



**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

### **(A)Typical Food**

**Beatriz Rodrigues, Diogo Casal Novo**

**Francisco Neves, Gabriela Prata, João Carvalho**

Novembro, 2021

Data de Recepção	
Responsável	
Avaliação	
Observações	

## **(A)Typical Food**

**Beatriz Rodrigues, Diogo Casal Novo,**

**Francisco Neves, Gabriela Prata, João Carvalho**

Novembro, 2021

## Resumo

Este relatório foi elaborado no âmbito do desenvolvimento de uma aplicação que possibilita ter acesso a informações acerca de restaurantes de gastronomia diferente da tipicamente portuguesa, de forma a permitir o utilizador tomar decisões informadas e de forma a incentivá-lo a experimentar algo novo.

A primeira etapa correspondeu à contextualização do projeto e à definição da sua motivação e objetivos, da sua justificação e utilidade, da sua identidade, dos recursos necessários, da sua maqueta e das medidas de sucesso a ter em conta para obter o melhor resultado possível.

Na segunda etapa, o projeto começou a ganhar forma, com o levantamento e análise dos requisitos. Fixou-se a estratégia a seguir e enumeraram-se os requisitos que o sistema deveria cumprir quando implementado, procurando-se posteriormente a sua validação.

O passo seguinte tratou-se da especificação e modelação do software, o que envolveu a criação de um modelo de domínio e múltiplos diagramas de *Use Case* para deixar explícitos os aspetos estruturais e comportamentais da aplicação, através de UML.

Quando esta parte foi finalizada, procedeu-se ao desenvolvimento da base de dados criando-se, para este fim, um diagrama ER e um Modelo Lógico produzido com o MySQL. As entidades e as relações entre elas foram depois descritas em detalhe e também se procedeu ao cálculo do espaço em disco que uma entrada de cada entidade gastaria.

Por fim, esboçaram-se as interfaces de interação com o utilizador, de forma a prever a aparência da parte gráfica da aplicação, procurando obter um resultado esteticamente agradável, simplista e prático.

**Área de Aplicação:** Gastronomia, Cultura, Desenvolvimento de software, Engenharia de software

**Palavras-Chave:** MySQL, Base de dados, software

# Índice

Resumo	i
Índice	ii
Índice de Figuras	iv
Índice de Tabelas	v
1. Introdução	1
1.1. Contextualização	1
1.2. Apresentação do Caso de Estudo	2
1.3. Motivação e Objetivos	2
1.4. Estrutura do Relatório	2
1.5. Justificação e utilidade do sistema	3
1.6. Identidade do projeto	3
1.7. Recursos necessários	4
1.8. Maqueta do sistema	4
1.9. Definição das medidas de sucesso	5
1.10. Plano de desenvolvimento	6
2. Levantamento e análise de requisitos	7
2.1. Estratégia e método	7
2.2. Requisitos (funcionais e não funcionais) levantados	7
2.3. Validação dos requisitos estabelecidos	13
3. Especificação e Modelação do software	14
3.1. Apresentação geral da especificação	14
3.2. Aspetos estruturais	14
3.3. Aspetos comportamentais	16
4. Conceção do sistema de dados	25
4.1. Apresentação geral da estrutura	25
4.2. Descrição dos elementos e dos seus relacionamentos	26
5. Esboço das interfaces do sistema	30
5.1. Estrutura geral	30
5.2. Caracterização	31
6. Conclusões e Trabalho Futuro	37

6.1. Conclusões	37
6.2. Trabalho futuro	38
Referências	39
Lista de Siglas e Acrónimos	40

## Índice de Figuras

Figura 1: Maquete do Sistema	4
Figura 2: Protótipos do Sistema	5
Figura 3: Especificação das Tarefas	6
Figura 4: Diagrama de Gantt	6
Figura 5: Modelo de Domínio	15
Figura 6: Diagrama Use Case - Geral	16
Figura 7: Diagrama Use Case - Procurar Restaurante	17
Figura 8: Modelação Conceptual - Diagrama ER	25
Figura 9: Modelo Lógico	26
Figura 10: Ecrã Inicial da Aplicação	31
Figura 11: Menu de Criação de Conta	31
Figura 12: Menu de Autenticação	32
Figura 13: Ecrã Principal da Aplicação (após autenticação)	32
Figura 14: Pesquisa por Restaurante	33
Figura 15: Lista de Restaurantes	33
Figura 16: Menu de informação do Restaurante	34
Figura 17: Menu de Escrita de uma Avaliação	35
Figura 18: Definições	35
Figura 19: Website em Mobile	36

## Índice de Tabelas

Tabela 1: Identidade do Sistema	3
Tabela 2: Use Case - Criar Conta	18
Tabela 3: Use Case - Autenticar	19
Tabela 4: Use Case - Eliminar Conta	19
Tabela 5: Use Case - Alterar Informação Pessoal	20
Tabela 6: Use Case - Permitir Acesso à Localização	21
Tabela 7: Use Case - Procurar Restaurante Próximo	21
Tabela 8: Use Case - Procurar Restaurante de uma Nacionalidade	22
Tabela 9: Use Case - Procurar Restaurante com um Intervalo de Preços	22
Tabela 10: Use Case - Procurar um Restaurante através do nome	23
Tabela 11: Use Case - Adicionar Avaliação a um Restaurante	23
Tabela 12: Use Case - Selecionar Restaurante Aleatório	24
Tabela 13: Use Case - Terminar Sessão	24
Tabela 14: Caracterização das Entidades	28
Tabela 15: Espaço ocupado por cada entrada de um Restaurante	28
Tabela 16: Espaço ocupado por cada entrada de Comida	29
Tabela 17: Espaço ocupado por cada entrada de Cliente	29
Tabela 18: Espaço ocupado por cada entrada de Avaliação	29
Tabela 19: Espaço em Disco Total para uma Entrada de cada	29

# 1. Introdução

## 1.1. Contextualização

Nos tempos que correm, a alimentação deixou, sobretudo nos países mais desenvolvidos, de se tratar de uma mera questão de sobrevivência. Desde os primórdios da humanidade, cada cultura gerou o seu próprio estilo de culinária, o que, em muitas circunstâncias, trouxe a cada região um sabor específico que, de uma forma quase intuitiva, já associamos a cada país, como por exemplo, as comidas com uma utilização extensa de temperos à Índia ou a comida mais picante ao México. Apesar disto, para grande parte da população, muitas das gastronomias mundiais são ainda totalmente desconhecidas.

Atualmente, é essencial a experiencição das mais diversas culturas, não apenas por questões de prazer, como também de conhecimento e respeito pelos demais povos do nosso planeta. Surge, portanto, inevitavelmente, a necessidade de experiencição e de degustação da gastronomia das diversas culturas que poderá, indubitavelmente, combater muitos dos aspetos negativos que ainda hoje se encontram patentes na sociedade, tais como o racismo e a xenofobia.

Graças à globalização, é possível termos acesso aos preparados de outras culturas sem ser necessário deslocarmo-nos ao país ou região dos quais são provenientes, podendo, muitas vezes, encontrar-se o preparado desejado na zona em que se habita, principalmente, se esta se tratar de uma grande cidade.

Além de tudo isto, segundo o SEFSTAT, no ano de 2020 viviam em Portugal 662095 pessoas provenientes de outros países, o maior número de sempre da história do país. Destas 662095, 411809 provinham de países fora da Europa, pelo que, de países longínquos que, em grande parte dos casos, seria difícil lá retornarem mais do que uma vez por ano. Apesar do multiculturalismo, é também verdade, que muitas vezes apenas desejamos poder comer uma comida que já conhecemos e que nos faça lembrar de casa o que, em países distantes da nossa residência, poderá não ser muito fácil.



## 1.2. Apresentação do Caso de Estudo

(A) *Typical Food* é um software que disponibiliza informação acerca de restaurantes com gastronomia diferente da tipicamente portuguesa mais próximos do utilizador, em Braga.

