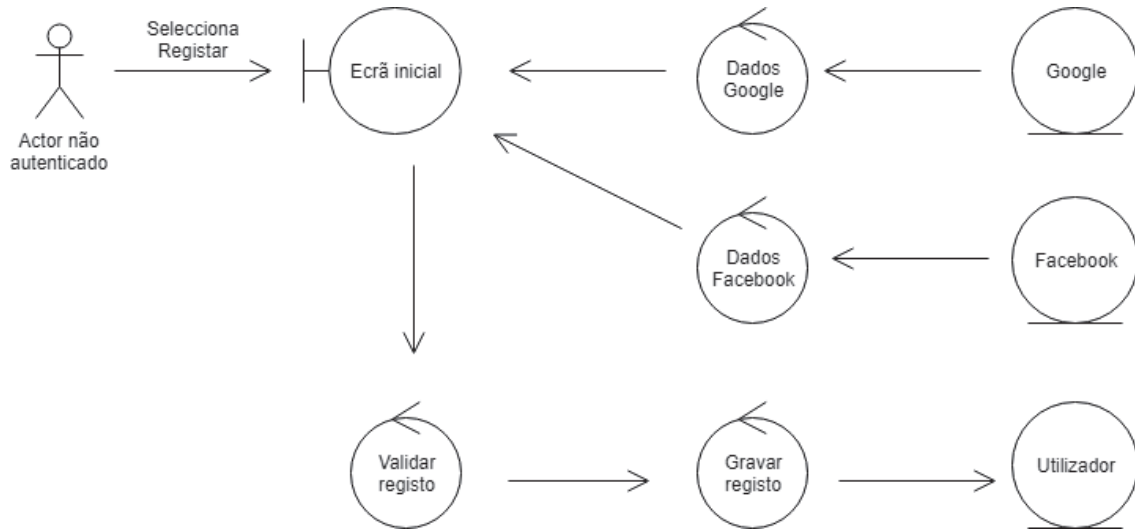


### Diagrama de Sequência Aprovar Evento

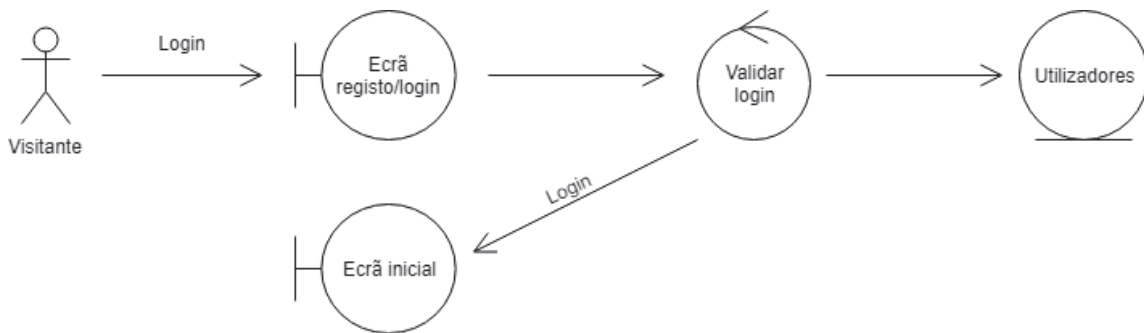


### III. Diagramas de Robustez

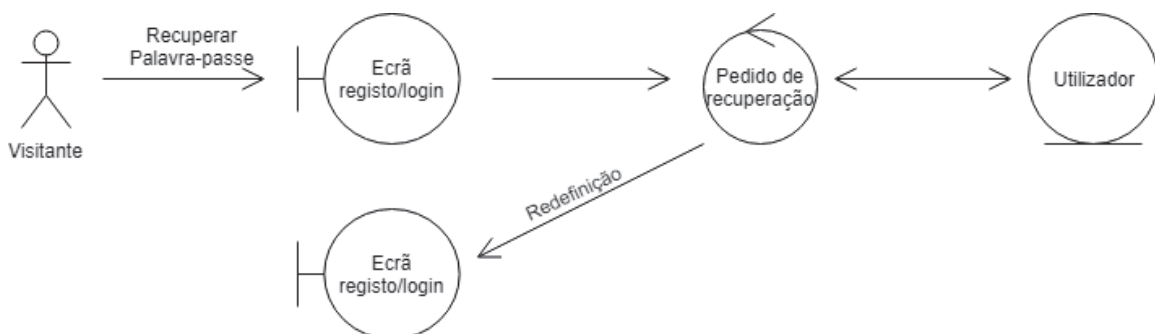
