



Universidade do Minho

Escola de Engenharia

Licenciatura em Engenharia informática

Mestrado Integrado em Engenharia Informática

Unidade Curricular de Laboratórios de Informática IV

Ano Letivo de 2021/2022

Guia Turístico

Gonçalo Braz Afonso (a93178)

Gonçalo José Teixeira Pereira (a93168)

Simão Pedro Sá Cunha (a93262)

Tiago Luís Dias da Silva (a93277)

Novembro, 2021

Data de Recepção	
Responsável	
Avaliação	
Observações	

Guia Turístico

Gonçalo Braz Afonso (a93178)

Gonçalo José Teixeira Pereira (a93168)

Simão Pedro Sá Cunha (a93262)

Tiago Luís Dias da Silva (a93277)

Novembro, 2021

Resumo

O seguinte relatório destina-se ao projeto da Unidade Curricular de Laboratórios de Informática IV, que tem como objetivo a concretização de projeto de software que consiga, solidamente, executar uma solução para a temática apresentada, “Um Guia para Locais de Interesse”.

Este projeto é composto por 3 fases.

Uma primeira, que este relatório pretende abranger, onde se realizou uma fundamentação dos objetivos que o projeto pretende atingir, através do estudo do mercado existente na área das aplicações de guias de viagem, mapas e sites de compilação de vários locais de interesse. Deste estudo, concluímos o que poderia ser implementado numa ferramenta deste género de modo a diferenciá-la da concorrência, sem negligenciar a sua validade como um modelo de negócio.

Posteriormente foi escrita uma especificação detalhada do software a implementar. Para tal, auxiliámo-nos através da geração de representações UML, tais como modelos de domínio, use cases e sequência. Para tal, fizemos um levantamento dos requisitos necessários para a concretização do projeto.

Realizado este estudo, compreendemos a necessidade de validar a integridade dos dados requeridos para o funcionamento do sistema planeado. Com esta necessidade em mente, construímos a projeção da base de dados a ser usada, que servirá como guia para a implementação da mesma.

Finalmente, de modo que potenciais clientes/financiadores possam ter uma mais clara ideia da implementação e uso do software, ilustramos uma interface de utilizador protótipo, com uma respetiva explicação das suas previstas funcionalidades e operações que disponibilizará a um utilizador.

Na segunda fase, após realizada a troca de relatórios com uma outra equipa, realizar-se-á uma análise crítica do mesmo.

Na terceira e última fase, será implementado o projeto da equipa mencionada, e, posteriormente, será apresentado.

Área de Aplicação: Desenvolvimento de um software, utilizando metodologias de engenharia de software, destinado a auxiliar a pesquisa de atividades disponíveis a um dado utilizador.

Palavras-Chave: Engenharia de Software, Guia de locais de interesse, Base de dados, Diagrama de GANTT, Aplicação web-based, Aplicação mobile.

Índice

1. Introdução	1
1.1. Contextualização	1
1.2. Apresentação do Caso de Estudo	1
1.3. Motivação e Objetivos	2
1.4. Justificação e utilidade do sistema	2
1.5. Estabelecimento da identidade do projeto	3
1.6. Identificação dos recursos necessários	3
1.7. Maqueta do sistema	4
1.8. Definição de um conjunto de medidas de sucesso	4
1.9. Plano de desenvolvimento (diagrama GANTT)	5
2. Levantamento e análise de requisitos	6
3. Especificação e modelação do software	9
3.1. Modelo de Domínio	10
3.2. Modelo de Use Case	11
3.2.1. Diagramas de Use Case	11
3.2.2. Especificações dos Use Case	15
3.3. Diagramas de sequência	26
3.4. Diagrama de componentes	39
3.5. Diagrama de classes	40
4. Conceção do sistema de dados	41
4.1. Apresentação geral da estrutura do sistema de dados	41
4.2 Identificação e associação de atributos	42
5. Esboço das Interfaces do Sistema	44
6. Conclusões e trabalho futuro	50
6.1. Conclusões	50
6.2. Trabalho futuro	50
Referências	51

Índice de Figuras

Figura 1. Maqueta do Sistema	4
Figura 2. Diagrama de GANTT	5
Figura 3. Modelo de Domínio	10
Figura 3. Modelo de domínio	10
Figura 4. Filtrar locais	11
Figura 5. Gerir o plano	12
Figura 6. Gerir perfil	13
Figura 7. Use case geral	14
Figura 8. Registar-se na aplicação	15
Figura 9. Autenticar-se na aplicação	15
Figura 10. Consultar a lista de locais	16
Figura 11. Selecionar local	16
Figura 12. Filtrar locais por nome	17
Figura 13. Filtrar locais por proximidade	17
Figura 14. Filtrar locais por classificação	18
Figura 15. Filtrar locais por categoria	18
Figura 16. Fazer review	19
Figura 17. Consultar reviews de um local	19
Figura 18. Criar plano	20
Figura 19. Consultar plano	21
Figura 20. Editar plano	22
Figura 21. Eliminar plano	23
Figura 22. Consultar perfil	23
Figura 23. Editar dados do utilizador	23
Figura 24. Consultar lista de reviews do utilizador	24
Figura 25. Consultar review do utilizador	24
Figura 26. Eliminar review do utilizador	24
Figura 27. Editar locais favoritos	25
Figura 28. DS "Registo de Utilizador"	26
Figura 29. DS "Autenticação de Utilizador"	26
Figura 30. DS "Consultar lista de locais"	27
Figura 31. DS "Selecionar local turístico"	27
Figura 32. Filtrar locais por nome	28
Figura 32. Filtrar locais por nome	28
Figura 33. DS "Filtrar locais por proximidade"	28
Figura 34. DS "Filtrar locais por classificação"	29
Figura 35. DS "Filtrar locais por categoria"	29
Figura 36. DS "Fazer review de um local turístico"	30

Figura 37. DS "Consultar reviews de um local"	Erro!
Marcador não definido.	
Figura 38. DS "Criar plano de viagem"	31
Figura 39. DS "Consultar plano de viagem"	32
Figura 40. DS "Editar plano de viagem"	33
Figura 41. DS "Escolher intervalo de tempo do local"	34
Figura 42. DS "Eliminar plano de viagem"	34
Figura 43. DS "Consultar perfil do utilizador"	35
Figura 44. DS "Editar dados do perfil do utilizador"	35
Figura 45. DS "Consultar lista de reviews do Utilizador"	36
Figura 46. DS "Consultar review do utilizador"	36
Figura 47. DS "Eliminar review do Utilizador"	37
Figura 48. DS "Editar lista de favoritos"	38
Figura 49. Diagrama de componentes	39
Figura 50. Diagrama de classes	40
Figura 51. Diagrama da base de dados	41
Figura 52. Página inicial	44
Figura 53. Página de registo	45
Figura 54. Página de login	45
Figura 56. Página de favoritos do utilizador	46
Figura 55. Página de perfil	46
Figura 57. Página das reviews do utilizador	47
Figura 58. Página dos planos do utilizador	47
Figura 59. Página de pesquisa	48
Figura 60. Página de edição de um plano	48
Figura 61. Página de um local	49

Índice de Tabelas

Tabela 1. Entidade User	42
Tabela 2. Entidade Place	42
Tabela 3. Entidade Plan	42
Tabela 4. Entidade Review	43

1. Introdução

1.1. Contextualização

As pessoas sempre tiveram a necessidade de conhecer novos lugares e ter novas experiências, desde o tempo em que o ser humano era um nómada que mudava de casa frequentemente em busca do local com os recursos mais abundantes, até ao seu contemporâneo, que consegue observar um registo desta memória em museus.

Num mundo digital e em permanente desenvolvimento, existe um certo nível de stress causado, principalmente, pelos empregos das pessoas, problemas financeiros e problemas familiares. Assim, nasce a necessidade de descontrair, sozinho ou acompanhado, para esquecer (nem que apenas momentaneamente) os seus problemas. Para isso existem diversos locais tais como restaurantes, discotecas, cinemas, centros comerciais, museus, etc.

No entanto, existe, por vezes, uma indecisão quanto à escolha desse local, esta pode ser explicada pelo facto de existir muita oferta ou por uma incompatibilidade com o horário da pessoa ou grupo de pessoas, sendo impossível delinear um plano para uma “saída de sábado à noite”.

Neste contexto, a add&SEEK pretende oferecer uma solução simples e eficaz para os inconvenientes destas pessoas.

1.2. Apresentação do Caso de Estudo

O projeto a ser desenvolvido deverá disponibilizar ao utilizador:

- Listar locais turísticos através do uso de filtros tais como: nome, distância ao utilizador, classificação e categoria;
- Fazer uma *review* a um local turístico;
- Adicionar locais turísticos ao conjunto de favoritos do utilizador;
- Criar um plano para visitar locais turísticos numa dada localidade e num dado horário.

Fruindo a estas funcionalidades, o utilizador conseguirá encontrar um local que se adeque às suas preferências e que garanta a sua satisfação.

1.3. Motivação e Objetivos

O motivo para a criação e desenvolvimento deste projeto prende-se pelo facto de querer competir com os gigantes do mercado de localização e avaliação de locais turísticos.

Assim sendo, a add&SEEK pretende ser inovador neste ramo, distinguindo-se pela elaboração de um plano para alguém, com o desejo de realizar alguma atividade, mas indeciso sobre a mesma, bastando apenas ter o conhecimento da localização do utilizador e a duração da sua estadia nesse local.

Desta forma, a add&SEEK tem como objetivo oferecer uma opção simples e alternativa a outras *apps*, pondo ao dispor de qualquer utilizador registado na *app* um vasto leque de locais turísticos que facilmente são listados de modo a satisfazer as suas preferências. Queremos, então, que com esta nossa distinção aos nossos competidores, os utilizadores possam ver as suas vidas facilitadas e, simultaneamente, estimular a economia local.

1.4. Justificação e utilidade do sistema

A ideia para a criação da add&SEEK surgiu devido ao facto de termos constatado que a sociedade nem sempre consegue encontrar o lugar ideal para passar o seu tempo livre.

As pessoas vivem atualmente uma vida atarefada e stressante devido a, por exemplo, aos seus empregos e, muitas vezes, não têm muito tempo “para desperdiçar” em escolhas de locais turísticos, que, embora destinadas a prazer e deste modo aliviar a tensão, resultam num acréscimo destes devido à dificuldade da tomada de decisão, ou, simplesmente, não têm ideias e precisam de auxílio externo para a tomada de decisão. Atualmente, as pessoas utilizam algumas aplicações já criadas (concorrentes ao nosso programa) para decidir onde jantar, onde levar a família de férias ou mesmo que monumentos devemos visitar numa localidade. No entanto, estas nem sempre são as melhores opções, uma vez que requerem que o utilizador tenha o trabalho de pesquisar os locais de forma manual, podendo gerar um certo grau de impaciência na pessoa e desistir da ideia de escolher um local turístico. É aqui que a add&SEEK pretende atuar, satisfazendo esta necessidade referida, além de fornecer algumas das funcionalidades de programas da concorrência.

1.5. Estabelecimento da identidade do projeto

Foi-nos imposto como tema de projeto “Um guia para locais de interesse”, onde tivemos que, tal como o nome indica, criar uma aplicação que nos listasse os locais de interesse na região onde o utilizador se situa. Surgiu assim a [nome da aplicação].

O utilizador terá acesso a uma lista de locais turísticos da região em que se encontra, onde pode filtrar segundo alguns critérios, como o nome, categoria, classificação e distância.

Além disto, o utilizador pode elaborar um plano de viagem. Para tal, o utilizador pode adicionar um nome ao mesmo e definir um intervalo de tempo para o qual o plano deve ser definido. De seguida, o horário criado vai ser enviado por e-mail ao utilizador.

Adicionalmente, o utilizador pode adicionar locais à sua lista de favoritos, deste modo criando um atalho para referenciar os mesmos quando quiser fazer um plano ou uma visita rápida.

Ainda para mais, o utilizador pode avaliar os locais turísticos, realizando uma review na página do mesmo, atualizando assim a classificação do local.

1.6. Identificação dos recursos necessários

Para a elaboração da nossa aplicação, são necessários vários recursos relativos ao software:

- Visual Paradigm para a modelação do projeto;
- Microsoft Word para a escrita do relatório;
- IDE's para a escrita do código;
- Software para desenho da base de dados;
- Uso de Adobe Illustrator e Adobe Photoshop para a projeção do design das interfaces

No entanto, não existe aplicação sem uma equipa de engenheiros competentes e dinâmicos que consiga efetuar as tarefas em tempo útil. Foi eleito um coordenador de grupo para, além de interagir com a equipa docente, garantir que todo o trabalho fosse concretizado com qualidade e dentro dos prazos estabelecidos.

Além disto, será necessário ter uma noção do meio em que a nossa aplicação se insere. Para tal, perguntamos a pessoas próximas como era a experiência de decisão de locais turísticos a visitar. Assim poderá ser possível para que a equipa de engenheiros possa ter uma ideia de como poderá ajudar as pessoas nos problemas descritos ao longo do relatório.

1.7. Maqueta do sistema

Esta aplicação foi desenhada de forma a poder ser utilizada em dispositivos móveis. Assim, o utilizador é encorajado a usar o seu smartphone, tal como se observa na imagem abaixo, para procurar locais turísticos ou planear férias com apenas alguns cliques de forma simples, segura e eficaz.



Figura 1. Maqueta do Sistema

1.8. Definição de um conjunto de medidas de sucesso

Por forma a avaliar o êxito do produto final, foi definido um conjunto de medidas de sucesso. Consideramos que, no prazo de 3 anos, pelo menos 1% da população de cada distrito de Portugal Continental seja utilizador assíduo e que, em 5 anos, a aplicação atinja os 500 mil downloads.

add&SEEK, sendo uma aplicação que visa a resolver um problema comum à grande maioria das pessoas, tenciona, portanto, a atingir as faixas etárias acima dos 14 anos, inclusive, que abrangem aqueles que possuem liberdade e vontade para nos seus tempos livres procurar atividades no exterior para realizar.

1.9. Plano de desenvolvimento (diagrama GANTT)

Constituído o objetivo de realizar o projeto através de uma definição e justificação apropriada da sua motivação, é necessário, com o intuito de tornar o desenvolvimento o mais eficiente e coordenado possível, gerar um plano de desenvolvimento especificando as metas e os correspondentes intervalos para as suas concretizações, assim como os seus responsáveis. Foi, portanto, recorrido à construção do diagrama de Gantt de seguida apresentado. Este apresenta uma planificação da primeira fase deste projeto, especificação e planeamento, e uma projeção do esperado para a fase de implementação.

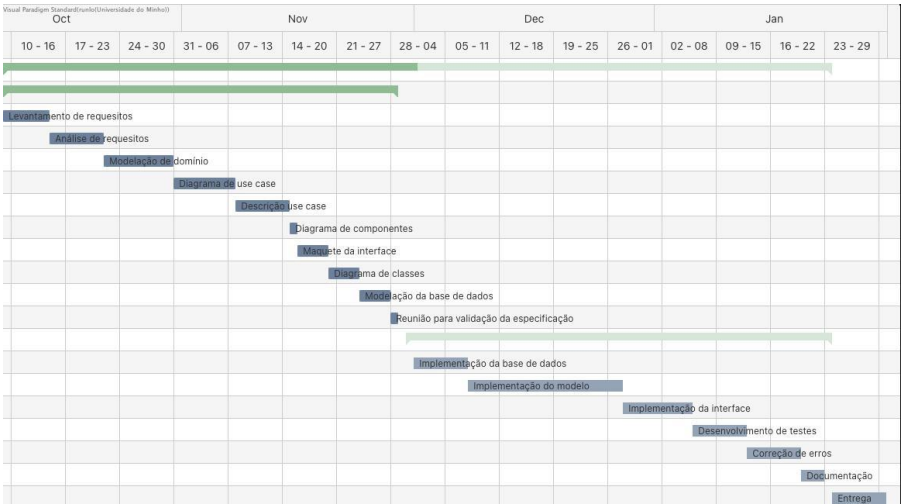


Figura 2. Diagrama de GANTT

2. Levantamento e análise de requisitos

Ao elaborar um projeto de software, a fase de levantamento de requisitos tem uma elevada importância, uma vez que garante uma perfeita simbiose entre as ideias para a aplicação e as funcionalidades do produto final.