Através das avaliações fornecidas por clientes e descrições dos restaurantes, o software poderá informar os utilizadores, além de uma descrição sucinta do restaurante e da sua gastronomia típica de um determinado país, dos seus preços habituais e da qualidade que outros utilizadores encontraram nele, para que, desta forma, um possível cliente possa tomar uma decisão informada e não apenas pelo que o restaurante lhe poderá parecer.

Desta forma, visto que, a qualidade dos serviços fornecidos por um determinado restaurante será comentada e avaliada por diversos utilizadores, formar-se-á uma comunidade de confiança e melhor informada acerca dos estabelecimentos disponíveis e se estes apresentam uma boa relação qualidade/preço.

## 1.3. Motivação e Objetivos

A falta de informação inibe, muitas vezes, as pessoas de procurarem novas experiências o que, em determinados casos, poderia combater preconceitos enraizados na sociedade que, com a evolução do multiculturalismo, se têm vindo a demonstrar como sendo errados. Existem ainda muitas pessoas desinformadas acerca de gastronomias de países com muito sabor para oferecer e muitas histórias que se podem explicar com base na comida.

O número de estrangeiros em Portugal e, nomeadamente, em Braga tem vindo a aumentar de ano para ano, muitos deles em situações difíceis, nas quais, apenas uma visita a um sítio que os faça lembrar de casa poderá fazer uma diferença tremenda no dia a dia.

Apesar de não ser o principal alvo do (A) *Typical Food*, este software poderá ainda ajudar novos restaurantes que, devido ao seu carácter diferente do tradicionalmente português, poderão passar por dificuldades na angariação de novos clientes, mesmo apresentando um bom serviço.

## 1.4. Estrutura do Relatório

O presente relatório aborda as várias etapas do planeamento da aplicação, desde a sua contextualização, motivação e objetivos, até às circunstâncias que a justificam e que determinam a sua utilidade para os utilizadores, a sua identidade, os recursos necessários para a concretizar e a sua maquete.

Para além disso, efetua-se um levantamento e análise de requisitos, a especificação e modelação do software, a construção da base de dados e das interfaces do sistema.

Por fim, é também incluída as conclusões resultantes de todo este processo.

## 1.5. Justificação e utilidade do sistema

(A)Typical Food é uma aplicação que pretende estabelecer uma ponte entre o utilizador e um ramo de possibilidades de gastronomias de outras culturas por conhecer, defendendo os benefícios que provêm de arriscar e de experimentar algo diferente. No entanto, este pode se sentir desconfortável, seja por temer que desperdice o dinheiro em algo que acabe por não gostar ou seja por simplesmente ser alguém habituado à sua rotina.

O objetivo é assim que o utilizador tenha informação suficiente para tomar uma decisão informada acerca do que lhe apetece experimentar, tendo acesso às avaliações de outros utilizadores que já o fizeram e a uma estimativa do dinheiro a gastar e, desta forma, sentindo-se mais seguro e menos perdido num contexto que desconhece.

## 1.6. Identidade do projeto



**(A)TYPICAL FOOD**

Nome	(A)Typical Food
Prazo	24/01/2022
Faixa Etária	13+
Orçamento	10000€
Descrição	Aplicação que permite consultar restaurantes com gastronomia diferente da tipicamente portuguesa na cidade de Braga. Pode obter a localização do utilizador e fazer recomendações tendo-a em consideração e também indicar o caminho para o lugar pretendido. Para além disso, permite filtrar restaurantes pelo nome e por nacionalidade.

Tabela 1: Identidade do Sistema

## 1.7. Recursos necessários

Para o funcionamento da aplicação, será essencial obter as posições do utilizador e dos restaurantes de gastronomias de outras culturas na sua proximidade que existem na cidade de Braga. Para isso, pretende-se utilizar *Bing Maps*.

Para além disso, também é indispensável uma base de dados para armazenar a informação recolhida. Esta será feita utilizando o sistema *MySQL* por ser fácil de instalar e utilizar, por ser o segundo mais utilizado do mercado e porque consome poucos recursos. Para auxiliar o processo, será criado um diagrama ER através do programa *brModelo*.

Relativamente a recursos humanos, consideram-se necessários dois engenheiros de software, dois programadores e um designer gráfico.

Por fim, será usado o *Figma* para criar os *mockups* da aplicação.

## 1.8. Maqueta do sistema

O nosso sistema divide-se em duas vertentes principais: a primeira corresponde ao *Back-End*, que se trata da parte da aplicação responsável por gerir as funcionalidades disponíveis, por armazenar os dados pessoais e credenciais dos utilizadores e por conter também as informações acerca dos restaurantes multiculturais disponíveis na cidade da Braga; a segunda corresponde ao *Front-End*. Esta camada permite ao utilizador tomar conhecimento das funcionalidades que a aplicação tem para oferecer e interagir com o sistema de forma a aproveitá-las.

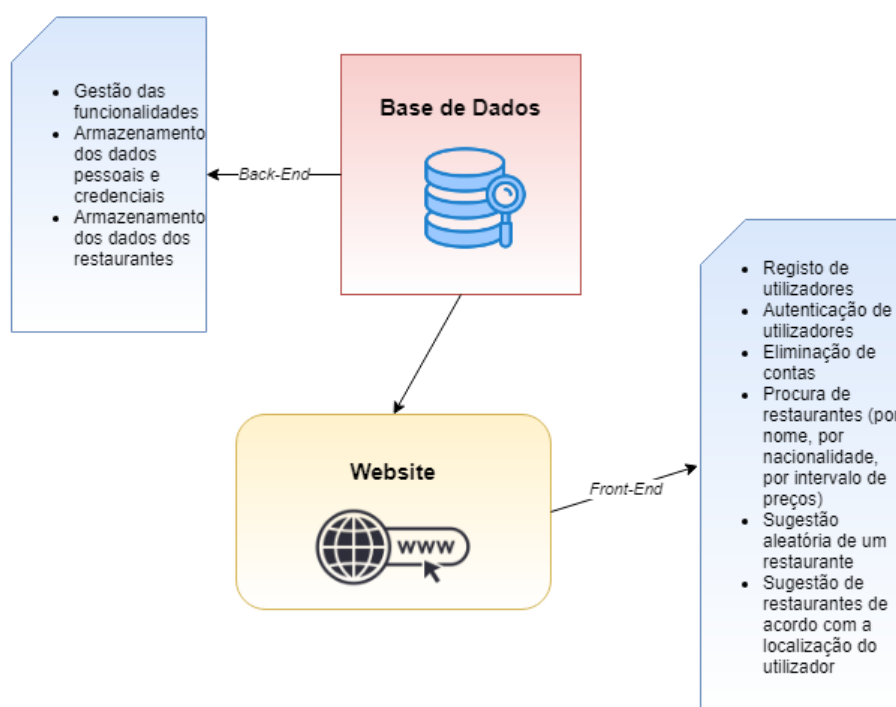


Figura 1: Maquete do Sistema



Figura 2: Protótipos do Sistema

## 1.9. Definição das medidas de sucesso

É pretendido que *(A)Typical Food* seja uma aplicação com um número elevado de utilizadores e que estes interajam regularmente com a aplicação, deixando avaliações aos restaurantes que visitarem de forma a melhorarem a experiência de novos utilizadores.

Assim, com *(A)Typical Food* é esperado acompanhar os utilizadores a novas experiências da forma mais confortável, eficiente e segura.