#### Diagrama de Robustez Registrar



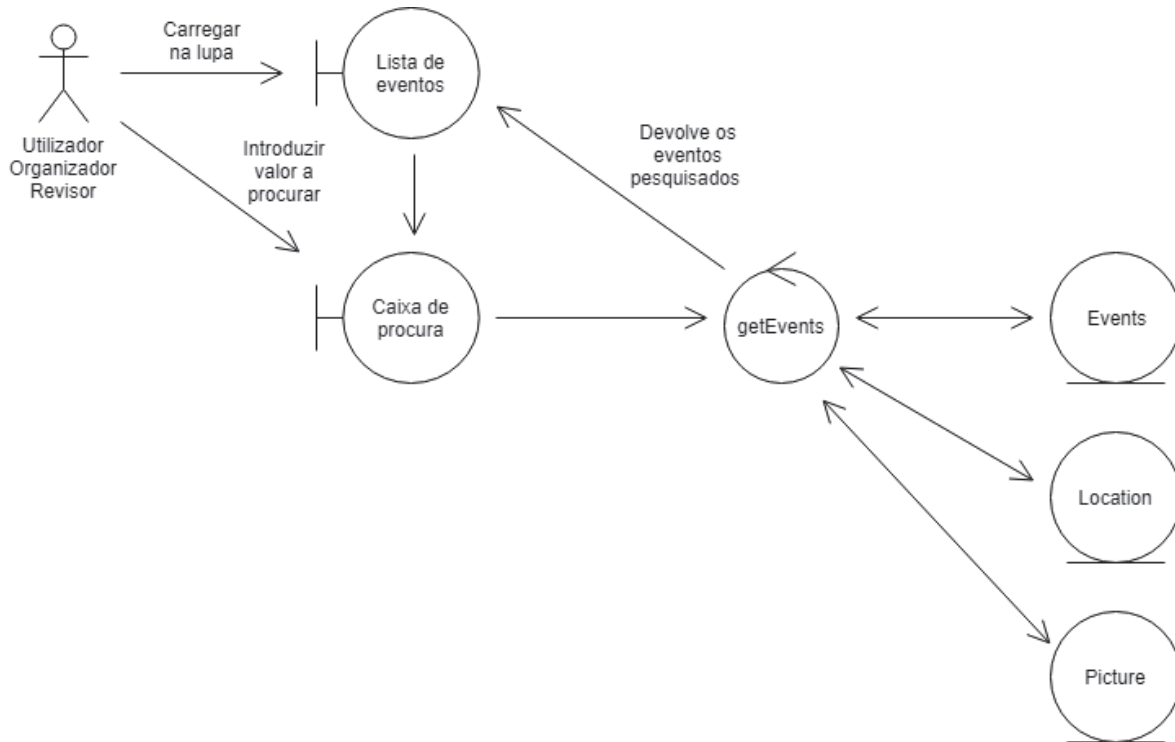
#### Diagrama de Robustez Login



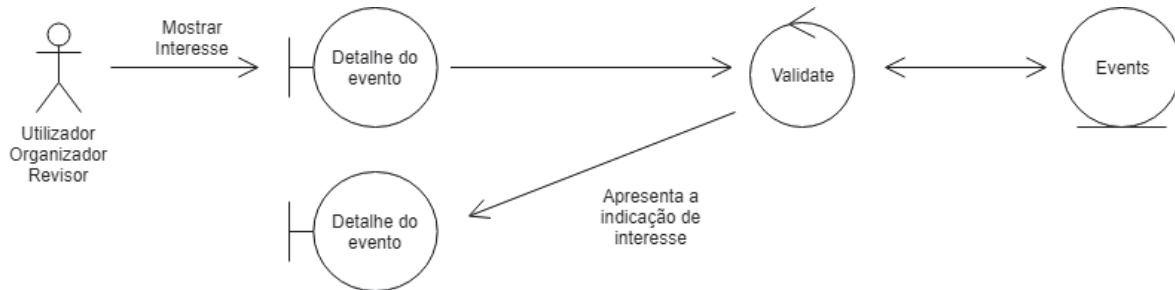
#### Diagrama de Robustez Recuperar Palavra-Passe