Serão, de seguida, apresentados os requisitos do utilizador que foram obtidos através dos questionários referidos anteriormente.

2.1 Requisitos do utilizador

Para ter acesso às funcionalidades da aplicação um utilizador necessitará de possuir um e-mail, de modo a conseguir registar a sua conta, assim como de algum modo disponibilizar a sua localização, por exemplo, permitindo acesso à sua localização no sistema. Sendo portador de uma conta, ele deve ter permissão de alterar os seus dados, nomeadamente o seu e-mail, password, locais favoritos e planos de viagem.

2.2. Descrição geral dos requisitos

Requisitos funcionais

1. Registo na aplicação

- O sistema solicita o fornecimento de dados por parte do utilizador, e-mail, nome de utilizador e password.
- O sistema deve validar os dados fornecidos de modo a não permitir a existência de dois utilizadores com o mesmo e-mail.
- Validados os dados, o sistema tem de registar o novo utilizador.

2. Autenticação na aplicação

- O sistema recebe os dados do utilizador, verificando se o seu par e-mail e password pertence a algum utilizador na base de dados.
- Consequente da validação, o sistema conclui a autenticação.

3. Edição do perfil do utilizador

- O sistema deve atualizar as edições, respetivamente e-mail e password, locais favoritos e planos.
- O sistema deve garantir a segurança e privacidade destes dados, permitindo a sua alteração apenas por parte do utilizador a quem dizem respeito.

4. Pesquisa de locais turísticos

- O sistema deve apresentar ao utilizador a disponibilidade da aplicação de filtros na pesquisa de locais.
- Escolhidos os filtros, o sistema deve processá-los e apresentar uma listagem dos locais que validem esses requerimentos.
- O sistema deve permitir ao utilizador selecionar um local turístico da listagem e de seguida, redirecioná-lo para a página correspondente.

5. **Página de um local turístico**

- O sistema deve apresentar uma descrição do local turístico, assim como a sua classificação geral e secção de reviews.
- O sistema deve permitir ao utilizador seleccionar o local como destino.
- O sistema deve permitir ao utilizador criar uma única review dedicada ao local.

6. **Listagens dos planos**

- O sistema deve listar todos os planos do utilizador por ordem do dia da semana.

7. **Criação de um plano**

- O sistema deve garantir ao utilizador a possibilidade de criação de um novo plano.
- O sistema deve pedir ao utilizador que fornece os dados necessários à criação do novo plano, nome, dia da semana, intervalo de tempo e local.

8. **Consultar um plano**

- O sistema deve permitir a consulta dos dados de um plano, elaborando, o itinerário completo com todos os locais a visitar e os seus respetivos intervalos de tempo no horário.

9. **Edição de um plano**

- Consultado um plano, o sistema deve permitir a opção de o editar.
- Durante a edição do plano, o sistema deve garantir a coerência do intervalo de tempo definido para o plano, requisitando uma extensão de tempo ao utilizador caso necessário.
- O sistema deve, aquando da adição de locais, listar de acordo com o filtro local, equivalente ao local do plano.
- O sistema deve garantir que modificações feitas a planos são apenas possíveis por parte do utilizador, seu criador.
- O sistema deve atualizar os dados do plano no final da operação da edição.

Requisitos não funcionais

1. A aplicação deverá estar disponível 24 horas por dia, 7 dias por semana.
2. A aplicação deverá assegurar a privacidade e a segurança dos dados dos utilizadores assim como a sua localização.

3. Especificação e modelação do software

Com recurso ao ambiente do visual paradigm e respetivas ferramentas, desenhámos uma solução para a implementação dos vários requisitos da aplicação, anteriormente levantados. Utilizando os conhecimentos adquiridos através do estudo da UC Desenvolvimento de Sistemas de Software, representamos esta solução através das ilustrações UML que se seguem.

3.1. Modelo de Domínio

Recorrendo aos requisitos previamente analisados, desenvolvemos um modelo de domínio no âmbito de explicitar as entidades, interações e reações de que o sistema irá dispor.

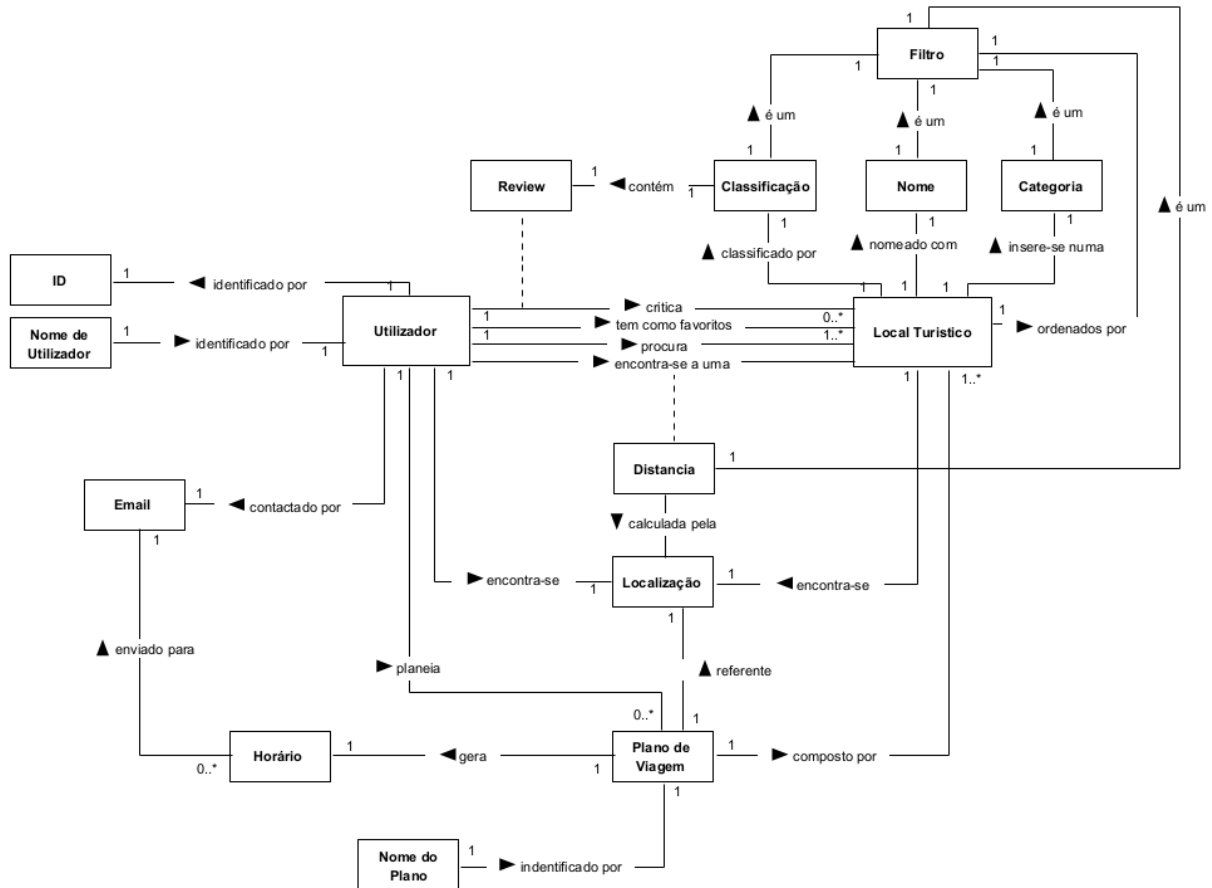


Figura 3. Modelo de Domínio

Observando sucintamente o modelo podemos observar que os objetivos acima descritos foram conseguidos de forma simples e clara, permitindo facilmente, por exemplo, entender o papel do Utilizador: procura de um local de destino, a sua critica e a criação de planos.

3.2. Modelo de Use Case

3.2.1. Diagramas de Use Case

Para o ator que interage com o sistema, o utilizador, delimitámos 3 casos de uso, **filtragem de locais**, **gerência de planos** e **gerência do seu perfil**, que por sua vez se englobam no caso de uso **geral**.

A **filtragem de locais**, consiste em permitir a um utilizador **autenticado**, realizar uma pesquisa por locais de interesse, filtrados através de 4 opções: nome do local, distância ao utilizador, classificação mínima do local e categoria do mesmo.

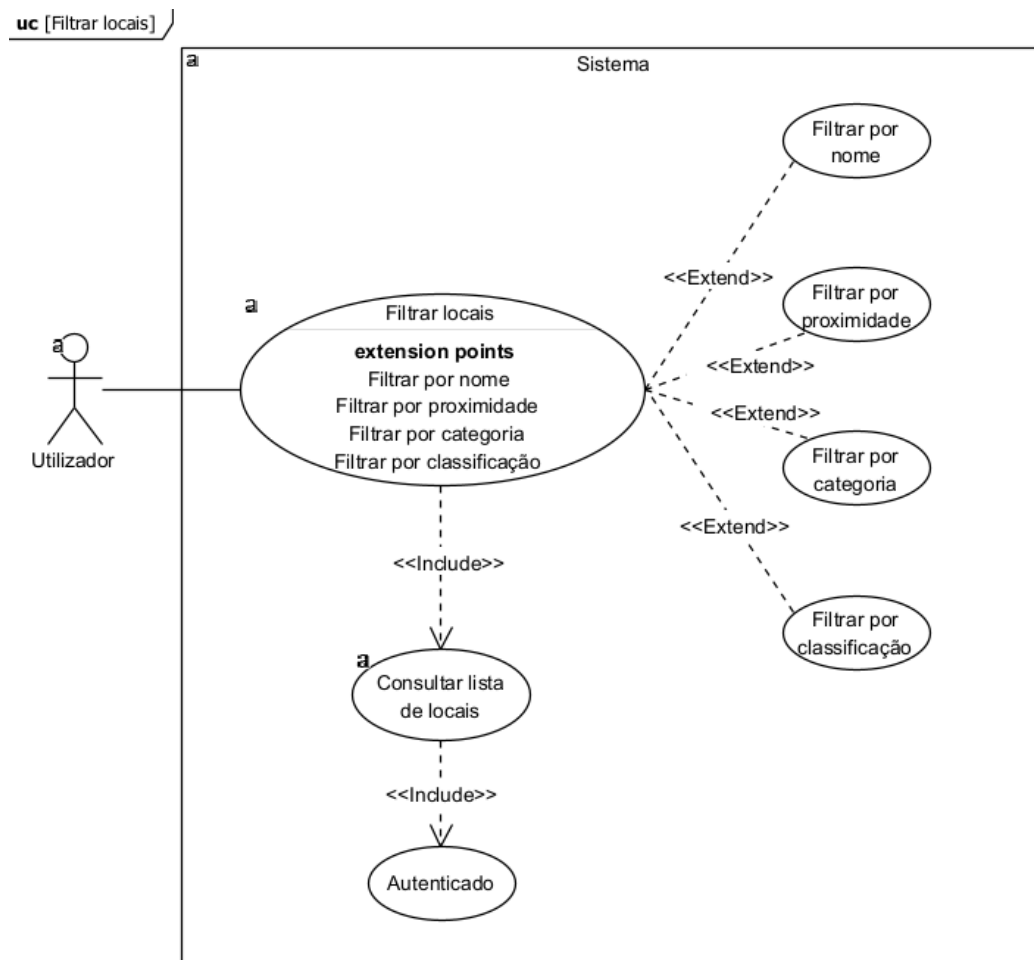


Figura 4. Filtrar locais

A **gerência de planos** garante ao ator a criação e edição de itinerários designados para certos dias da semana. Nestes o utilizador poderá adicionar, remover e editar locais a visitar nesse dia, desde que sejam válidos em relação ao intervalo de tempo e região designados ao plano.

uc [Gerir planos]

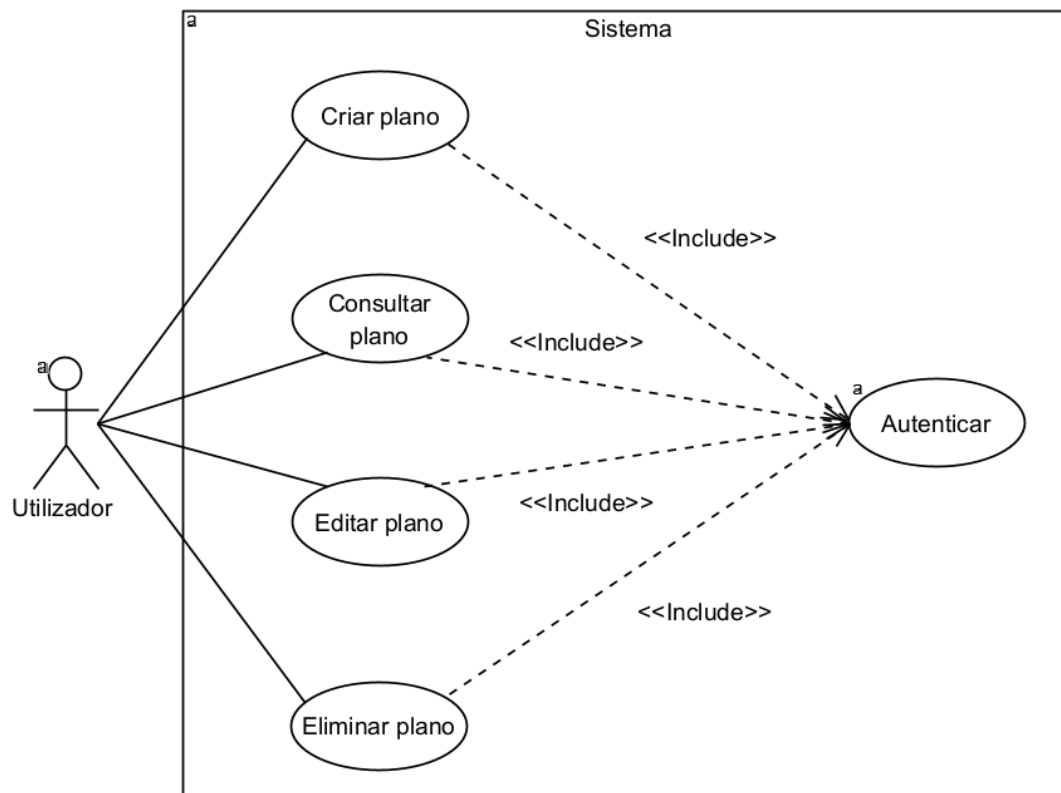


Figura 5. Gerir o plano

Gerência de perfil permite ao utilizador alterar os seus dados, ou seja, alterar o seu e-mail, palavra-passe, a lista de locais favoritos e as suas reviews. Cada um destes é validado antes da sua atualização.

uc [Gerir perfil]

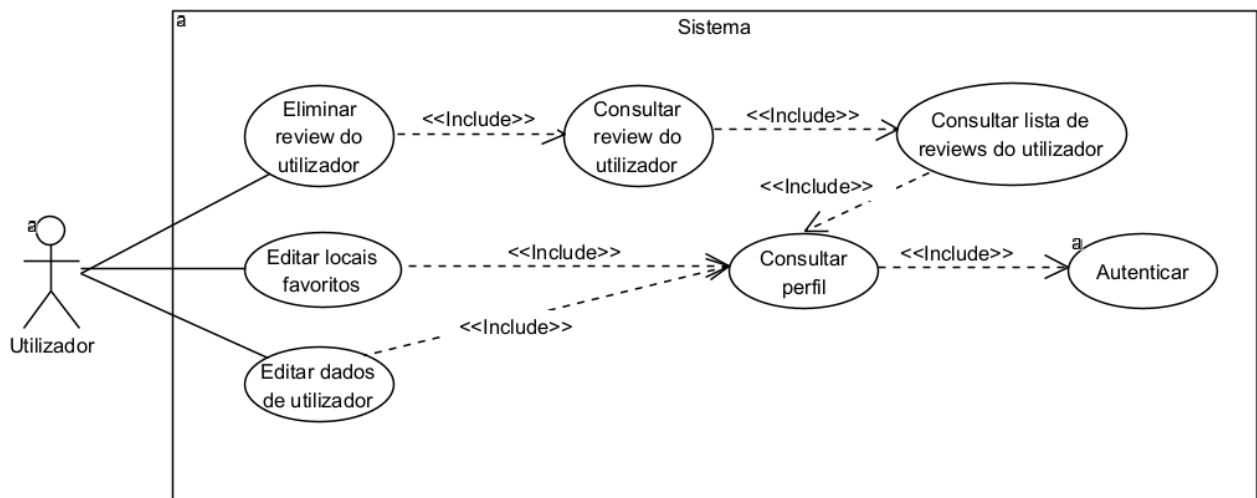


Figura 6. Gerir perfil

O **caso de uso geral** engloba os três cenários descritos e ainda descreve o poder de criar uma conta e fazer reviews de locais, do utilizador.