## 1.10. Plano de desenvolvimento

		Nome	Duração	Início	Fim	Antecessores	Nomes dos Recursos
1		<b>Desenvolvimento do (A)Typical Food</b>	<b>71 dias?</b>	<b>18-10-2021 8:00</b>	<b>24-01-2022 17:00</b>		
2		<b>Planeamento</b>	<b>31 dias?</b>	<b>18-10-2021 8:00</b>	<b>29-11-2021 17:00</b>		
3		<b>Levantamento e Análise de Requisitos</b>	<b>16 dias?</b>	<b>18-10-2021 8:00</b>	<b>08-11-2021 17:00</b>		
4		Reuniões com cidadãos de Braga	2 dias?	18-10-2021 8:00	19-10-2021 17:00		Cidadão;Engenheiro de Software
5		Reuniões com donos de estabelecimentos	2 dias?	20-10-2021 8:00	21-10-2021 17:00	4	Dono estabelecimento;Engenheiro de Software
6		Levantamento dos requisitos	5 dias?	21-10-2021 8:00	27-10-2021 17:00		Engenheiro de Software
7		<b>Validação dos requisitos</b>	<b>4 dias?</b>	<b>28-10-2021 8:00</b>	<b>02-11-2021 17:00</b>		
8		Reuniões com cidadãos de Braga	2 dias?	28-10-2021 8:00	29-10-2021 17:00	6	Cidadão;Engenheiro de Software
9		Reuniões com donos de estabelecimentos	2 dias?	01-11-2021 8:00	02-11-2021 17:00	8	Dono estabelecimento;Engenheiro de Software
10		Documentação	12 dias?	22-10-2021 8:00	08-11-2021 17:00		Engenheiro de Software
11		<b>Especificação e Modelação do Softwa...</b>	<b>5 dias?</b>	<b>09-11-2021 8:00</b>	<b>15-11-2021 17:00</b>	<b>3</b>	
12		Modelação do Diagrama de Domínio	1 dia?	09-11-2021 8:00	09-11-2021 17:00		Engenheiro de Software
13		Modelação dos Diagramas de Use Case	3 dias?	10-11-2021 8:00	12-11-2021 17:00	12	Engenheiro de Software
14		Documentação	5 dias?	09-11-2021 8:00	15-11-2021 17:00		Engenheiro de Software
15		<b>Conceção da Base de Dados</b>	<b>4 dias?</b>	<b>16-11-2021 8:00</b>	<b>19-11-2021 17:00</b>	<b>11</b>	
16		Elaboração do Diagrama ER	2 dias?	16-11-2021 8:00	17-11-2021 17:00		Engenheiro de Software
17		Elaboração do Modelo Lógico	1 dia?	18-11-2021 8:00	18-11-2021 17:00	16	Engenheiro de Software
18		Documentação	4 dias?	16-11-2021 8:00	19-11-2021 17:00		Engenheiro de Software
19		<b>Interfaces</b>	<b>4 dias?</b>	<b>22-11-2021 8:00</b>	<b>25-11-2021 17:00</b>	<b>15</b>	
20		Desenho de Mockups	4 dias?	22-11-2021 8:00	25-11-2021 17:00		Designer Gráfico;Engenheiro de Software
21		Documentação	4 dias?	22-11-2021 8:00	25-11-2021 17:00		Engenheiro de Software
22		Acertos finais na documentação	2 dias?	26-11-2021 8:00	29-11-2021 17:00	19	Engenheiro de Software
23		<b>Construção</b>	<b>36 dias?</b>	<b>06-12-2021 8:00</b>	<b>24-01-2022 17:00</b>	<b>2</b>	
24		<b>Arquitetura do Sistema</b>	<b>2 dias?</b>	<b>06-12-2021 8:00</b>	<b>07-12-2021 17:00</b>		
25		Construção da arquitetura do Sistema	1 dia?	06-12-2021 8:00	06-12-2021 17:00		Programador
26		Validação da arquitetura do sistema	1 dia?	07-12-2021 8:00	07-12-2021 17:00	25	Programador
27		<b>Criação da Base de Dados</b>	<b>4 dias?</b>	<b>08-12-2021 8:00</b>	<b>13-12-2021 17:00</b>		
28		Implementação da Base de Dados	3 dias?	08-12-2021 8:00	10-12-2021 17:00	26	Programador
29		Povoamento da Base de Dados	1 dia?	13-12-2021 8:00	13-12-2021 17:00	28	Programador
30		Programação da Camada de Dados	8 dias?	14-12-2021 8:00	23-12-2021 17:00	29	Programador
31		Programação da Camada de Negócio	7 dias?	27-12-2021 8:00	04-01-2022 17:00	30	Programador
32		Programação da Camada de Interface	5 dias?	05-01-2022 8:00	11-01-2022 17:00	31	Programador
33		<b>Validação</b>	<b>5 dias?</b>	<b>12-01-2022 8:00</b>	<b>18-01-2022 17:00</b>	<b>32</b>	
34		Realização de Testes	5 dias?	12-01-2022 8:00	18-01-2022 17:00		Programador
35		Correção de Erros	5 dias?	12-01-2022 8:00	18-01-2022 17:00		Programador
36		Documentação	36 dias?	06-12-2021 8:00	24-01-2022 17:00		Programador

Figura 3: Especificação das Tarefas

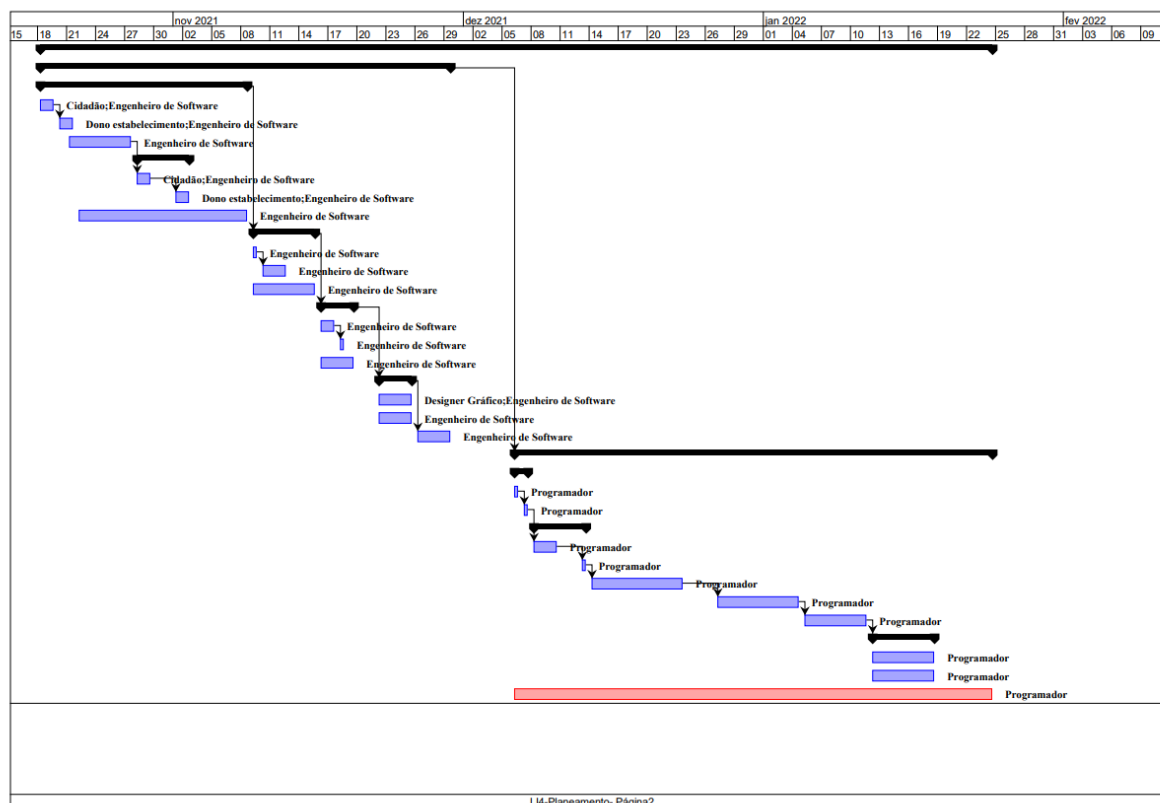


Figura 4: Diagrama de Gantt

## 2. Levantamento e análise de requisitos

### 2.1. Estratégia e método

O desenvolvimento da aplicação foi centrado em adaptá-la de forma a agradar o utilizador acima de tudo e favorecer os donos dos restaurantes abrangidos. Assim, efetuaram-se algumas reuniões com possíveis clientes da cidade de Braga, juntamente com alguns donos de estabelecimentos multiculturais, com o objetivo de recolher informações acerca das funcionalidades que estes gostariam de ver implementadas no projeto referido.