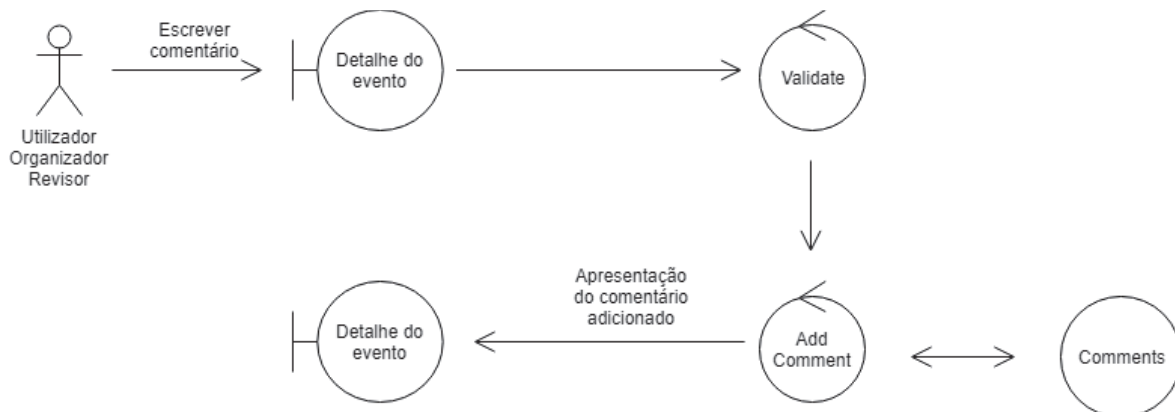
### Diagrama de Robustez Procurar Evento



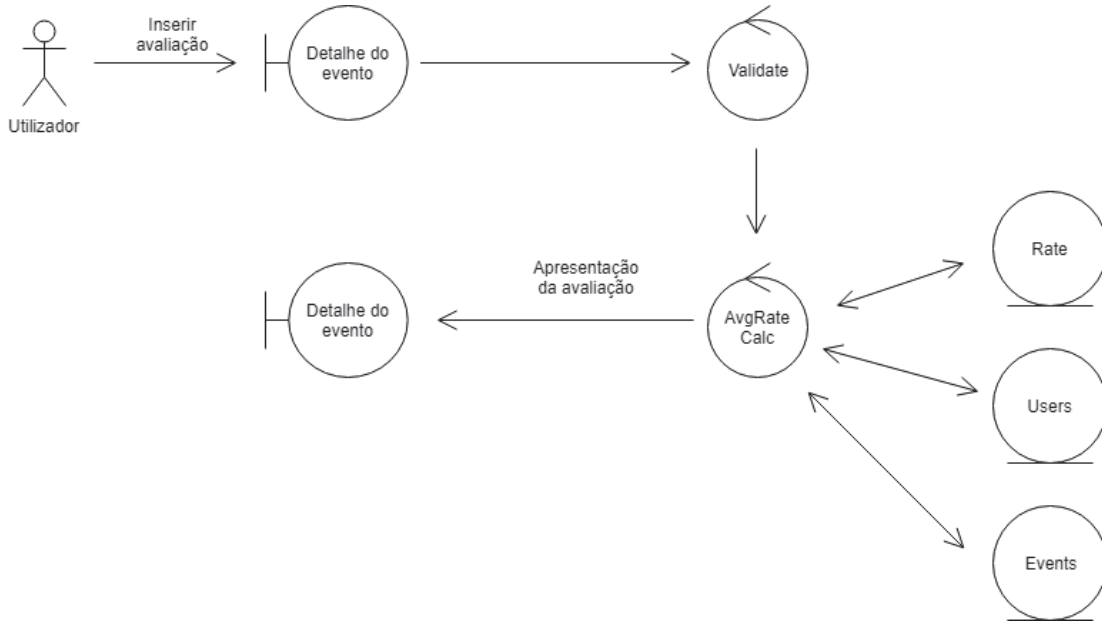
### Diagrama de Robustez Mostrar Interesse