uc [Geral]

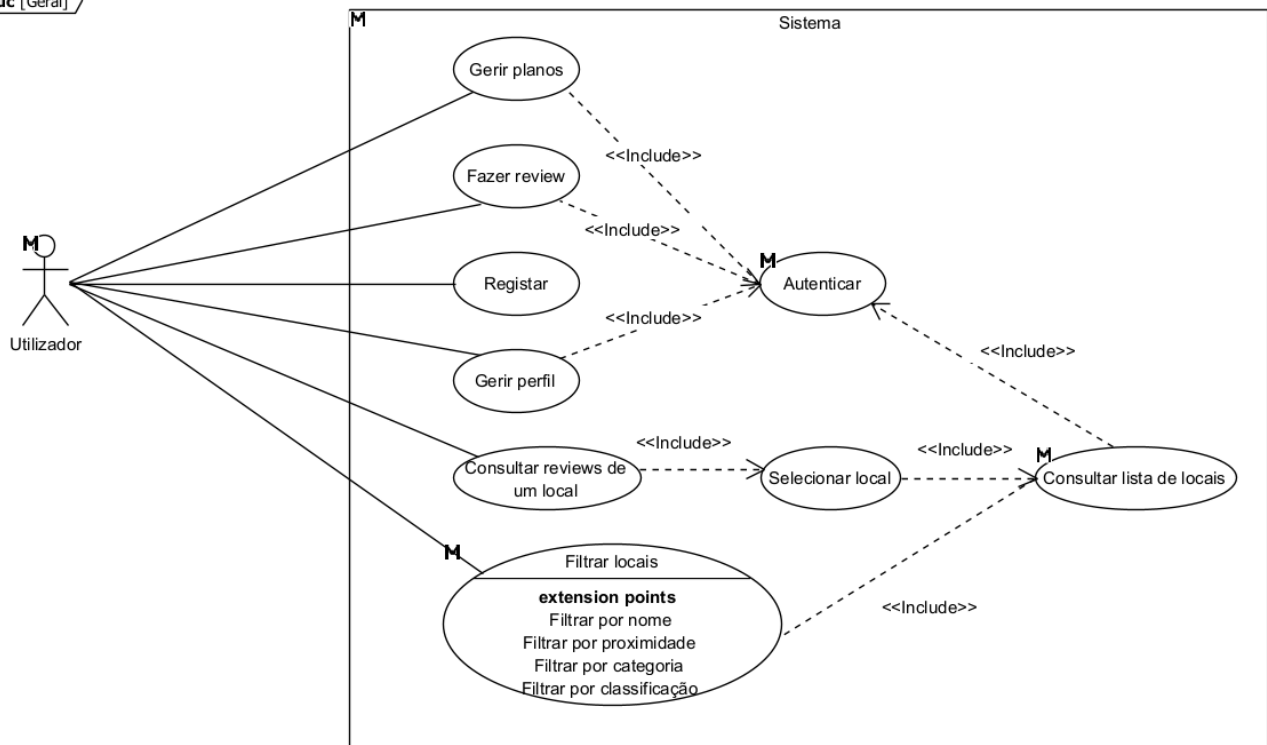


Figura 7. Use case geral

3.2.2. Especificações dos Use Case

O registo na aplicação permite, a utilizadores sem uma conta no sistema, criar uma no mesmo, para se poderem autenticar.

Use Case:	Registar	
Ator:	Utilizador	
Pré-condição:	True	
Pós-condição:	Utilizador registado no sistema	
	Ator	Sistema
Fluxo normal:	1. Fornece dados de registo.	
		2. Verifica se já existe um utilizador no sistema com os dados fornecidos.
		3. Guarda registo do utilizador.
		4. Informa sucesso do registo.
Fluxo exceção 1. [Já existe um utilizador com os mesmos dados no sistema] (Passo 2):		2.1. Informa insucesso do registo.

Figura 8. Registar-se na aplicação

Para realizar quase todas as operações do sistema, um utilizador necessita estar autenticado. Assim, ele deverá fornecer as suas credenciais, e-mail e palavra-passe e, no caso de serem validados, será redirecionado para o menu principal.

Use Case:	Autenticar	
Ator:	Utilizador	
Pré-condição:	Ator não autenticado	
Pós-condição:	Ator autenticado	
	Ator	Sistema
Fluxo normal:	1. Indica os dados de autenticação.	
		2. Verifica validade dos dados introduzidos.
		3. Redireciona para menu principal.
Fluxo exceção 1 [Dados intruzidos inválidos] (Passo 2):		2.1. Informa que os dados introduzidos são inválidos.

Figura 9. Autenticar-se na aplicação

Consulta mais simples da informação que aplicação disponibiliza. O sistema lista todos os locais de interesse, sem processar quaisquer filtros.

Use Case:	Consultar lista de locais	
Ator:	Utilizador	
Pré-condição:	Ator autenticado	
Pós-condição:	Sistema apresenta lista de locais	
	Ator	Sistema
Fluxo normal:		1. Apresenta uma lista com os locais.

Figura 10. Consultar a lista de locais

Tendo ao seu dispor a listagem dos locais, o utilizador tem a possibilidade de seleccionar um deles, após o qual, lhe será apresentada a página do mesmo.

Use Case:	Selecionar local	
Ator:	Utilizador	
Pré-condição:	Utilizador na lista de locais	
Pós-condição:	Sistema apresenta local selecionado	
	Ator	Sistema
Fluxo normal:		1. Apresenta o local selecionado.

Figura 11. Selecionar local

No pedido da realização das listagens dos locais de interesse, o utilizador pode pedir que à pesquisa seja aplicado um filtro de nome. Assim o sistema irá pedir o fornecimento de um nome, e após a sua receção, processará a listagem de locais que obedeçam a este requisito. No caso de o nome fornecido não ser válido, o sistema informará o utilizador da invalidade e repetirá o processo de filtragem por nome.

Use Case:	Filtrar por nome	
Ator:	Utilizador	
Pré-condição	Utilizador numa lista de locais	
Pós-condição	Sistema apresenta lista pontos turísticos filtrados por nome	
	Ator	Sistema
Fluxo normal:	1. Fornece nome.	2. Verifica se nome é válido.
		3. Apresenta lista de todos os pontos turísticos pelo nome fornecido.
Fluxo alternativo 1. [Nome inválido] (Passo 2):		2.1. Informa que nome é inválido.
		2.2. Retorna passo 1.

Figura 12. Filtrar locais por nome

Da mesma forma, a pesquisa pode ser filtrada para apenas encontrar os locais que se encontrem a um raio de distância máxima do utilizador.

Use Case:	Filtrar por proximidade	
Ator:	Utilizador	
Pré-condição	Utilizador numa lista de locais	
Pós-condição	Sistema apresenta lista pontos turísticos filtrados por proximidade	
	Ator	Sistema
Fluxo normal:	1. Fornece distância máxima.	2. Verifica se distância é válida.
		3. Apresenta lista de todos os pontos turísticos num raio da distância máxima fornecido, com o centro na localização do utilizador.
Fluxo alternativo 1. [Distância inválida] (Passo 2):		2.1. Informa que distância é inválida.
		2.2. Retorna passo 1.

Figura 13. Filtrar locais por proximidade

À filtragem por classificação, é fornecida uma classificação mínima que um local deve ter, e consoante esta, apenas serão apresentados os locais com classificação média igual ou superior à pedida pelo utilizador.

Use Case:	Filtrar por classificação	
Ator:	Utilizador	
Pré-condição	Utilizador numa lista de locais	
Pós-condição	Sistema apresenta lista pontos turísticos filtrados por classificação	
	Ator	Sistema
Fluxo normal:	1. Fornece classificação mínima.	
		2. Verifica se classificação é válida.
		3. Apresenta lista de todos os pontos turísticos com classificação superior à classificação mínima fornecida.
Fluxo alternativo 1. [Classificação inválida] (Passo 2):		2.1. Informa que classificação é inválida.
		2.2. Retorna passo 1.

Figura 14. Filtrar locais por classificação

Por último, a pesquisa pode ser limitada a certas categorias, por exemplo, restaurantes, museus ou monumentos.

Use Case:	Filtrar por categoria	
Ator:	Utilizador	
Pré-condição:	Utilizador numa lista de locais	
Pós-condição:	Sistema apresenta lista pontos turísticos filtrados por categoria	
	Ator	Sistema
Fluxo normal:	1. Fornece uma categoria.	
		2. Verifica se categoria é válida.
		3. Apresenta lista de todos os pontos turísticos da categoria fornecida.
Fluxo alternativo 1. [Categoria inválida] (Passo 2):		2.1. Informa que categoria é inválida.
		2.2. Retorna passo 1.

Figura 15. Filtrar locais por categoria

Quando o utilizador tiver escolhido um local de interesse e se encontrar na página correspondente, ele tem a opção de fazer uma review do mesmo. Feita a review o sistema atualiza a base de dados.

Use Case:	Fazer Review	
Ator:	Utilizador	
Pré-condição:	Local Turístico selecionado	
Pós-condição:	Review registada na base de dados	
	Ator	Sistema
Fluxo normal:	1. Fornece comentário.	
	2. Fornece classificação.	
		3. Regista review na base de dados.

Figura 16. Fazer review

Também na página de um local, o utilizador pode consultar todas as reviews sobre esse local. No caso de haver reviews, o sistema lista-as, caso contrário, o utilizador permanecerá na página do local.

Use Case:	Consultar Reviews de um local	
Ator:	Utilizador	
Pré-condição:	Local selecionado	
Pós-condição:	Apresenta reviews do local	
	Ator	Sistema
Fluxo normal:		1. Apresenta as reviews do local.
Fluxo exceção 1. (Não existem reviews a serem apresentadas) [Passo 1]:		1.1. Regressa ao local selecionado.

Figura 17. Consultar reviews de um local

No menu principal, o utilizador pode optar por criar um novo plano de viagem. Começada esta operação, o utilizador tem de definir um dia de semana para o plano, um intervalo de tempo em que ele irá decorrer e uma região a que ele se irá restringir. O sistema valida esta região e para terminar a criação de um plano, pede ao utilizador que indique um nome para o plano. Validado este nome, o sistema regista o novo plano na base de dados.

No caso da região ou do nome serem inválidos, o sistema irá informar o utilizador e pedir um novo fornecimento destes dados.

Use Case:	Criar Plano	
Ator:	Utilizador	
Pré-condição:	Ator autenticado	
Pós-condição:	Plano registado na base de dados	
	Ator	Sistema
Fluxo normal:	1. Indica dia da semana.	
	2. Indica intervalo de tempo.	
	3. Indica cidade.	
		4. Valida cidade.
	5. Fornece nome do plano.	
		6. Valida nome do plano.
		7. Sistema regista plano na base de dados.
Fluxo Alternativo 1. (Cidade inválida) [Passo 3]:		4.1. Informa que cidade é inválida.
		4.2. Retorna passo 3.
Fluxo Alternativo 2. (Nome do plano já utilizado) [Passo 6]:		6.1. Sistema informa que nome já foi utilizado.
		6.2. Retorna passo 5.

Figura 18. Criar plano

O sistema disponibiliza também ao utilizador a visualização da lista dos seus planos e consequentemente seleccionar um dos planos para consulta.

Caso não existam planos de viagem disponíveis o sistema informa o utilizador.

Use Case:	Consultar Plano	
Ator:	Utilizador	
Pré-condição:	Ator autenticado	
Pós-condição:	Apresenta plano de viagem	
	Ator	Sistema
Fluxo normal:		1. Apresenta os planos de viagem do utilizador.
	2. Seleciona um plano de viagem para consultar.	
		3. Apresenta o plano de viagem selecionado.
Fluxo exceção 1. (Sem planos para apresentar) [Passo 1]		1.1. Indica que não existem planos de viagens para apresentar.

Figura 19. Consultar plano

Durante a consulta de um plano, este deve ter a opção de ser editado. A edição do plano possibilita a adição e eliminação dos locais definidos para o itinerário.

Para a operação de adição de locais, durante a listagem dos locais a apresentar ao utilizador, o sistema filtra aqueles que se situem num local válido em relação à região associada ao plano de viagem. Escolhido o local a adicionar, o utilizador terá de fornecer um intervalo de tempo para o visitar, que será validado pelo sistema.

Ao adicionar um local, o sistema valida se o intervalo de tempo definido ao plano é suficiente para as durações de visita definidas na lista de locais e, em caso de não o ser, avisa o utilizador da necessidade de uma extensão do intervalo de tempo. Aqui se o utilizador recusar esta proposta, o sistema cancelará a edição do plano, retornando o seu estado para antes do utilizador ter escolhido a opção de edição.

A remoção de locais de interesse do plano, apenas é possível no caso de já terem sido adicionados planos anteriormente.

Use Case:	Editar Plano	
Ator:	Utilizador	
Pré-condição:	Plano seleccionado pelo utilizador	
Pós-condição:	Registo do plano atualizado na base de dados	
	Ator	Sistema
Fluxo normal:		1. Apresenta os locais do plano.
	2. Indica que quer adicionar um local ao plano.	
		3. Apresenta lista de locais da cidade associada ao plano.
	4. Seleciona local.	
		5. Requesita intervalo de tempo associado ao local.
	6. Fornece intervalo de tempo.	
		7. Valida intervalo de tempo.
		8. Adiciona local turístico ao plano.
		9. Atualiza plano turístico.
Fluxo Alternativo 1. (Clica para remover local turístico do plano) [Passo 2]:	2.1. Escolhe local turístico a eliminar.	
		2.2. Pede confirmação.
	2.3. Confirma a operação.	
		2.4. Elimina local turístico do plano.
		2.5. Regressa ao passo 6.
Fluxo Alternativo 2. (Intervalo de tempo ultrapassado) [Passo 7]:		7.1. Informa que o intervalo de tempo do plano foi ultrapassado.
		7.2. Requesita aumento do intervalo de tempo do plano.
	7.3. Confirma aumento.	
		7.4. Regressa ao passo 9.
Fluxo Alternativa 3. (Intervalo de tempo sobreposto a outro) [Passo 7]		7.1. Informa que o intervalo de tempo está sobreposto a outro de um local do plano.
		7.2. Retorna ao passo 6.
Fluxo Alternativo 4. (Recusa operação) [Passo 2.3]:		2.3.1. Regressa ao passo 9.
Fluxo Alternativo 5. (Recusa aumento) [Passo 7.3]:		7.3.1. Regressa ao passo 6.
Fluxo exceção 1. (Plano sem locais turísticos) [Passo 2.1]:		2.2.1. Regressa ao passo 9.

Figura 20. Editar plano

O utilizador pode ainda, eliminar um plano da lista de planos.

Use Case:	Eliminar Plano	
Ator:	Utilizador	
Pré-condição:	Plano seleccionado pelo utilizador	
Pós-condição:	Registo do plano removido da base de dados	
	Ator	Sistema
Fluxo normal:		1. Pede confirmação.
	2. Confirma operação.	
		3. Remove plano da base de dados.
Fluxo exceção 1. (Utilizador cancela operação) [Passo 2]:		2.1. Regressa à visualização do plano.

Figura 21. Eliminar plano

O sistema tem de garantir que o utilizador consegue visualizar os dados do seu perfil, isto é, o seu e-mail e nome de utilizador.

Use Case:	Consultar perfil	
Ator:	Utilizador	
Pré-condição:	Ator autenticado	
Pós-condição:	Apresenta perfil do utilizador	
	Ator	Sistema
Fluxo normal:		1. Apresenta o perfil do utilizador.

Figura 22. Consultar perfil

No menu de perfil, o sistema permite ainda a sua edição, perguntando ao utilizador os novos dados e, caso este decida gravar a atualização, atualiza o utilizador na base de dados.