### 2.2. Requisitos (funcionais e não funcionais) levantados

#### 2.2.1. Requisitos funcionais

##### 2.2.1.1. Criação de conta

**Função:** Permitir que o utilizador possa utilizar a aplicação.

**Descrição:** Após a criação de uma conta através de um email, nome, de uma password, de um número de telemóvel, data de nascimento e nome de utilizador, este passa a poder utilizar a aplicação.

**Inputs:** Email, password, nome, nome de utilizador, número de telemóvel e data de nascimento.

**Source:** Input introduzido pelo utilizador.

**Output:** Mensagem de sucesso ou de insucesso.

**Ação:** O sistema solicita ao utilizador os dados necessários durante o registo e deve armazenar a informação referida na base de dados.

**Requerimentos:** O email, o número de telemóvel e o nome de utilizador não podem estar em

utilização por outro utilizador.

**Pré-condição:** Nenhuma.

**Pós-condição:** O utilizador está registado no sistema.

### 2.2.1.2. Autenticação

**Função:** Permite que o utilizador inicie sessão na aplicação.

**Descrição:** Após validação dos seus dados de acesso, o utilizador autenticado pode utilizar a aplicação.

**Inputs:** Email e password do utilizador.

**Source:** Input inserido pelo utilizador.

**Output:** Mensagem de sucesso ou de insucesso.

**Ação:** O sistema deve solicitar ao utilizador que introduza os seus dados de acesso. Deve verificar se as credenciais correspondem a um utilizador existente. De seguida, valida se estas estão corretas: em caso afirmativo, o utilizador é autenticado, caso contrário informa o utilizador que estas não existem ou não combinam e permite que este tente novamente.

**Requerimentos:** As credenciais devem estar armazenadas no sistema.

**Pré-condição:** O utilizador deve ter conta criada e não ter sessão iniciada.

**Pós-condição:** O utilizador está autenticado no sistema.

### 2.2.1.3. Editar informação pessoal

**Função:** Permite ao utilizador atualizar a sua informação pessoal.

**Descrição:** O utilizador pode editar as suas informações pessoais.

**Inputs:** Informações pessoais do utilizador.

**Source:** Input do utilizador.

**Output:** Mensagem de sucesso ou de insucesso.

**Ação:** O sistema apresenta ao utilizador os seus dados e dá-lhe a possibilidade de fazer as alterações pretendidas. O utilizador deve de seguida confirmar as mudanças efetuadas e o sistema valida se estes entram em conflito com outra conta já existente: em caso afirmativo, estas são descartadas, em caso negativo os dados são alterados e é feita a substituição na base de dados.

**Requerimentos:** O nome de utilizador, o email e o número de telemóvel não podem estar em utilização noutra conta.

**Pré-condição:** O utilizador tem sessão iniciada.

**Pós-condição:** Os dados do utilizador são modificados.

#### 2.2.1.4. Permitir acesso à localização do utilizador

**Função:** Permitir ao sistema obter a localização do utilizador.

**Descrição:** Para procurar os restaurantes próximos e calcular os caminhos para eles, o sistema precisa de ter a localização do utilizador.

**Inputs:** Seleção do utilizador.

**Source:** Input do utilizador.

**Output:** Não existe.

**Ação:** O utilizador seleciona se pretende permitir o acesso à localização. O sistema atualiza de acordo com essa informação.

**Requerimentos:** A API de localização está funcional.

**Pré-condição:** O utilizador tem sessão iniciada.

**Pós-condição:** O sistema obtém a localização do utilizador.

#### 2.2.1.5. Eliminar conta

**Função:** Permite ao utilizador eliminar a conta.

**Descrição:** Caso o utilizador já não queira utilizar a aplicação, pode apagar a sua conta do sistema.

**Inputs:** Inserção dos dados de autenticação.

**Source:** Input inserido pelo utilizador.

**Output:** Mensagem de sucesso/insucesso.

**Ação:** O sistema solicita que o utilizador insira as suas credenciais de autenticação: caso estejam corretas, apaga a conta do sistema, caso contrário descarta a tentativa.

**Requerimentos:** Os dados de autenticação inseridos têm de estar corretos.

**Pré-condição:** O utilizador tem sessão iniciada.

**Pós-condição:** A conta do utilizador está apagada do sistema.

#### 2.2.1.6. Procurar restaurante na proximidade

**Função:** Permite ao utilizador procurar restaurantes próximos de si.

**Descrição:** Com o acesso à sua localização, a aplicação sugere ao utilizador os restaurantes perto de si.

**Inputs:** Localização do utilizador.

**Source:** Bing Maps.



**Output:** Lista ordenada por proximidade de restaurantes ou mensagem de insucesso.

**Ação:** O sistema acede à localização do utilizador. Caso o acesso ocorra sem problemas, fornece-lhe a lista de restaurante próximos de si; caso contrário informa o utilizador que ocorreu um erro. Por outro lado, se não existirem restaurantes próximos de si o sistema informa que não existem restaurantes próximos disponíveis naquele momento.

**Requerimentos:** O sistema consegue aceder à localização do utilizador.

**Pré-condição:** O utilizador tem sessão iniciada e permite o acesso à localização.

**Pós-condição:** O utilizador recebe uma lista de restaurantes na sua proximidade.

#### 2.2.1.7. Procurar restaurante de uma nacionalidade específica

**Função:** Permite ao utilizador filtrar os restaurantes de acordo com uma nacionalidade.

**Descrição:** O utilizador pode expressar que deseja visitar um restaurante com gastronomia de uma nacionalidade específica e o sistema fornece-lhe a lista de opções disponíveis.

**Inputs:** Nacionalidade selecionada.

**Source:** Input selecionado pelo utilizador.

**Output:** Lista ordenada de restaurantes pela classificação média.

**Ação:** O sistema oferece ao utilizador as opções de nacionalidades das quais existem restaurante típicos e este deve selecionar a que lhe interessa. Após a seleção, recebe uma lista ordenada de acordo com a classificação média atribuída.

**Requerimentos:** O utilizador seleciona uma nacionalidade.

**Pré-condição:** O utilizador tem sessão iniciada.

**Pós-condição:** O utilizador recebe uma lista dos restaurantes de uma dada nacionalidade.

#### 2.2.1.8. Procurar restaurante dentro de um intervalo de preços

**Função:** Permite ao utilizador filtrar os restaurantes de acordo com um intervalo de preços pretendido.

**Descrição:** O utilizador pode indicar que pretende obter apenas sugestões de restaurantes cujos preços típicos estejam dentro de um dado intervalo.

**Inputs:** Intervalo de preços.

**Source:** Input selecionado pelo utilizador.

**Output:** Lista ordenada pela classificação média dos restaurantes contidos no intervalo de preços selecionado pelo utilizador.

**Ação:** O sistema apresenta ao utilizador as opções de intervalos de preço disponíveis e este seleciona o pretendido. De seguida, o sistema apresenta-lhe uma lista ordenada pela

classificação média das opções que cumprem o seu pedido.

**Requerimentos:** O utilizador selecciona um intervalo de preços.

**Pré-condição:** O utilizador tem sessão iniciada.

**Pós-condição:** O utilizador recebe uma lista de opções de restaurantes dentro do intervalo de preço seleccionado.

### 2.2.1.9. Procurar restaurante através do nome

**Função:** Permite ao utilizador filtrar restaurantes a partir do nome.

**Descrição:** O utilizador pode procurar restaurantes com um determinado nome.

**Inputs:** Nome do restaurante pretendido.

**Source:** Input do utilizador.

**Output:** Lista ordenada pela classificação média dos restaurantes com o nome inserido ou mensagem de insucesso.

**Ação:** O sistema recebe o nome do restaurante pretendido proveniente do utilizador e verifica se existem restaurantes com esse nome: em caso afirmativo, apresenta uma lista ordenada pela classificação média das opções; em caso negativo, informa-o que não existem.