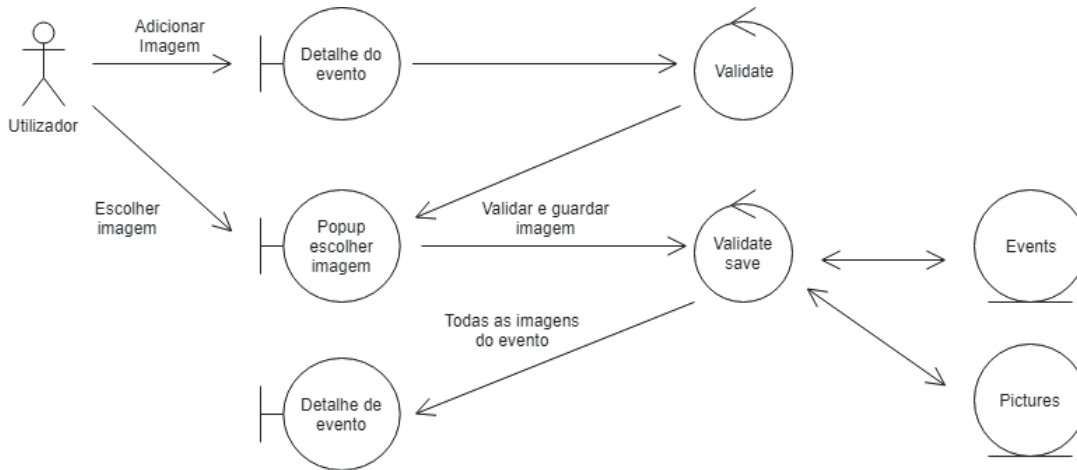
### Diagrama de Robustez Comentar



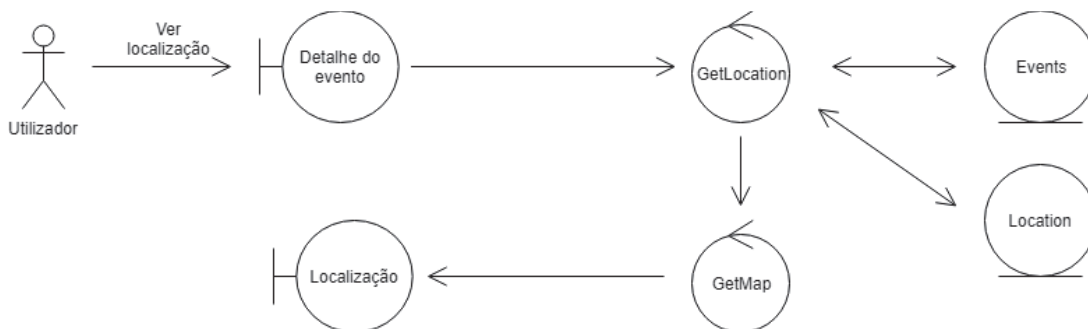
### Diagrama de Robustez Avaliar



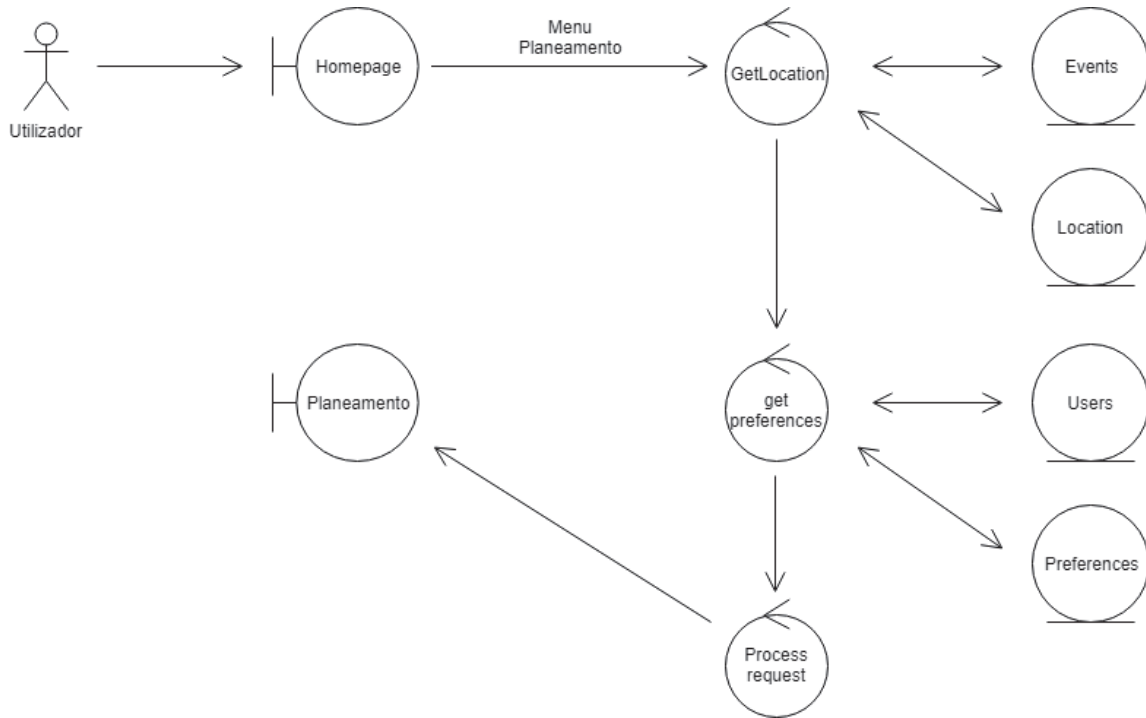
### Diagrama de Robustez Adicionar Imagens