Use Case:	Editar dados do utilizador	
Ator:	Utilizador	
Pré-condição:	Utilizador a consultar o perfil	
Pós-condição:	Dados do utilizador atualizados	
	Ator	Sistema
Fluxo normal:		1. Apresenta os campos editáveis de perfil.
	2. Introduce novos dados no campo que pretende editar.	
	3. Grava as alterações.	
		4. Atualiza os dados do utilizador e retorna à página do perfil.
Fluxo exceção 1. (Cancela alterações) [Passo 3]:		3.1. Retorna à página do perfil do utilizador.

Figura 23. Editar dados do utilizador

Dentro do perfil do utilizador, terá um registo de todas as reviews feitas pelo utilizador, acessível por ele.

Use Case:	Consultar lista das reviews do utilizador	
Ator:	Utilizador	
Pré-condição:	Utilizador a consultar o perfil	
Pós-condição:	Apresenta reviews do local	
	Ator	Sistema
Fluxo normal:		1. Apresenta as reviews feitas pelo utilizador.
Fluxo exceção 1. (Não existem reviews a serem apresentadas) [Passo 1]:		1.1. Retorna à página do perfil do utilizador.

Figura 24. Consultar lista de reviews do utilizador

A consulta anterior permite a seleção de uma review individual para observação mais detalhada.

Use Case:	Consultar review do utilizador	
Ator:	Utilizador	
Pré-condição:	Utilizador a consultar a lista de reviews feitas	
Pós-condição:	Apresenta reviews do local no ecrã	
	Ator	Sistema
Fluxo normal:		1. Apresenta no ecrã a review selecionada pelo utilizador.

Figura 25. Consultar review do utilizador

Uma review do utilizador pode ser por ele eliminada.

Use Case:	Eliminar review do utilizador	
Ator:	Utilizador	
Pré-condição:	Utilizador a consultar uma review feita	
Pós-condição:	Apresenta reviews do local	
	Ator	Sistema
Fluxo normal:		1. Pede confirmação.
	2. Confirma operação.	
		3. Elimina review da base de dados.
		4. Regressa à lista de reviews feitas.
Fluxo exceção 1. (Utilizador recusa operação):		2.1. Regressa ao passo 4.

Figura 26. Eliminar review do utilizador

Por último, no perfil do utilizador está guardada uma lista dos lugares favoritos dele, que pode ser editada.

O utilizador pode adicionar locais aos seus favoritos e eliminar locais previamente existentes. Ambos os casos seguem verificações por parte do sistema tal como as edições estudadas acima.

Use Case:	Edtar locais favoritos	
Ator:	Utilizador	
Pré-condição:	Utilizador a consultar o perfil	
Pós-condição:	Utilizador atualiza a sua lista de locais favoritos	
	Ator	Sistema
Fluxo normal:		1. Apresentar a lista de locais favoritos.
	2. Indica que quer adicionar novo local favorito.	
		3. Apresenta os locais turísticos.
	4. Seleciona local turístico.	
		5. Valida local turístico.
		6. Adiciona local turístico à lista de favoritos.
		7. Atualiza lista de favoritos.
Fluxo Alternativo 1. (Clica para remover local turístico) [Passo 2]:	2.1. Escolhe local a eliminar.	
		2.2. Pede confirmação.
	2.3. Confirma ação.	
		2.4. Elimina local da lista de favoritos.
		2.5. Regressa ao passo 7.
Fluxo Alternativo 2. (Recusa operação) [Passo 2.3]:	2.3.1. Recusa ação.	
		2.3.2. Regressa ao passo 7.
Fluxo exceção 1. (Lista de locais favoritos vazia) [Passo 2.1]:		2.1.1. Regressa ao passo 7.

Figura 27. Editar locais favoritos

3.3. Diagramas de sequência

Os diagramas de sequência permitem uma clarificação simples do fluxo de dados no decorrer das interações dos Use Cases.

1. Registo de Utilizador

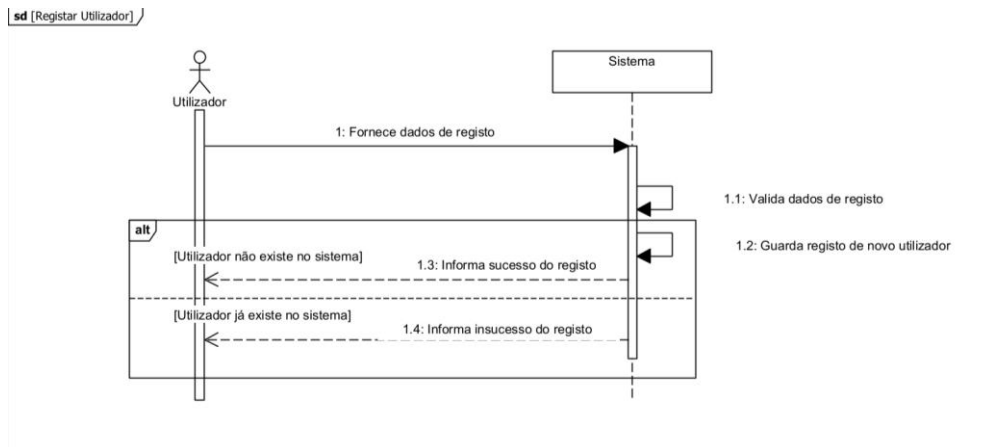


Figura 28. DS “Registo de Utilizador”

2. Autenticação de Utilizador

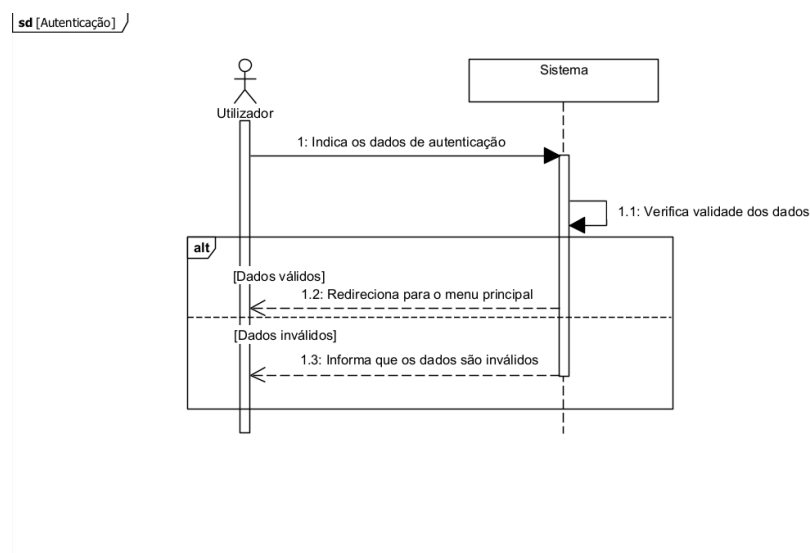


Figura 29. DS "Autenticação de Utilizador"

3. Consultar lista de locais

sd [Consultar lista de locais]

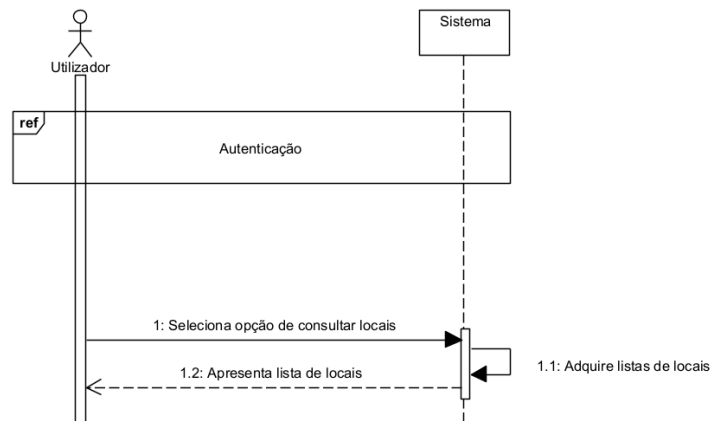


Figura 30. DS "Consultar lista de locais"

4. Selecionar local turístico

sd [Seleciona local]

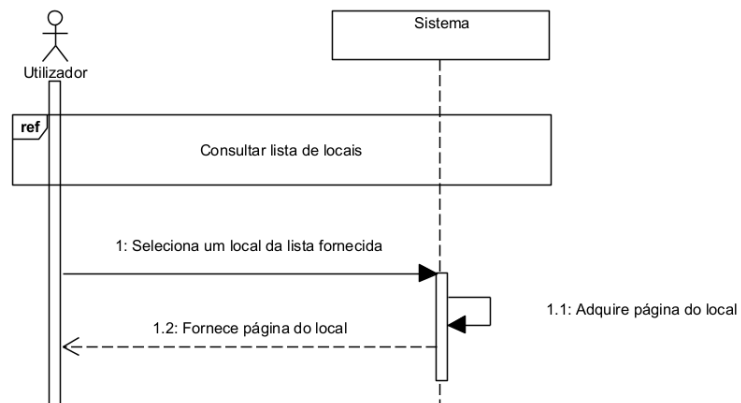


Figura 31. DS "Selecionar local turístico"

5. Filtros de pesquisa

a. Filtrar locais por nome

sd [Filtrar por nome] /

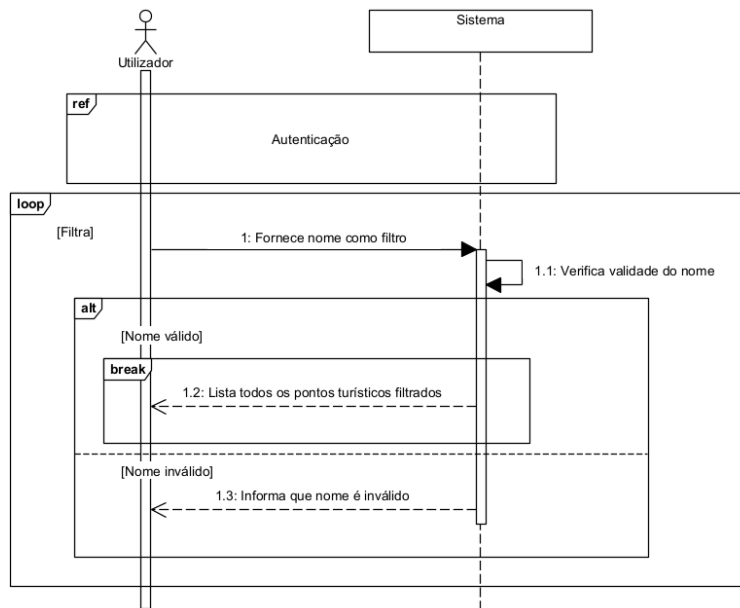


Figura 32. Filtrar locais por nome

b. Filtrar locais por proximidade

sd [Filtrar por proximidade] /

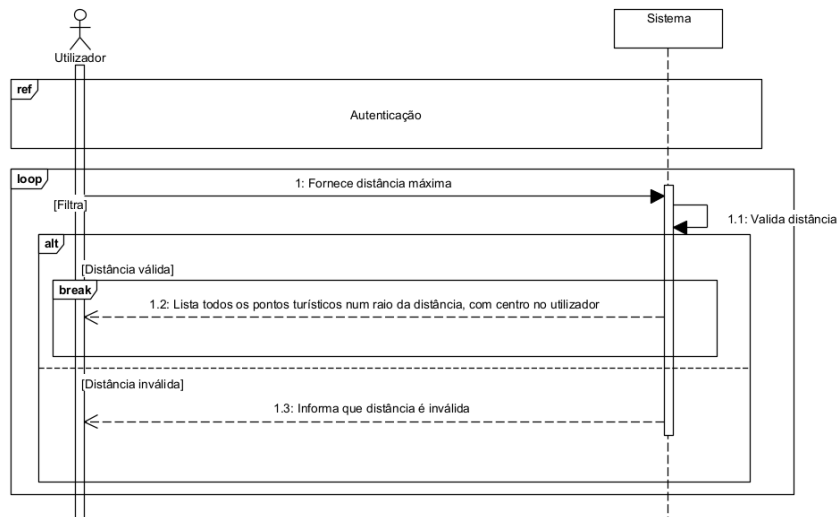


Figura 33. DS "Filtrar locais por proximidade"

c. Filtra locais por classificação

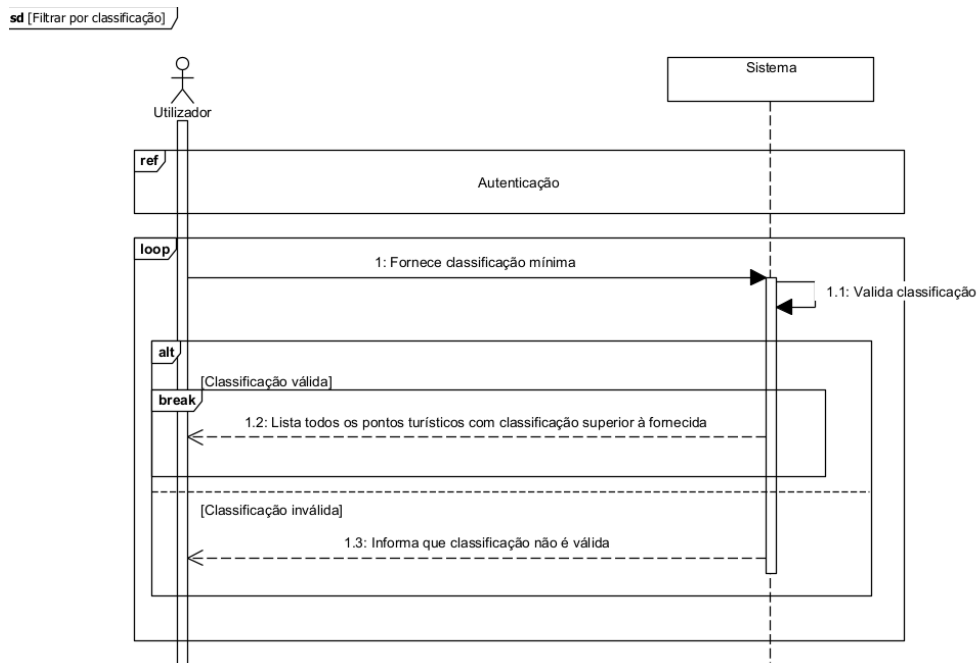


Figura 34. DS "Filtrar locais por classificação"

d. Filtrar locais por categoria

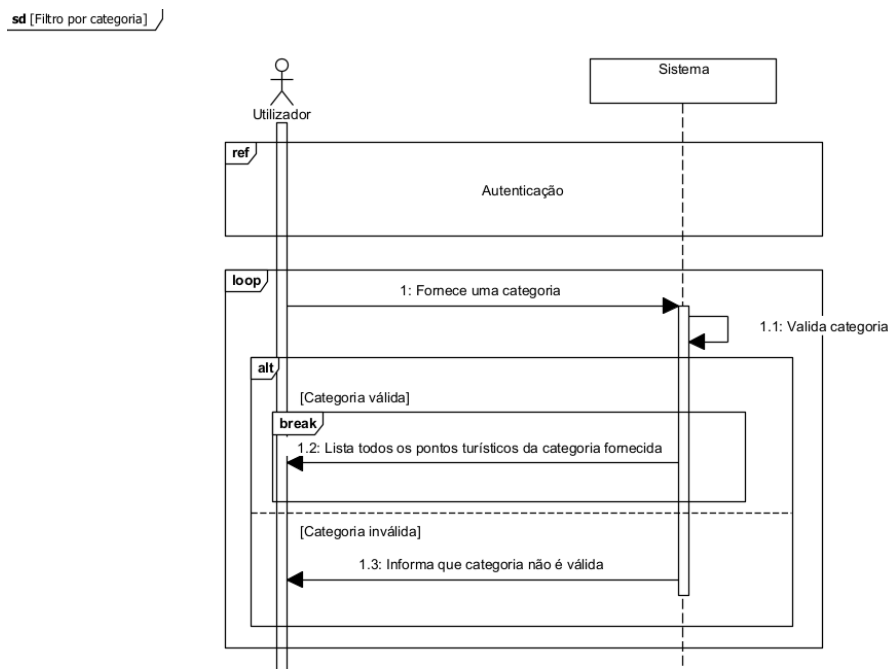


Figura 35. DS "Filtrar locais por categoria"