**Requerimentos:** O utilizador insere um nome.

**Pré-condição:** O utilizador tem sessão iniciada.

**Pós-condição:** O utilizador recebe uma lista ordenada pela classificação média dos restaurantes com o nome pretendido.

### 2.2.1.10. Sugerir um restaurante aleatório

**Função:** Permite ao utilizador receber uma sugestão aleatória.

**Descrição:** O utilizador pode receber uma sugestão aleatória de um restaurante contido no sistema.

**Inputs:** Não existe.

**Source:** Não existe.

**Output:** Restaurante gerado aleatoriamente.

**Ação:** O sistema fornece um restaurante contido no sistema aleatoriamente caso o utilizador o pretenda.

**Requerimentos:** Existem restaurantes na base de dados.

**Pré-condição:** O utilizador tem sessão iniciada.

**Pós-condição:** O utilizador recebe uma sugestão de um restaurante aleatório.

### 2.2.1.11. Adicionar avaliação a um restaurante

**Função:** Permite ao utilizador avaliar um restaurante.

**Descrição:** Um utilizador pode atribuir uma avaliação a um restaurante, no intervalo de 1 a 5 estrelas. Pode também acompanhar a avaliação de um texto.

**Inputs:** Avaliação do utilizador.

**Source:** Input selecionado e introduzido pelo utilizador.

**Output:** Mensagem de confirmação.

**Ação:** O utilizador atribui a um restaurante uma avaliação entre 1 e 5 estrelas e pode ou não acompanhá-la com uma crítica em forma de texto. O sistema recebe a avaliação e envia uma mensagem de confirmação.

**Requerimentos:** O restaurante existe na base de dados.

**Pré-condição:** O utilizador tem sessão iniciada.

**Pós-condição:** O sistema recebe a avaliação do utilizador.

### 2.2.1.12. Indicar o caminho para o restaurante selecionado

**Função:** Permite ao utilizador receber o caminho para o restaurante selecionado.

**Descrição:** Após selecionar um restaurante, recebe um caminho para conseguir visitá-lo, a partir da sua localização atual.

**Inputs:** Localização atual do utilizador.

**Source:** Bing-Maps.

**Output:** Caminho para o restaurante selecionado ou mensagem de erro.

**Ação:** Após selecionar o restaurante pretendido, o sistema acede à localização atual do utilizador e sugere-lhe um caminho que pode tomar para alcançar a sua opção. Caso não consiga aceder à sua localização, fornece uma mensagem de erro.

**Requerimentos:** O sistema tem acesso à localização do utilizador.

**Pré-condição:** O utilizador tem sessão iniciada e permite o acesso à localização.

**Pós-condição:** Caminho para o restaurante selecionado.

### 2.2.1.13. Permitir ao utilizador terminar sessão

**Função:** Permite ao utilizador terminar sessão na aplicação.

**Descrição:** Quando terminar de utilizar a aplicação, o utilizador pode terminar sessão.

**Inputs:** Seleção do utilizador.

**Source:** Input do utilizador.

**Output:** Menu de início de sessão.

**Ação:** O utilizador pressiona um botão que termina a sessão iniciada atualmente.

**Requerimentos:** Nenhum.

**Pré-condição:** O utilizador tem sessão iniciada.

**Pós-condição:** O utilizador deixa de ter sessão iniciada.

### 2.2.2. Requisitos não funcionais

Os requisitos não funcionais são as restrições e requerimentos impostos no sistema. Assim, estão relacionados com a sua sustentabilidade, performance, segurança, confiança, entre outros atributos. Definiram-se os seguintes para a aplicação:

- A aplicação deve ser simples e o mais acessível possível a todos os utilizadores.
- A privacidade dos dados dos utilizadores deve ser assegurada.
- A aplicação deve ser atualizada regularmente, tendo em vista potenciais novos restaurantes.
- O software deve funcionar nos diversos browsers existentes.
- O sistema deve ser capaz de aceder eficientemente à localização do utilizador através de uma API.

## 2.3. Validação dos requisitos estabelecidos

De forma a garantir que os requisitos eram relevantes e que não existiam outros em falta, procederam-se a novas reuniões com o grupo inicial de cidadãos da cidade de Braga e de donos de estabelecimentos multiculturais.

Face à grande taxa de aprovação recebida, foi possível avançar para a especificação e modelação do software.

## 3. Especificação e Modelação do software

### 3.1. Apresentação geral da especificação

Na nossa aplicação, apenas o utilizador é capaz de interagir com o sistema. Este recebe inicialmente um ecrã com as opções de criar conta ou efetuar autenticação. Deve ainda permitir a localização para o funcionamento completo da aplicação. Caso não o faça, ver-se-á limitado, uma vez que o sistema não lhe poderá indicar os restaurantes na sua proximidade nem guiá-lo até ao seu destino. Após a autenticação, é possível começar a consultar os restaurantes e a usufruir das potencialidades do sistema, tais como filtrar os restaurantes por intervalos de preço, pelo seu nome, entre outras. Caso se encontre insatisfeito ou já não tenha utilidade para ela, o utilizador poderá apagar a sua conta.

De forma a ser possível compreender a arquitetura da aplicação de uma forma mais simples e explícita, recorreu-se a UML para produzir modelação de domínio e de Use Cases.

### 3.2. Aspetos estruturais

O Modelo de Domínio tem a função de representar as classes conceptuais significativas no domínio do problema. Desta forma, deverá fornecer uma *framework* conceptual que permita raciocinar sobre o problema, tornando-se dessa forma a base para a análise de requisitos.

Para a sua construção, primeiramente, deverão ser identificadas as entidades envolvidas no sistema, em seguida, deverão ser estabelecidas as relações entre elas e, por fim, deverá ser definida a multiplicidade de cada entidade nas suas relações.

Após uma minuciosa análise dos requisitos levantados, foi decidido que as entidades necessárias seriam as seguintes:

- **Restaurante:** Local frequentado por um, ou mais, Clientes de forma a experienciarem Comida típica de um dado País.
- **Nome:** Característica que identifica de forma geral um Restaurante ou um Cliente.
- **ID:** Garante a unicidade de um Restaurante.

- **Número de telemóvel:** Forma de contacto preferencial de um Restaurante, também pode ser de um Cliente.
- **Avaliação:** Forma de classificar um Restaurante, poderá, além de conter uma classificação inteira entre 0 e 5, possuir um texto com uma descrição da experiência do Cliente.
- **Banda de preços:** Forma de indicar o preço que um Cliente poderá esperar de um dado Restaurante.
- **Localização:** Posição geográfica de um Cliente ou de um Restaurante.
- **Comida:** Produto de um Restaurante.
- **País:** Local do qual um Restaurante é típico.
- **Cliente:** Utilizador da aplicação.
- **Nome de utilizador:** Nome que identifica e garante a unicidade de um utilizador.
- **Email:** Endereço de correio eletrónico que permite o contacto com o Cliente, é utilizado para a autenticação.
- **Password:** Critério que permite garantir alguma segurança à conta de um utilizador.