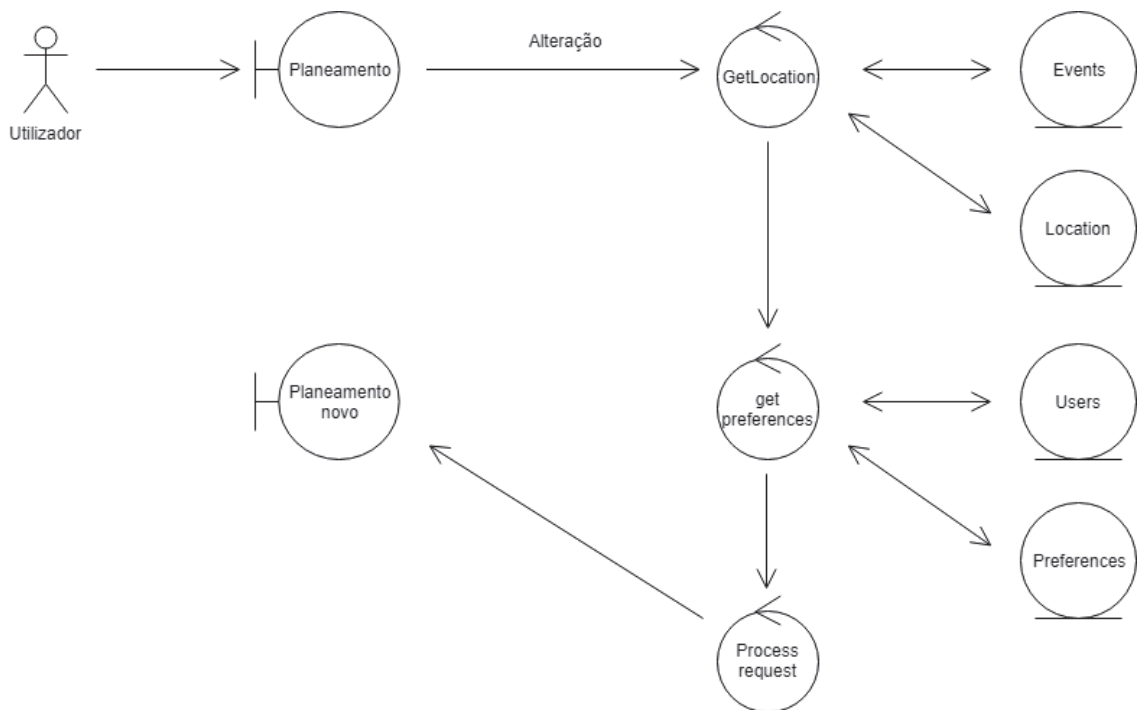
### Diagrama de Robustez Ver Localização