6. Fazer review de um local turístico

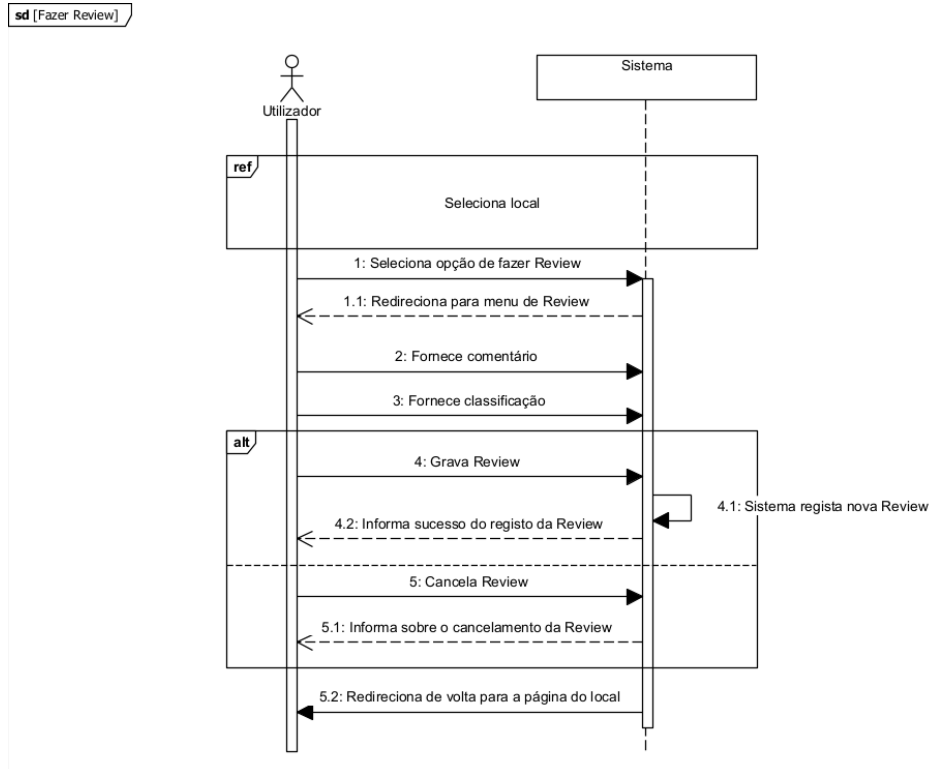


Figura 36. DS "Fazer review de um local turístico"

7. Consultar reviews de um local

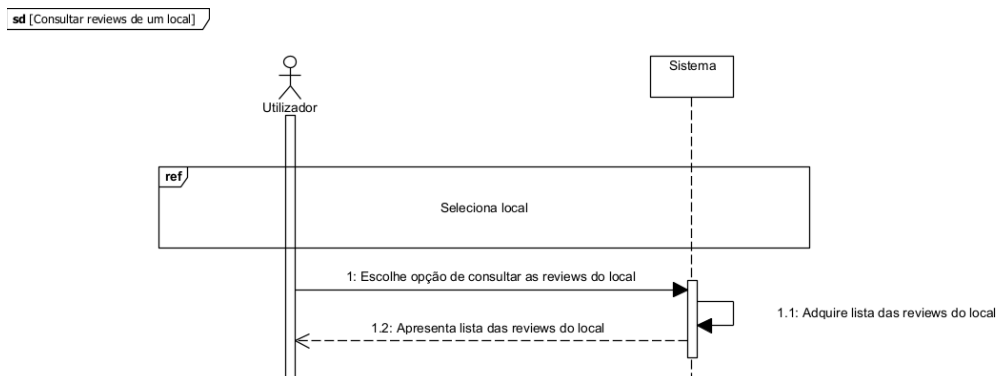


Figura 37. DS "Consultar reviews de um local"

8. Criar plano de viagem

sd [Criar Plano]

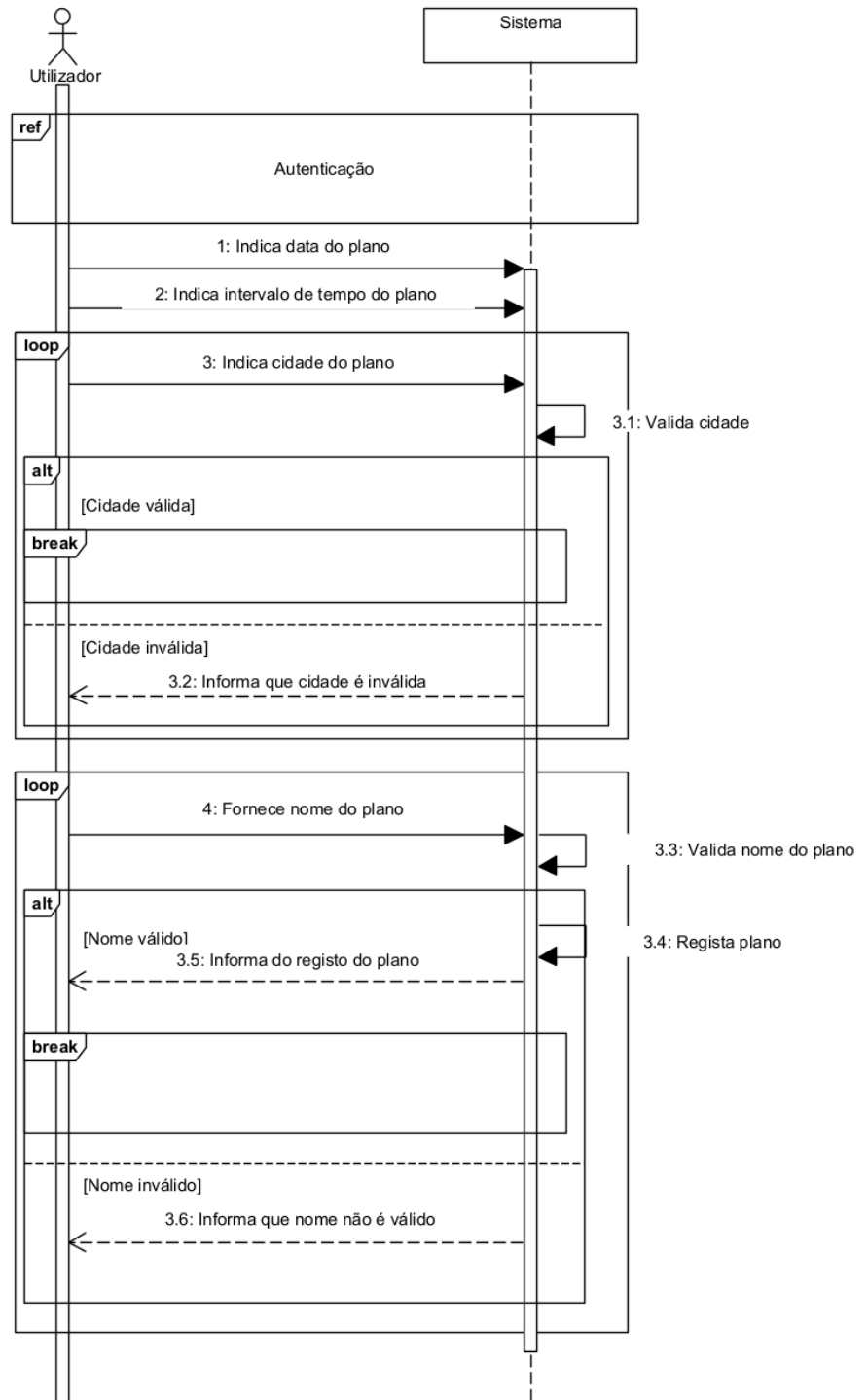


Figura 38. DS "Criar plano de viagem"

9. Consultar plano de viagem

sd [Consultar plano] /

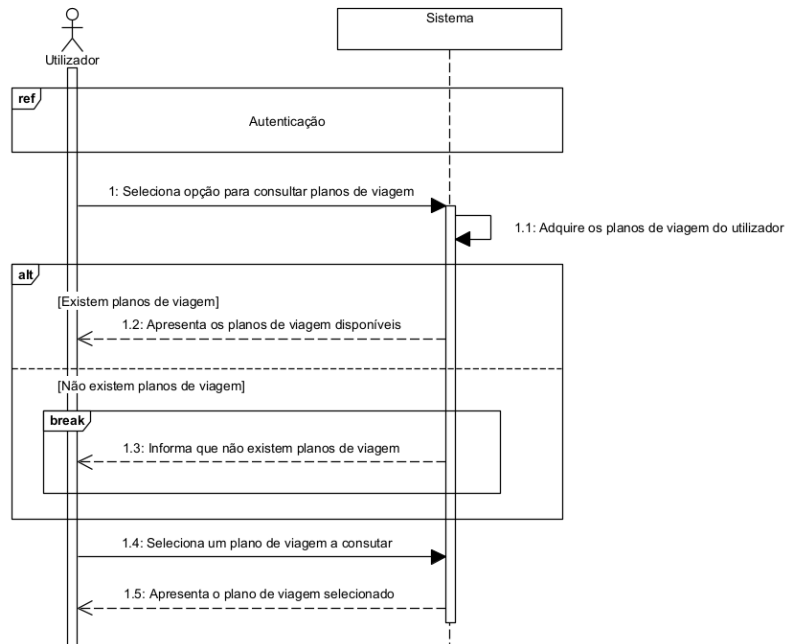


Figura 39. DS "Consultar plano de viagem"

10. Editar plano de viagem

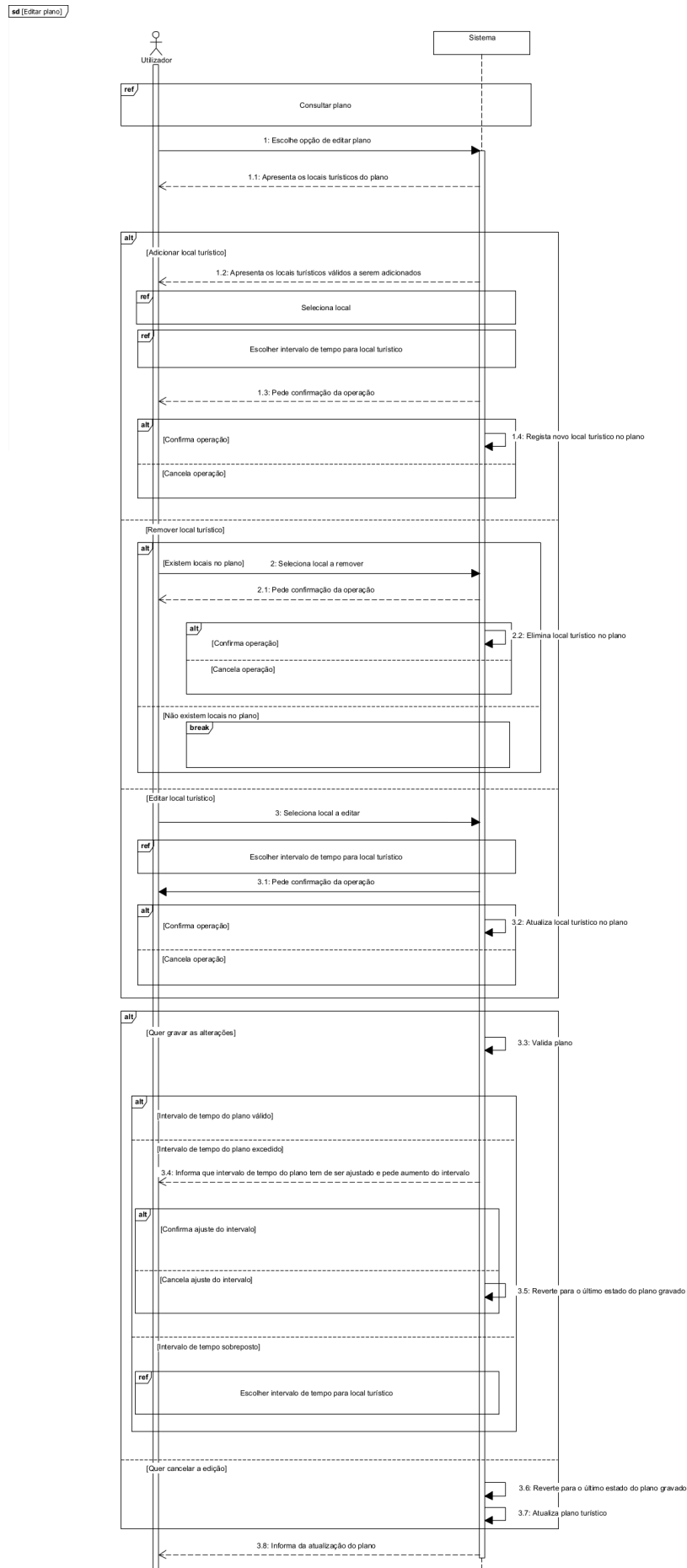


Figura 40. DS "Editar plano de viagem"

a. Escolher intervalo de tempo do local a adicionar no plano

sd [Escolher intervalo de tempo para local]

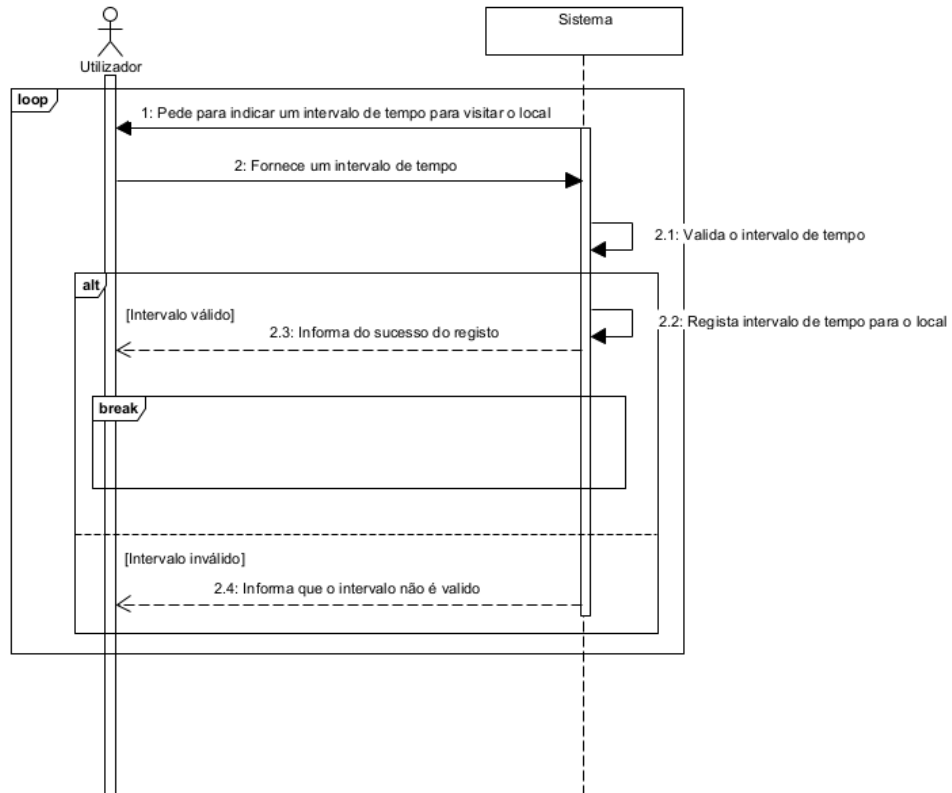


Figura 41. DS "Escolher intervalo de tempo do local"

11. Eliminar plano de viagem

sd [Eliminar plano]

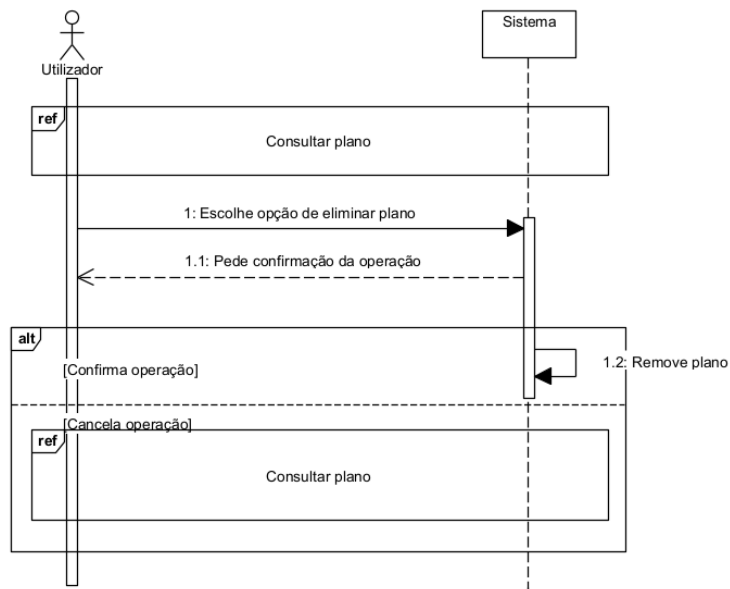


Figura 42. DS "Eliminar plano de viagem"