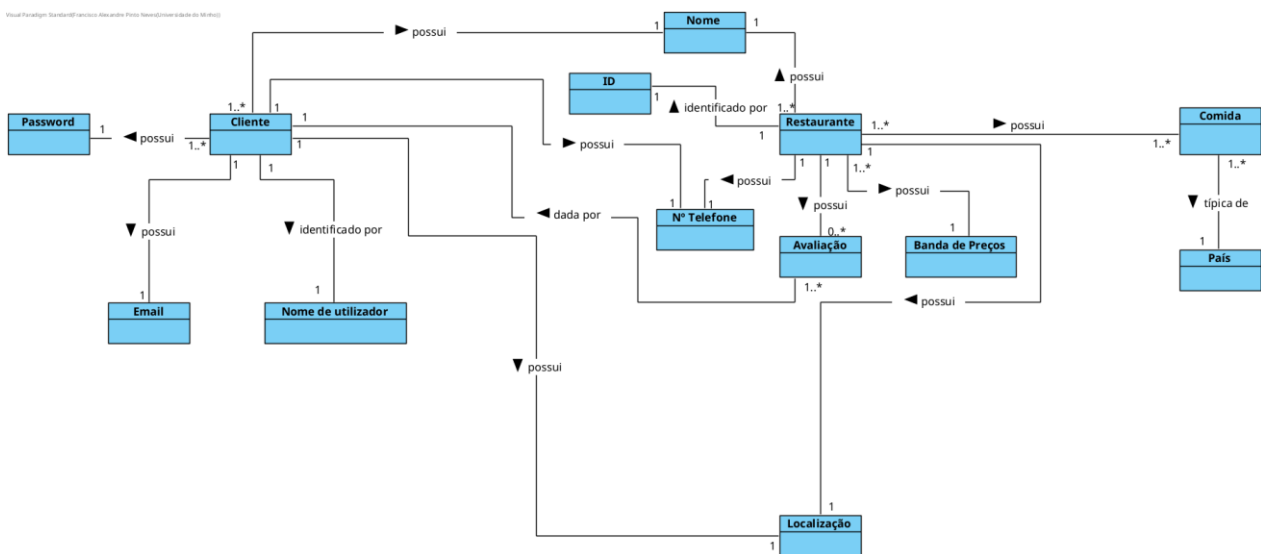


Figura 5: Modelo de Domínio

### 3.3. Aspectos comportamentais

Os Use Cases têm a função de descrever a forma como as entidades externas atingem os seus objetivos utilizando o Sistema, definindo, para isso, as relações entre os inputs dos Atores (entidades externas que interagem com o Sistema) e o comportamento do Sistema.

No caso desta aplicação, existe apenas um Ator: o Cliente, pelo que este será o único que efetuará interações com o sistema.

Apresentam-se na figura abaixo as várias funcionalidades das quais o Cliente pode usufruir, através de modelação por Use Cases. Tendo em conta a complexidade do modelo, foi decidido dividi-lo em dois diagramas de forma a ser mais perceptível a sua visualização:

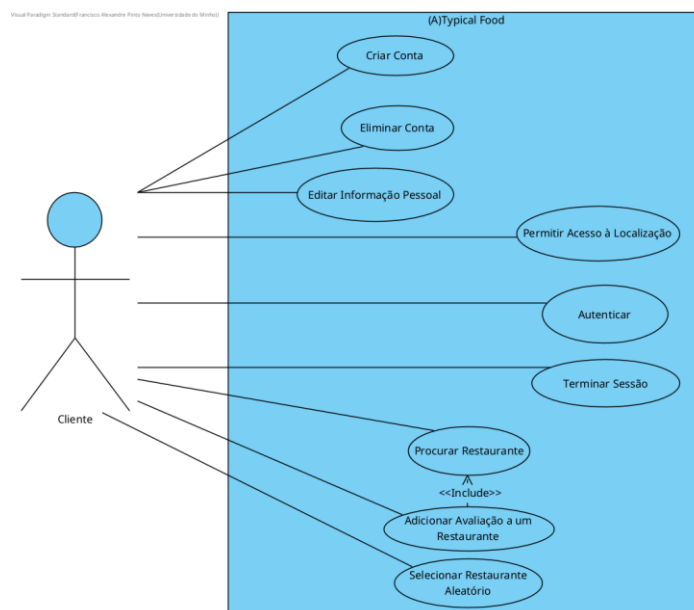


Figura 6: Diagrama Use Case - Geral

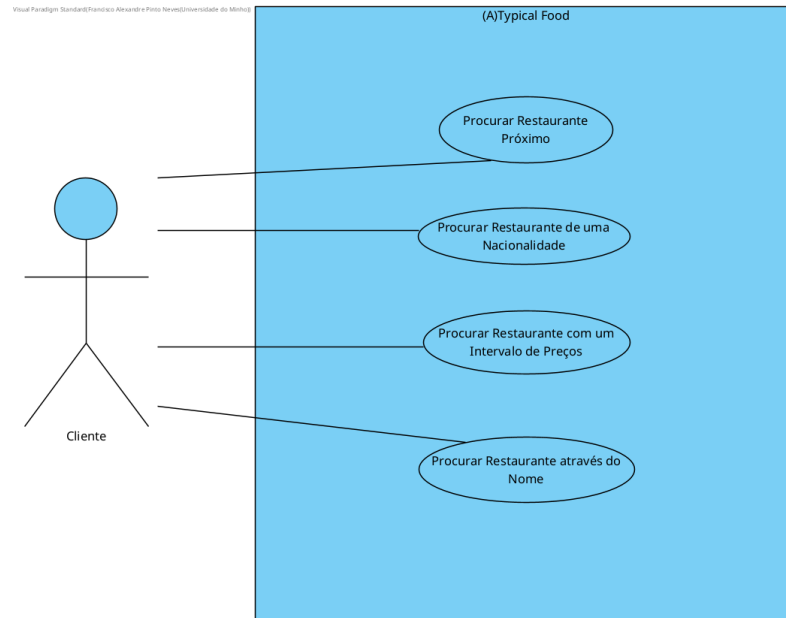


Figura 7: Diagrama Use Case - Procurar Restaurante

Especifica-se abaixo, em detalhe, cada uma das funcionalidades do sistema.



Use Case		
Use Case	Criar Conta	
Ator	Cliente	
Pré-Condição	VERDADEIRO	
Pós-Condição	Cliente tem conta criada.	
	Ator	Sistema
Fluxo Normal	1. Cliente seleciona a opção de criar nova conta.	
		2. Apresenta a tela de criação de conta.
	3. Preenche os diversos campos necessários para a criação de uma conta.	
		4. Confirma que a conta foi criada com sucesso.
		5. Retorna à tela de autenticação/criação de conta.
Fluxo de Exceção (1) [Já existe uma conta com o username inserido] (Passo 3)		3.1. Informa que já existe uma conta com o username inserido.
		3.2. Retorna à tela de autenticação/criação de conta.
Fluxo de Exceção (2) [Já existe uma conta com o email inserido] (Passo 3)		3.1. Informa que já existe uma conta com o email inserido.
		3.2. Retorna à tela de autenticação/criação de conta.
Fluxo de Exceção (3) [Já existe uma conta com o número de telemóvel inserido] (Passo 3)		3.1. Informa que já existe uma conta com o número de telemóvel inserido.
		3.2. Retorna à tela de autenticação/criação de conta.

Tabela 2: Use Case - Criar Conta

Use Case		
Use Case	Autenticar	
Ator	Cliente	
Pré-Condição	Cliente tem conta criada.	
Pós-Condição	Cliente está autenticado.	
	Ator	Sistema
Fluxo Normal	1. Cliente seleciona a opção de se autenticar.	
		2. Apresenta a tela de autenticação.
	3. Cliente insere as credenciais para se autenticar.	
		4. Apresenta a tela principal.
Fluxo de Exceção (1) [As credenciais não existem/não combinam] (Passo 4)		4.1. Informa que as credenciais inseridas não existem/não combinam.

Tabela 3: Use Case - Autenticar

Use Case		
Use Case	Eliminar Conta	
Ator	Cliente	
Pré-Condição	Cliente está autenticado.	
Pós-Condição	Cliente não possui conta.	
	Ator	Sistema
Fluxo Normal	1. Cliente seleciona a opção das definições.	
		2. Apresenta a tela de definições.
	3. Cliente seleciona a opção de eliminar a conta.	
		4. Pede ao cliente para colocar as suas credenciais de autenticação.
	5. Cliente insere as suas credenciais.	
		6. Informa que a conta foi eliminada com sucesso.
		7. Apresenta a tela inicial.
Fluxo de Exceção (1) [As credenciais não existem/não combinam] (Passo 6)		6.1. Informa que as credenciais inseridas não existem/não combinam.