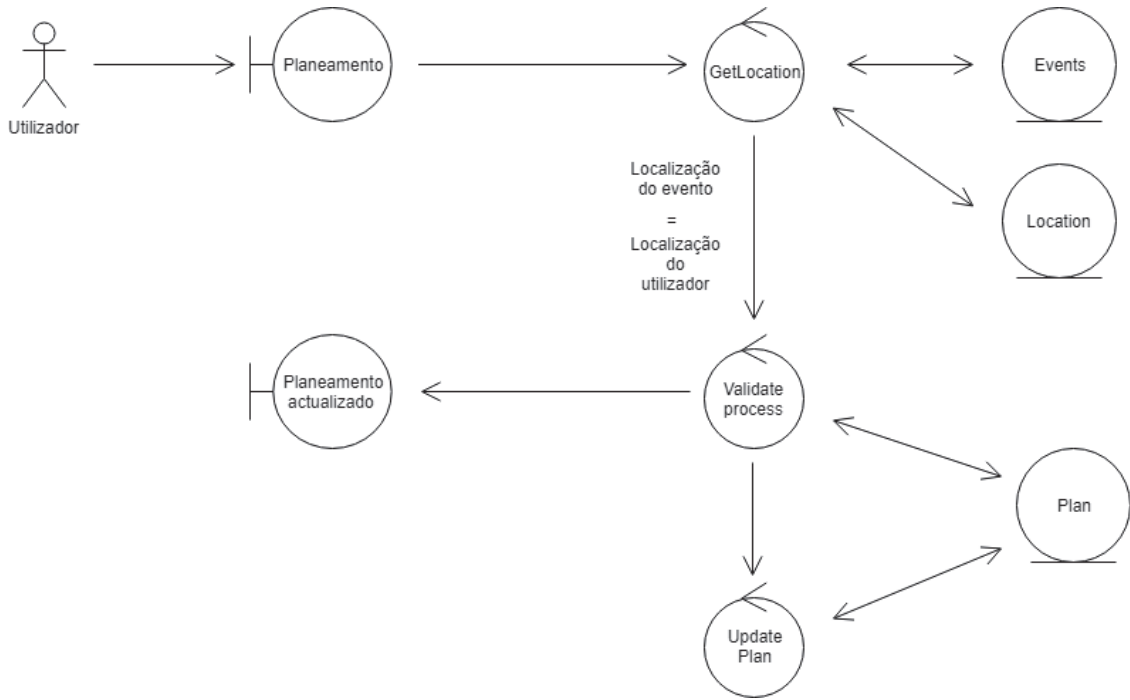
### Diagrama de Robustez Planear



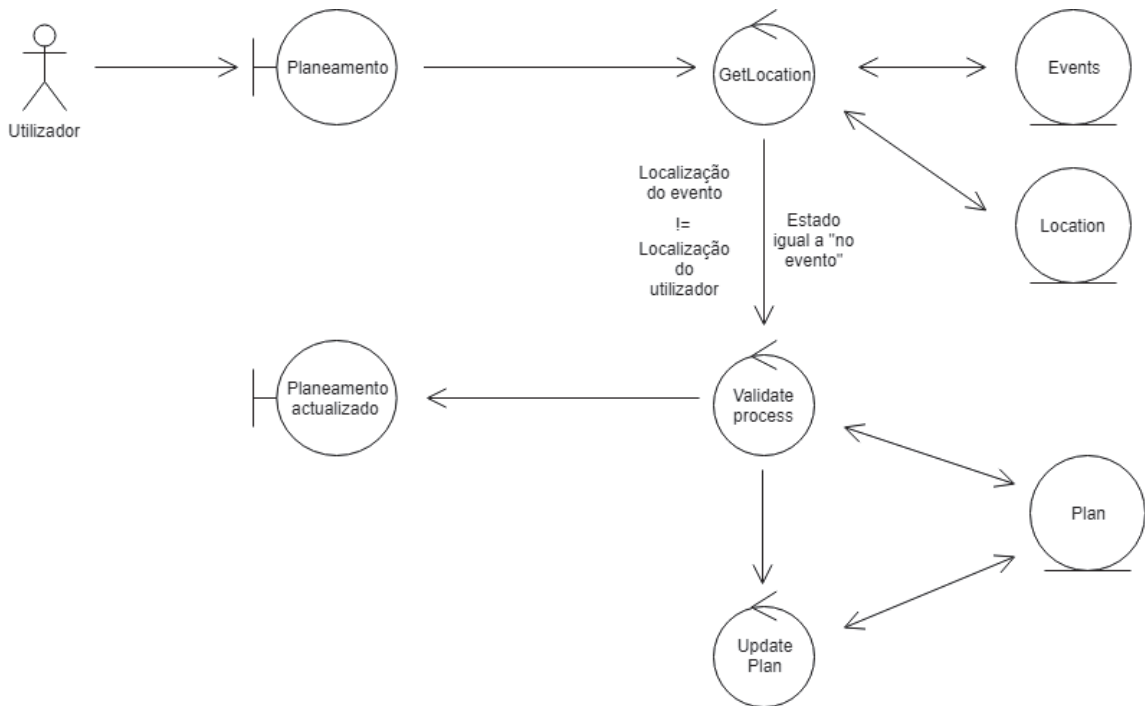
### Diagrama de Robustez Editar Plano



**Diagrama de Robustez Participar no Evento**



**Diagrama de Robustez Estive no Evento**



### **Diagrama de Robustez Consultar Viagem**