12. Consultar perfil do utilizador

sd [Consultar perfil] /

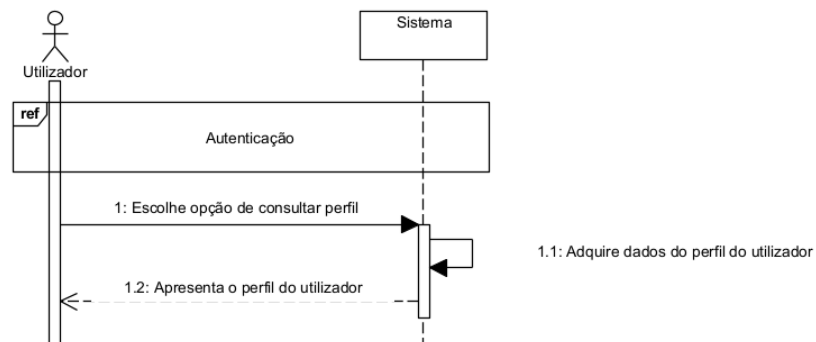


Figura 43. DS "Consultar perfil do utilizador"

13. Editar dados do perfil do utilizador

sd [Editar dados do utilizador] /

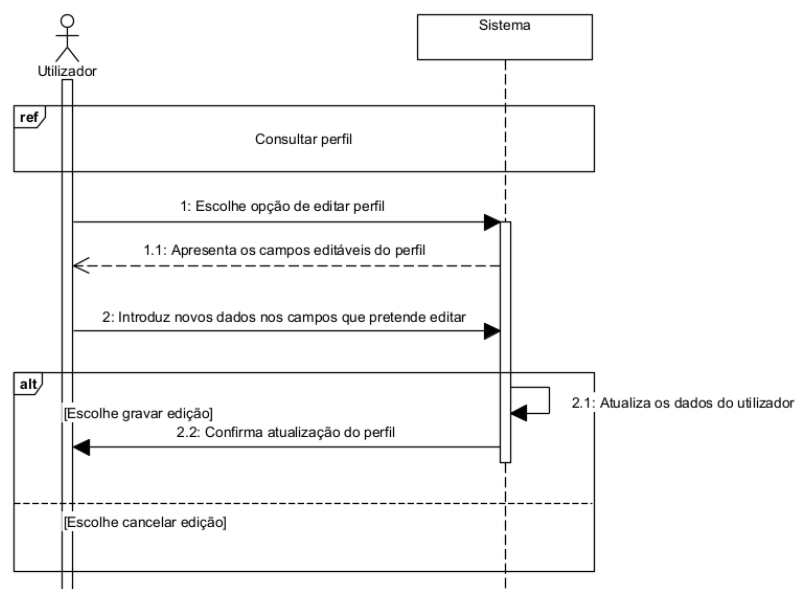


Figura 44. DS "Editar dados do perfil do utilizador"

14. Consultar lista de reviews do Utilizador

sd [Consultar lista de reviews do utilizador]

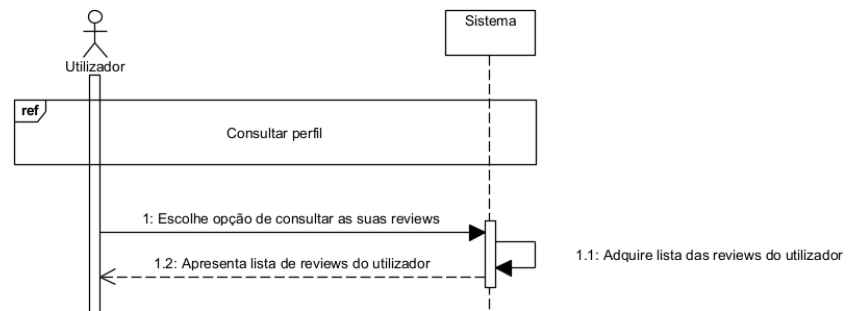


Figura 45. DS "Consultar lista de reviews do Utilizador"

15. Consultar review do Utilizador

sd [Consultar review do utilizador]

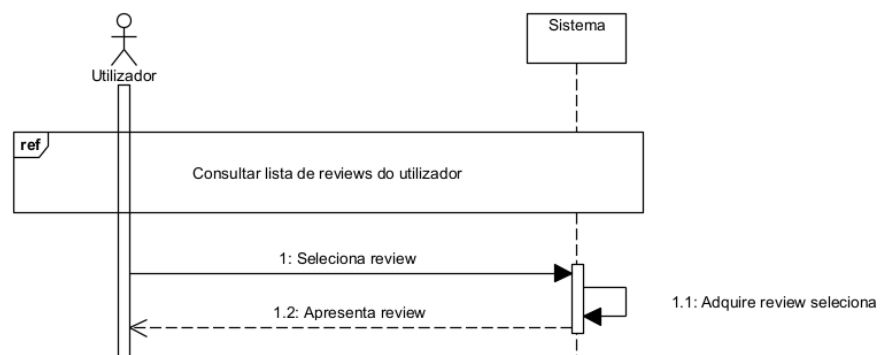


Figura 46. DS "Consultar review do utilizador"

16. Eliminar review do Utilizador

sd [Eliminar review do utilizador]

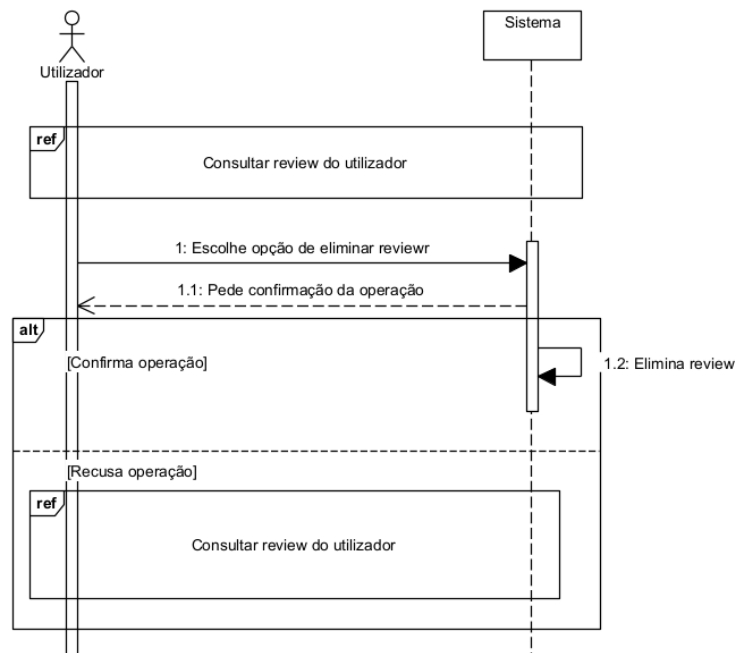


Figura 47. DS "Eliminar review do Utilizador"

17. Editar lista de favoritos

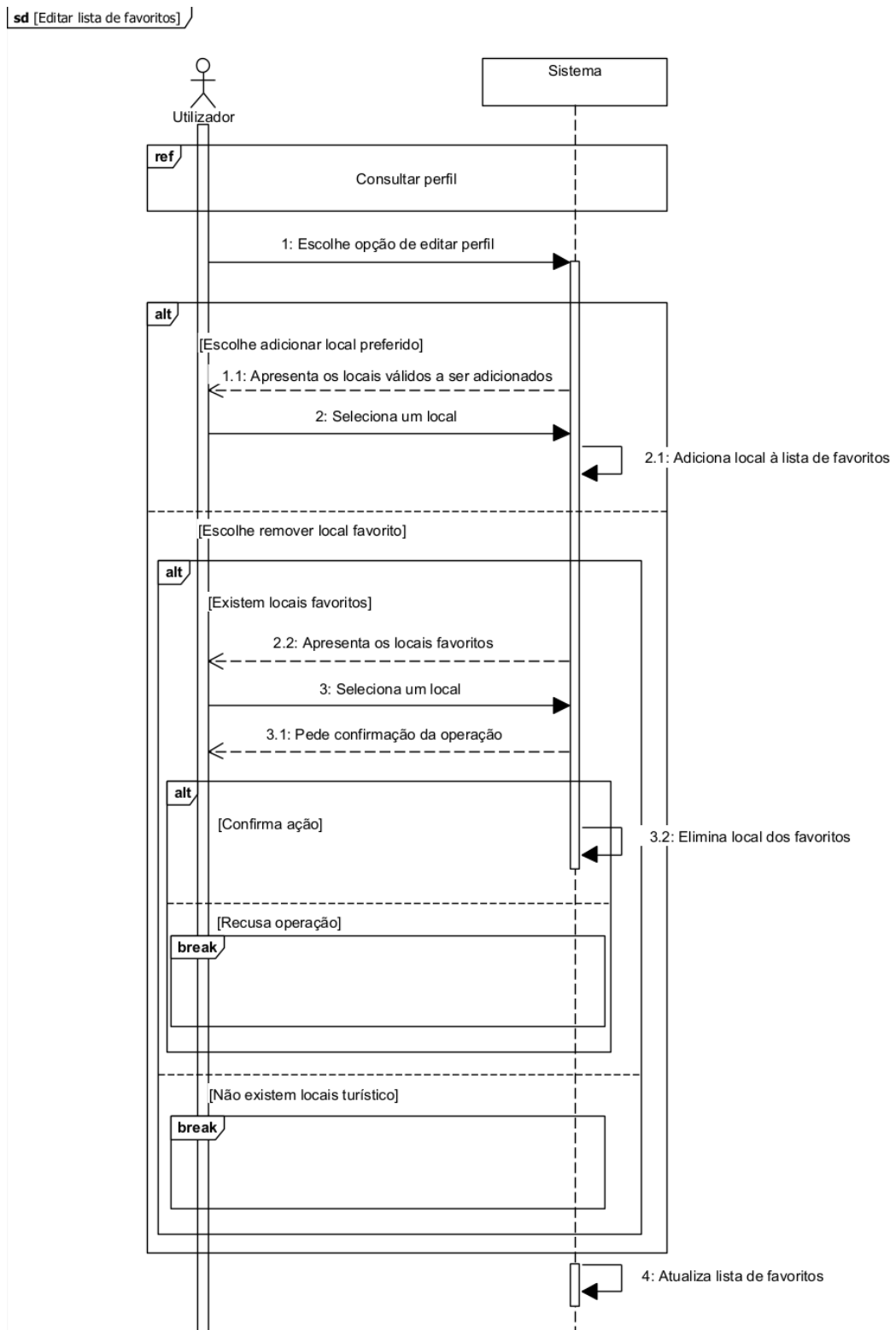


Figura 48. DS "Editar lista de favoritos"

3.4. Diagrama de componentes

Para criar uma melhor noção dos vários subsistemas que terão de ser implementados no projeto nas fases posteriores, assim como as suas interações e dependências, utilizamos como representação um diagrama de componentes.

A camada responsável pela apresentação do programa ao utilizador será **addandSeekUI**. Esta camada disponibilizará do uso da interface **laddandSeek**, implementada pelo sistema principal. O sistema principal será **addandSeek**, esta que por sua vez inclui os subsistemas **User subsystem** e **Places subsystems**, que, respetivamente, implementam a interface **IGestUsers** e **IGestPlaces** a tratar de dados de usuários e locais.

Por último, o movimento de dados entre a base de dados e o sistema, será da responsabilidade da camada de persistência, **CP**, que propõe uma interface **ICP** para o sistema executar as requisições e atualizações dos dados.

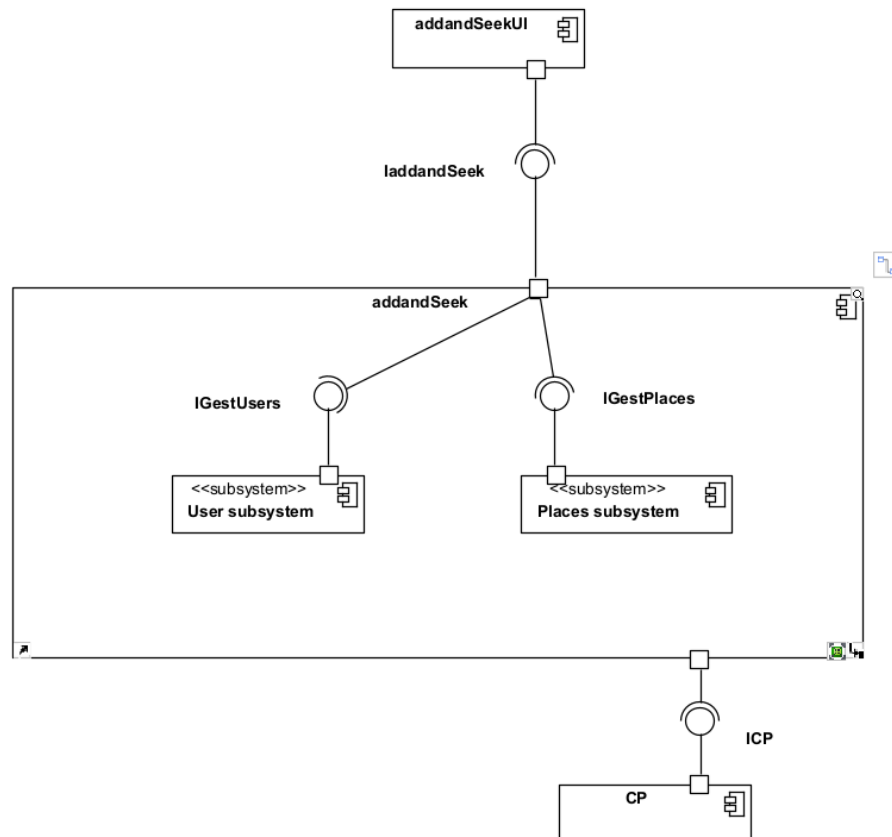


Figura 49. Diagrama de componentes

3.5. Diagrama de classes

A arquitetura do programa descreve as entidades principais como sendo, o **utilizador**, o **local**, e o **plano**.

O **utilizador** tem um nome, e-mail e password próprios, assim como um id único para o identificar mais facilmente. Além disso o utilizador terá uma lista de **planos** a ele associado, assim como uma lista de **reviews** e **locais favoritos**. Por fim, deverá ter as suas **coordenadas** ou **local** sempre atualizados para garantir uma boa filtragem dos locais de interesse.

O **local** terá a ele associado um **nome**, **descrição**, **coordenadas fixas** e uma lista das **reviews** a ele associado.

O **plano** deve ter um **dia da semana** fixo, um **intervalo de tempo** em que o itinerário irá decorrer e uma **lista de locais** e respetivas **durações de visita**.

Uma **review** tem um id, classificação e um comentário a ela associada.

Por fim, o programa terá uma facade onde se assegure o acesso e interação entre todas as entidades referidas.