Tabela 4: Use Case - Eliminar Conta

Use Case		
Use Case	Alterar Informação Pessoal	
Ator	Cliente	
Pré-Condição	Cliente está autenticado.	
Pós-Condição	Cliente tem informação pessoal alterada.	
	Ator	Sistema
Fluxo Normal	1. Cliente seleciona a opção das definições.	
		2. Apresenta a tela de definições.
	3. Cliente seleciona a opção de editar a conta.	
		4. Pede ao cliente para colocar as suas credenciais de autenticação.
	5. Cliente insere as suas credenciais.	
		6. Apresenta a tela de alteração de dados.
	7. Cliente altera os dados que deseja alterar.	
		8. Verifica que os dados alterados não entram em conflito com nenhum outro utilizador.
		9. Informa que os dados foram alterados com sucesso.
		10. Apresenta a tela principal.
Fluxo de Exceção (1) [As credenciais não existem/não combinam] (Passo 6)		6.1. Informa que as credenciais inseridas não existem/não combinam.
Fluxo de Exceção (2) [Os dados inseridos entram em conflito com outro utilizador] (Passo 8)		8.1. Informa que não foi possível alterar os dados visto entrarem em conflito com outra conta já existente.
		8.2. Retorna à tela de definições.

Tabela 5: Use Case - Alterar Informação Pessoal

Use Case		
Use Case	Permitir Acesso à Localização	
Ator	Cliente	
Pré-Condição	Cliente está autenticado.	
Pós-Condição	Sistema tem acesso à localização do cliente.	
	Ator	Sistema
Fluxo Normal	1. Cliente seleciona a opção das definições.	
		2. Apresenta a tela de definições.
	3. Marca a opção de permissão de acesso à localização.	
		4. Assinala a opção de permissão de acesso à localização.

Tabela 6: Use Case - Permitir Acesso à Localização

Use Case		
Use Case	Procurar Restaurante Próximo	
Ator	Cliente	
Pré-Condição	Cliente está autenticado e permitiu acesso à sua localização.	
Pós-Condição	Apresentada a lista de restaurantes próximos ordenada pela proximidade.	
	Ator	Sistema
Fluxo Normal	1. Cliente seleciona a opção de procura de restaurantes próximos.	
		2. Apresenta uma lista com os restaurantes ordenados pela proximidade ao Cliente.
Fluxo de Exceção (1) [Não existem restaurantes próximos] (Passo 2)		2.1. Informa que não existem restaurantes próximos disponíveis naquele momento.

Tabela 7: Use Case - Procurar Restaurante Próximo

Use Case		
Use Case	Procurar Restaurante de uma Nacionalidade	
Ator	Cliente	
Pré-Condição	Cliente está autenticado.	
Pós-Condição	Apresentada a lista dos restaurantes da nacionalidade desejada ordenada pela classificação média de cada um.	
	Ator	Sistema
Fluxo Normal	1. Cliente seleciona a opção de procura de restaurantes de uma nacionalidade.	
		2. Apresenta uma lista com as nacionalidades disponíveis.
	3. Cliente seleciona uma nacionalidade.	
		4. Apresenta uma lista com os restaurantes da nacionalidade selecionada ordenados pela classificação média.
Fluxo de Exceção (1) [Não existem nacionalidades] (Passo 2)		2.1. Informa que não existem restaurantes disponíveis naquele momento.

Tabela 8: Use Case - Procurar Restaurante de uma Nacionalidade

Use Case		
Use Case	Procurar Restaurante com um Intervalo de Preços	
Ator	Cliente	
Pré-Condição	Cliente está autenticado.	
Pós-Condição	Apresentada a lista dos restaurantes com os preços no intervalo desejado ordenada pela classificação média.	
	Ator	Sistema
Fluxo Normal	1. Cliente seleciona a opção de procura de restaurantes por preço.	
		2. Apresenta uma lista com os intervalos de preço possíveis.
	3. Cliente seleciona um intervalo de preços.	
		4. Apresenta uma lista com os restaurantes com os preços no intervalo desejado ordenados pela classificação média.
Fluxo de Exceção (1) [Não existem intervalos de preços] (Passo 2)		2.1. Informa que não existem restaurantes disponíveis naquele momento.

Tabela 9: Use Case - Procurar Restaurante com um Intervalo de Preços

Use Case		
Use Case	Procurar um Restaurante através do nome	
Ator	Cliente	
Pré-Condição	Cliente está autenticado.	
Pós-Condição	Apresentada a informação do restaurante desejado.	
	Ator	Sistema
Fluxo Normal	1. Cliente seleciona a aba de pesquisa.	
	2. Cliente digita o nome do restaurante desejado.	
		3. Apresenta uma lista com os restaurantes com o nome inserido ordenada pela classificação média.
	4. Cliente seleciona o restaurante desejado.	
		5. Apresenta a informação do restaurante desejado.
Fluxo de Exceção (1) [Não existem restaurantes com o nome inserido] (Passo 3)		3.1. Informa que não existe qualquer restaurante disponível naquele momento com o nome inserido.

Tabela 10: Use Case - Procurar um Restaurante através do nome

Use Case		
Use Case	Adicionar Avaliação a um Restaurante	
Ator	Cliente	
Pré-Condição	Cliente está autenticado.	
Pós-Condição	Avaliação adicionada à lista de avaliações do Restaurante.	
	Ator	Sistema
Fluxo Normal	1. Cliente procura o Restaurante desejado.	
	2. Cliente seleciona a opção de adicionar uma avaliação.	
		3. Apresenta a tela de criação de uma avaliação.
	4. Cliente seleciona a classificação a fornecer e insere um comentário sobre o restaurante.	
		5. Confirma que foi adicionada a avaliação ao restaurante.
Fluxo Alternativo (1) [Cliente não deseja inserir nenhum comentário] (Passo 4)	4.1. Cliente seleciona a classificação a fornecer.	
		4.2. Regressa a 5.

Tabela 11: Use Case - Adicionar Avaliação a um Restaurante

Use Case		
Use Case	Selecionar Restaurante Aleatório	
Ator	Cliente	
Pré-Condição	Cliente está autenticado.	
Pós-Condição	Apresentada a informação do restaurante selecionado.	
	Ator	Sistema
Fluxo Normal	1. Cliente seleciona a opção de escolha aleatória de um restaurante.	
		2. Escolhe um restaurante de forma aleatória e apresenta a sua informação.
Fluxo de Exceção (1) [Não existem restaurantes] (Passo 2)		2.1. Informa que não existem restaurantes disponíveis no momento.

Tabela 12: Use Case - Selecionar Restaurante Aleatório

Use Case		
Use Case	Terminar Sessão	
Ator	Cliente	
Pré-Condição	Cliente está autenticado.	
Pós-Condição	Cliente não está autenticado.	
	Ator	Sistema
Fluxo Normal	1. Cliente seleciona a opção de terminar sessão.	
		2. Apresenta a tela de autenticação/criação de conta.

Tabela 13: Use Case - Terminar Sessão

## 4. Conceção do sistema de dados

### 4.1. Apresentação geral da estrutura

Com a análise dos requisitos do sistema completa, seguiu-se o desenvolvimento da base de dados. Decidiu-se que esta seria relacional, tendo em conta que este tipo de base de dados permitir armazenar e aceder a dados relacionados uns com os outros e que este facto é extremamente útil para o desenvolvimento do projeto. Conceitos como Modelação Conceptual e Modelação Lógica foram importantes neste processo.

Relativamente à Modelação Conceptual, optamos por criar um Diagrama ER, visto este é um modelo não muito técnico e de compreensão fácil, pelo que, desta forma, torna-se mais fácil a comunicação com o utilizador sem ambiguidades, com o objetivo de ir ao encontro dos requisitos anteriormente expostos. O modelo conceptual tem em vista a representação das entidades e atributos necessários para a realização da base de dados do *(A)Typical Food*.