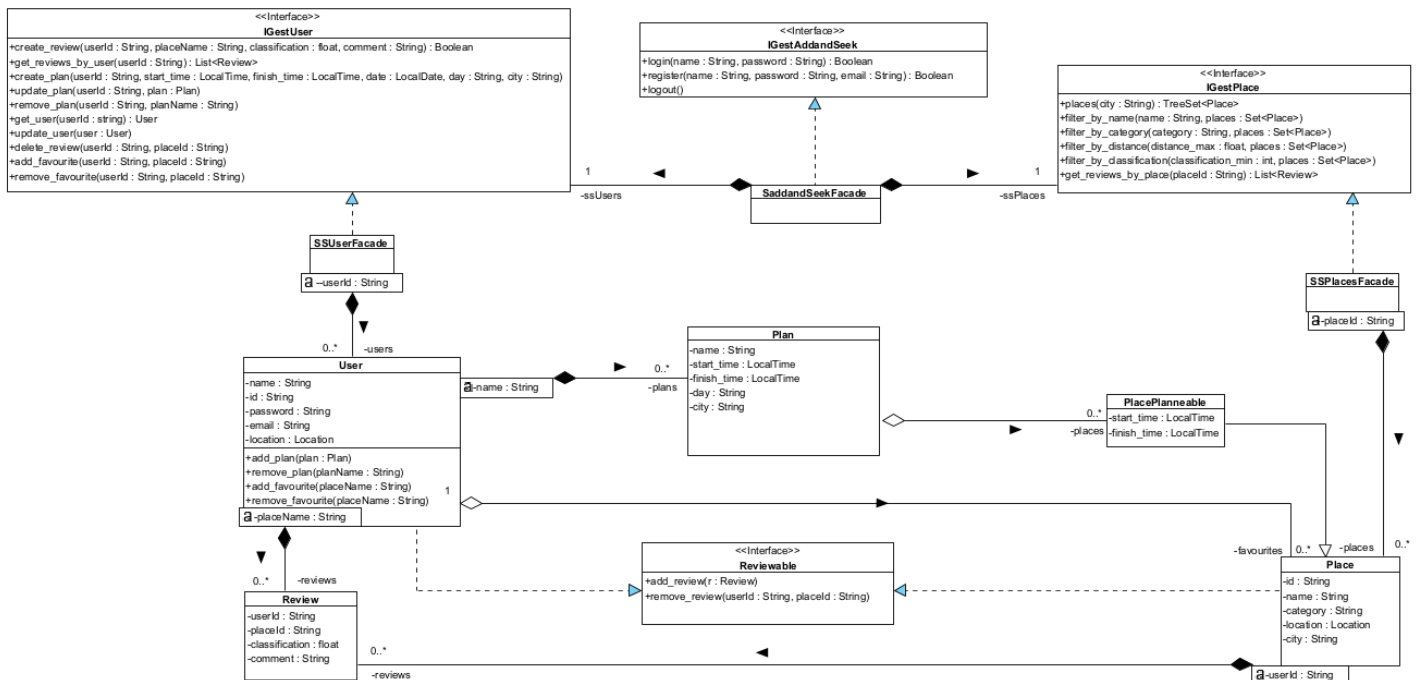


Figura 50. Diagrama de classes

4. Conceção do sistema de dados

A arquitetura da base de dados descreve o relacionamento das entidades e de como estas irão persistir no sistema. No seguinte diagrama apresentamos essas mesmas entidades necessárias para gerir os dados do sistema e a forma como estas se relacionam. As entidades são **users**, **places**, **reviews** e **plans**.

Users e **places** podem ser vistas como entidades principais, por outro lado, as **reviews** surgem da relação entre **users** e **places** visto que os **users** fazem uma **review** e **places** têm **reviews** associados. De forma semelhante a entidade **plans** está associado a **user**, visto que este é responsável pela criação desta.

4.1. Apresentação geral da estrutura do sistema de dados

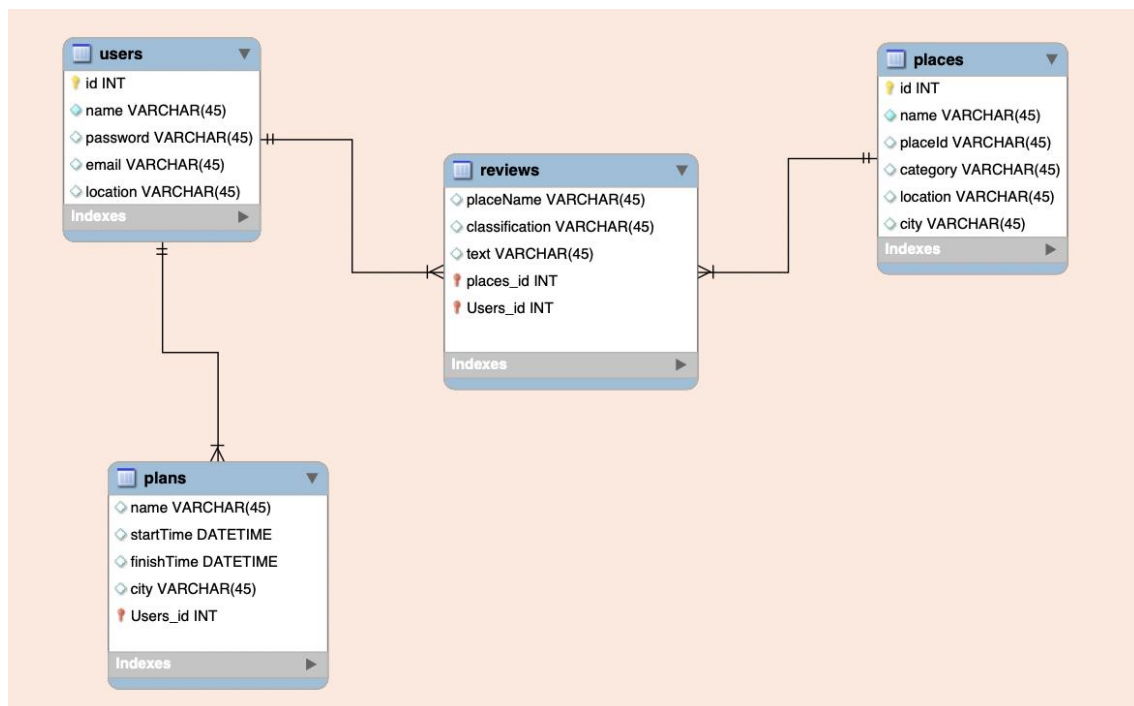


Figura 51. Diagrama da base de dados

4.2 Identificação e associação de atributos

Estudando o diagrama acima referido, podemos observar mais detalhadamente cada uma das entidades guardadas na base de dados.

Para cada entidade iremos descrever as suas chaves e os seus atributos com os respetivos domínios.

Chave primária: id

Entidade: users	
Atributo	Domínio
id	int
name	string
password	string
location	string

Tabela 1. Entidade User

Chave primária: id

Entidade: places	
Atributo	Domínio
Id	Int
Name	String
Category	String
Location	String
City	string

Tabela 2. Entidade Place

Chave estrangeira: user_id

Entidade: plans	
Atributo	Domínio
Name	String
startTime	dateTime
finishTime	DateTime
City	string

Tabela 3. Entidade Plan

Chave estrangeira: place_id, user_id

Entidade: reviews	
Atributo	Domínio
Id	Int
name	String
Password	String
location	string

Tabela 4. Entidade Review

5. Esboço das Interfaces do Sistema

Com o intuito de facilitar a compreensão do uso da aplicação, assim como, ajudar na construção da ferramenta que, além de útil, necessita ser intuitiva, construímos exemplos das interfaces do sistema.

add&SEEK, tendo como alvo toda a população, não exclusivamente, acima dos 14 anos, que esteja à procura de atividades para frequentar, necessita portanto de ter uma apresentação simples, moderna, intuitiva e que, apesar da grande quantidade de informação que possibilita ao utilizador de aceder, não o sature, convidando-o a retornar para uso futuro.

Para alcançar estes pontos, tivemos desde o início a consciência de que o design da aplicação teria de concretizar um ambiente amigável e que permita uma curva de aprendizagem de uso baixa.

Escolhemos esboçar as nossas mockups tendo em conta a vista de um browser de computador, ainda que sejam facilmente adaptadas para, por exemplo, uma aplicação de telemóvel.

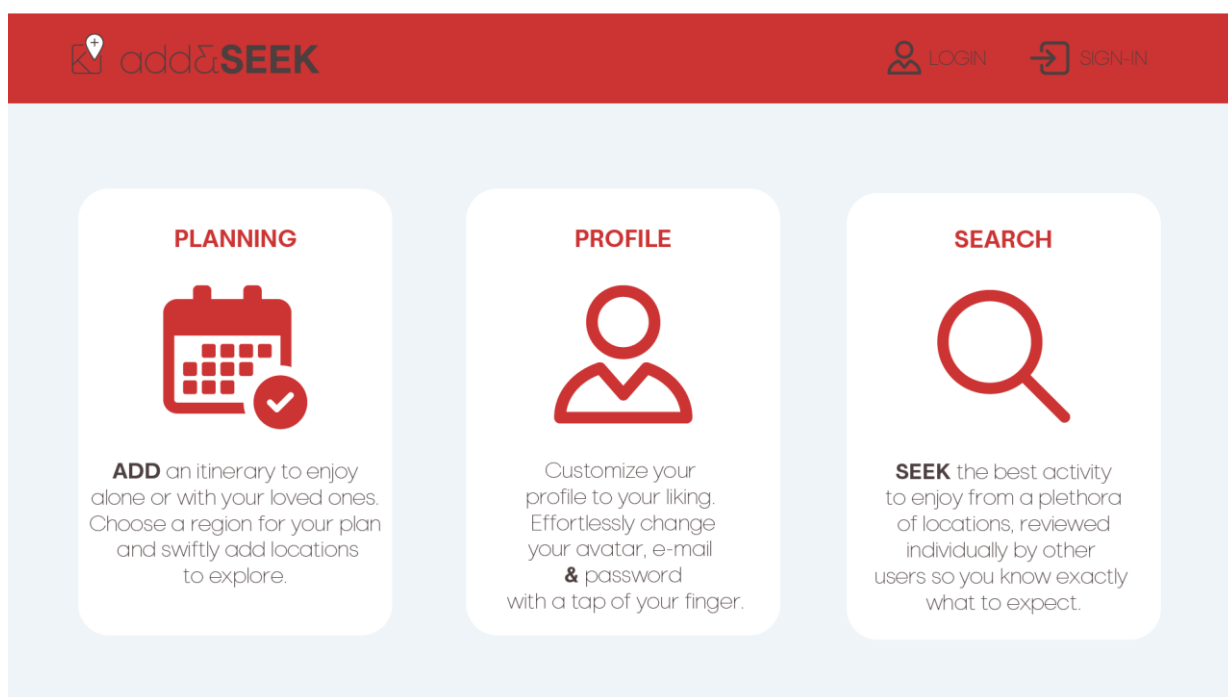


Figura 42. Página inicial

The screenshot shows the registration page of the add&SEEK website. The header is red with the add&SEEK logo on the left and 'LOGIN' and 'SIGN-IN' links on the right. The main content area is light blue and contains a white rounded rectangle titled 'REGISTER FORM'. Inside this form, there are four input fields: 'Name' with the value 'Gonçalo Braz', 'Email' with 'a93178@alunos.uminho.pt', 'Password' with masked characters '*****', and 'Repeat Password' with '*****'. Below the password fields is a checkbox labeled 'I've read and accept the terms and conditions'. At the bottom of the form is a green 'REGISTER' button.

add&SEEK

LOGIN SIGN-IN

REGISTER FORM

Name Gonçalo Braz

Email a93178@alunos.uminho.pt

Password *****

Repeat Password *****

☐ I've read and accept the terms and conditions

REGISTER

Figura 53. Página de registo

The screenshot shows the login page of the add&SEEK website. The header is red with the add&SEEK logo on the left and 'LOGIN' and 'SIGN-IN' links on the right. The main content area is light blue and contains a white rounded rectangle titled 'LOGIN'. Inside this form, there are two input fields: 'Email' with the value 'a93178@alunos.uminho.pt' and 'Password' with masked characters '***'. Below the password field is a red link labeled 'Forgot Password?'. At the bottom of the form is a red 'LOGIN' button.

add&SEEK

LOGIN SIGN-IN

LOGIN

Email a93178@alunos.uminho.pt

Password ***

[Forgot Password?](#)

LOGIN

Figura 54. Página de login

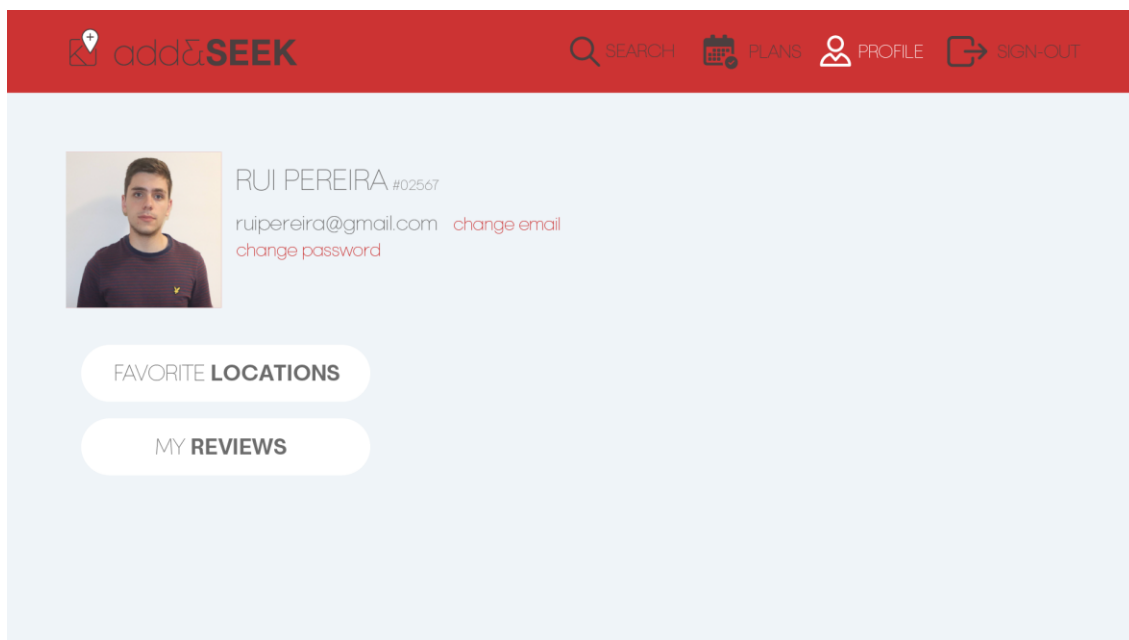


Figura 55. Página de perfil

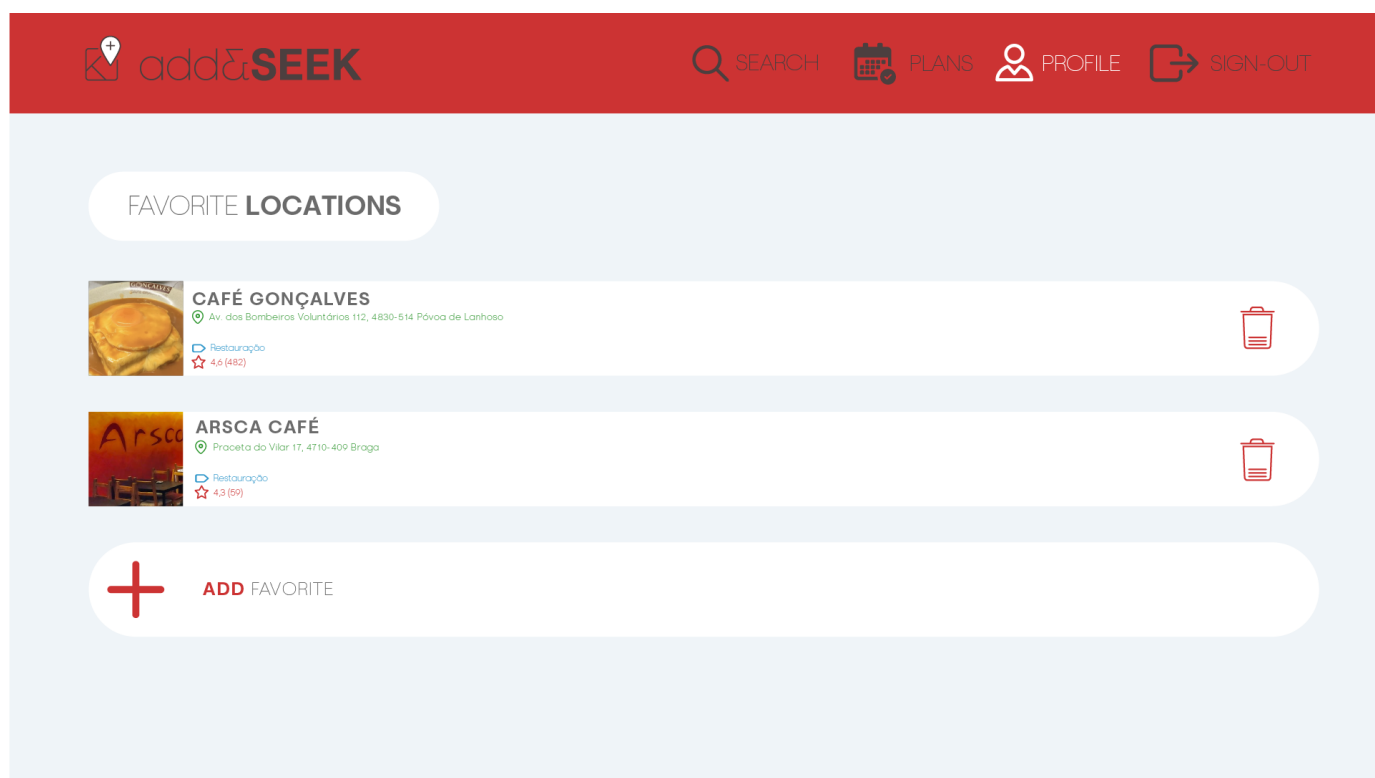


Figura 56. Página de favoritos do utilizador

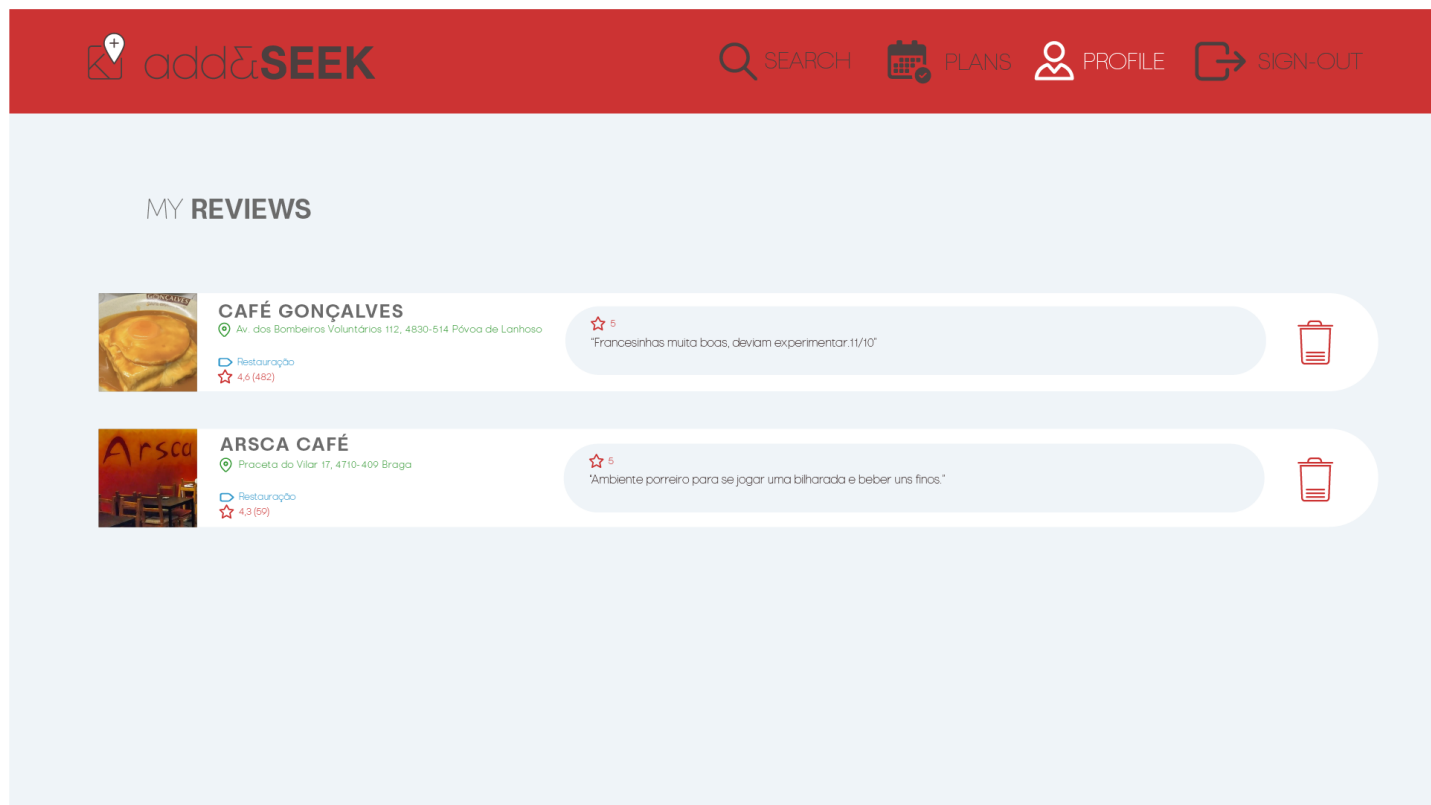


Figura 57. Página das reviews do utilizador

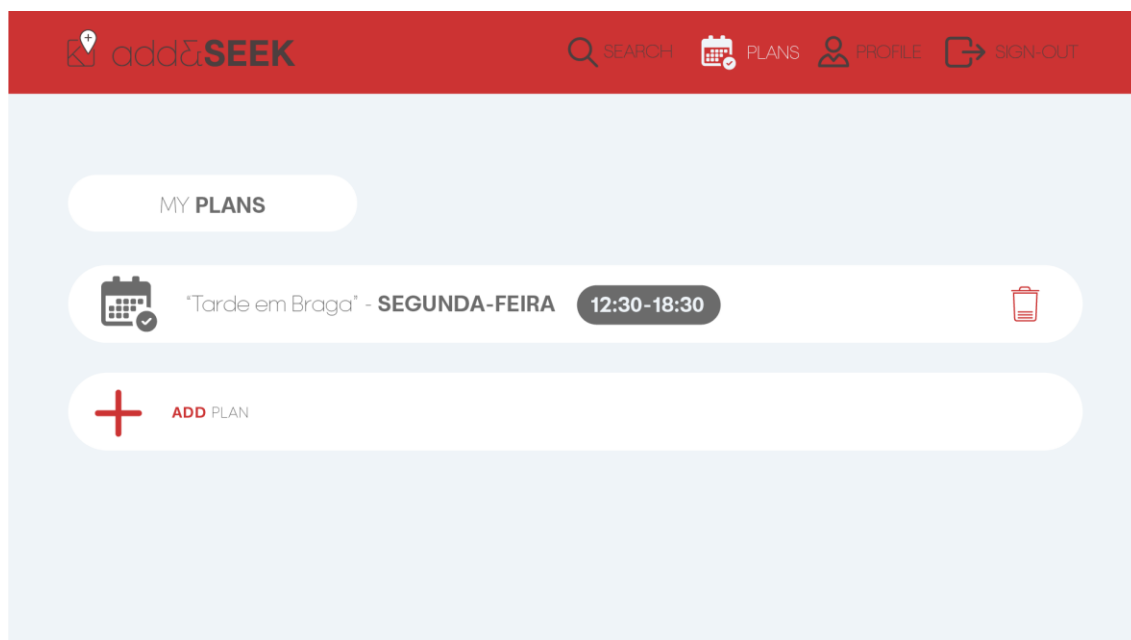


Figura 58. Página dos planos do utilizador

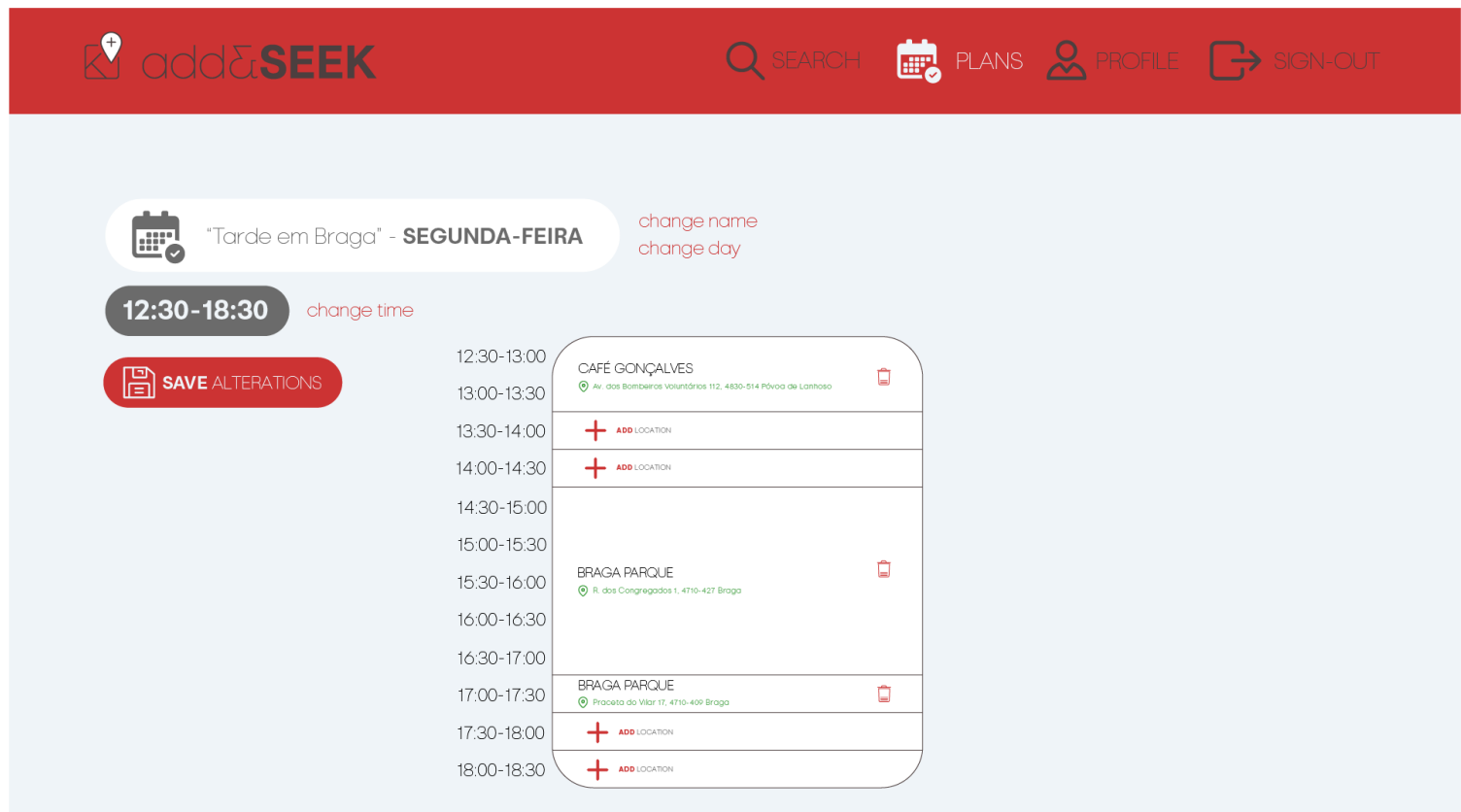


Figura 59. Página de edição de um plano

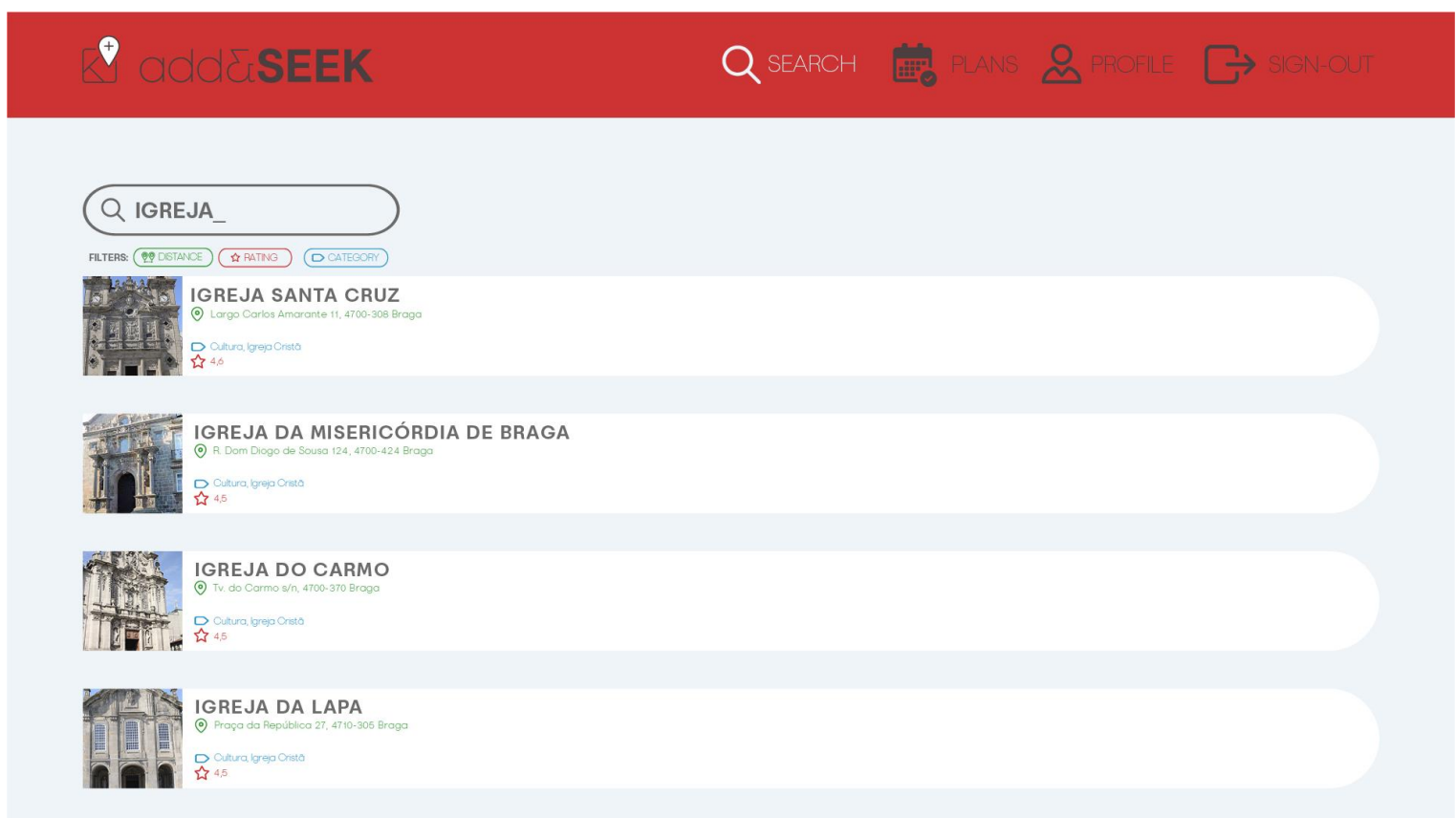


Figura 60. Página de pesquisa



IGREJA SANTA CRUZ

Largo Carlos Amarante 11, 4700-308 Braga

Cultura, Igreja Cristã

4,6

A Igreja de Santa Cruz é uma igreja de Braga, construída no século XVII, em estilo barroco maneirista, e possui no seu interior talha dourada invulgar. A nave, muito alta, é formada por uma abóbada de pedra esquadrelada. O interior da igreja foi do traço de Frei José de Santo António Vilaça. É de notar a talha dourada do órgão e dos púlpitos. O exterior é todo em pedra trabalhada com simetria central. Está localizada no Largo Carlos Amarante, no centro da cidade.

ADD REVIEW



—

VIEW REVIEWS

Figura 61. Página de um local

6. Conclusões e trabalho futuro

6.1. Conclusões

Concluída esta primeira fase do trabalho, cremos ter adquirido uma melhor compreensão sobre a importância de uma boa aplicação dos conhecimentos de desenvolvimento de sistemas de software, de modo a gerar um planeamento claro e que permita uma implementação futura mais guiada e eficaz, evitando refazê-la várias vezes sem qualquer tipo de orientação.

6.2. Trabalho futuro

Para a fase de implementação do projeto, esperamos que seja, primariamente, conseguida uma implementação sólida da aplicação seguindo os requisitos descritos anteriormente.

Algumas sugestões para implementações adicionais, após a conclusão destes requisitos mínimos seriam: a inclusão de preferências do utilizador, com um algoritmo que aproveitasse estes dados, sugerindo locais ao utilizador; inclusão de uma lista de amigos que, por sua vez, disponibilizaria a consulta dos locais favoritos destes; utilizadores reviewers, ou seja, cujas reviews tivessem um maior impacto nas páginas dos locais.

Referências

- PowerPoint's e material de estudo fornecido pela UC de Desenvolvimento de Sistemas de Software
- Google Maps