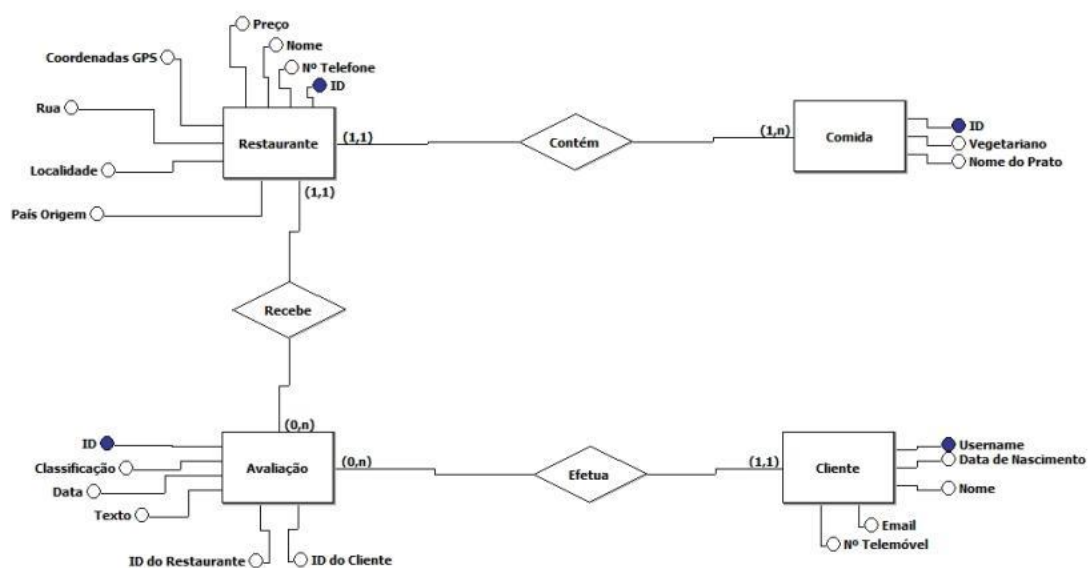


Figura 8: Modelação Conceptual - Diagrama ER



Seguidamente, foi possível partir para o desenho de um Modelo Lógico da base de dados.

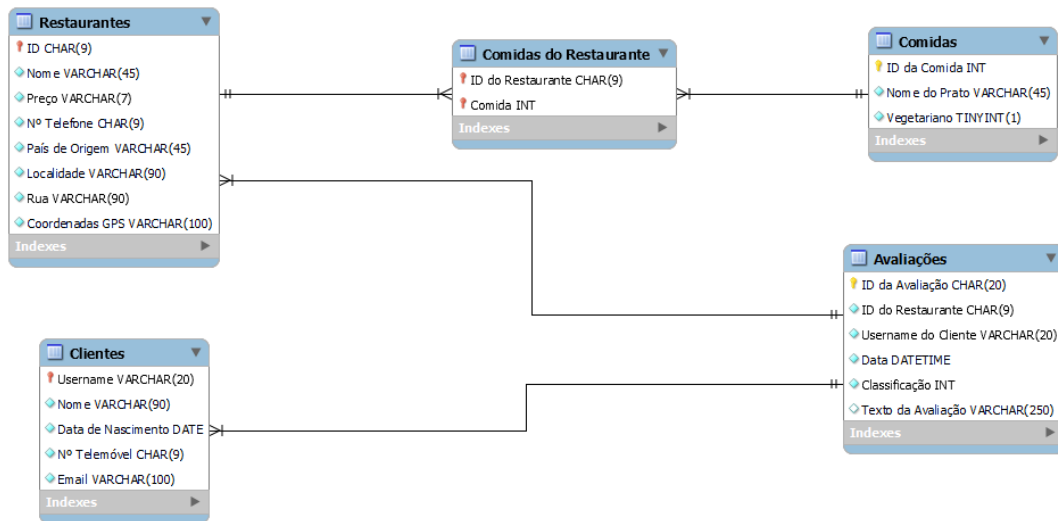


Figura 9: Modelo Lógico

## 4.2. Descrição dos elementos e dos seus relacionamentos

Identificamos as seguintes entidades:

- **Restaurante:** Representa um local frequentado pelo Cliente, de forma a obter Comida típica de uma dada cultura. Tem como atributos o seu país de origem, localidade, rua, coordenadas GPS, preço, nome, número de telefone e ID.
- **Avaliação:** Representa uma forma de classificar um Restaurante, poderá, além de conter uma classificação inteira entre 0 e 5, possuir um texto com uma descrição da experiência do Cliente. Tem como atributos um ID, classificação, data, texto, o ID do restaurante e o username do cliente.
- **Cliente:** Representa os utilizadores que se registam no sistema. Tem como atributos o seu username, data de nascimento, nome, email e número de telemóvel.
- **Comida:** Representa o produto de um Restaurante. Tem como atributos um ID, se é vegetariano e o nome do prato.

Estas relacionam-se da seguinte forma:

- Um Cliente pode efetuar múltiplas Avaliações a um Restaurante.
- Um Restaurante contém múltiplas Comidas.
- Um Restaurante pode receber múltiplas Avaliações.

A seguinte tabela especifica a caracterização das entidades de uma forma mais detalhada, tornando explícitos os seus atributos e as suas características.

Entidade	Atributos	Descrição	Tipo de Atributo	Tipo de dados e tamanho
Restaurante	País Origem	País do qual provêm os seus pratos típicos	Simples	VARCHAR(45)
	Localidade	Localidade onde se localiza	Simples	VARCHAR(90)
	Rua	Rua onde se localiza	Simples	VARCHAR(90)
	Coordenadas GPS	Posição geográfica onde se encontra	Simples	VARCHAR(100)
	Preço	Valor médio que os seus Clientes gastam	Simples	VARCHAR(7)
	Nome	Nome do restaurante	Simples	VARCHAR(45)
	Número de telefone	Forma de contactar consigo	Simples	CHAR(9)
	ID	Identificador único que o distingue	Chave primária	CHAR(9)
Comida	ID	Identificador único que a distingue	Chave primária	INT
	Vegetariano	Booleano que indica se o prato é vegetariano	Simples	TINYINT(1)
	Nome do prato	Nome atribuído ao prato	Simples	VARCHAR(45)

Cliente	Username	Nome de utilizador	Chave Primária	VARCHAR(20)
	Data de Nascimento	Data em que nasceu	Simples	DATE
	Nome	O seu nome	Simples	VARCHAR(90)
	Email	O seu email	Simples	VARCHAR(100)
	Número de telemóvel	O seu contacto por telemóvel	Simples	CHAR(9)
Avaliação	ID	Identificador que a distingue	Chave Primária	CHAR(20)
	Classificação	Avaliação média atribuída	Simples	INT
	Data	Data em que foi realizada	Simples	DATETIME
	Texto	Texto	Simples	VARCHAR(250)
	ID do Restaurante	Identificador único do restaurante	Simples	CHAR(9)
	Username do Cliente	Identificador único do cliente	Simples	CHAR(14)

Tabela 14: Caracterização das Entidades

Para além disso, estimou-se o espaço em disco que a base de dados gastará, de acordo com o MySQL Reference Manual.

Entidade	Atributo	Tipo de Dados	Espaço no Disco
Restaurante	País Origem	VARCHAR(45)	46
	Localidade	VARCHAR(90)	91
	Rua	VARCHAR(90)	91
	Coordenadas GPS	VARCHAR(100)	101
	Preço	VARCHAR(7)	8
	Nome	VARCHAR(45)	46
	Número de telefone	CHAR(9)	9
	ID	CHAR(9)	9
Total			401

Tabela 15: Espaço ocupado por cada entrada de um Restaurante

Entidade	Atributo	Tipo de Dados	Espaço no Disco
Comida	ID	INT	4
	Vegetariano	TINYINT(1)	1
	Nome do prato	VARCHAR(45)	46
Total			51

Tabela 16: Espaço ocupado por cada entrada de Comida

Entidade	Atributo	Tipo de Dados	Espaço no Disco
Cliente	Username	VARCHAR(20)	21
	Data de Nascimento	DATE	3
	Nome	VARCHAR(90)	91
	Email	VARCHAR(100)	101
	Número de telemóvel	CHAR(9)	9
Total			225

Tabela 17: Espaço ocupado por cada entrada de Cliente

Entidade	Atributo	Tipo de Dados	Espaço no Disco
Avaliação	ID	CHAR(20)	20
	Classificação	INT	4
	Data	DATETIME	8
	Texto	VARCHAR(250)	251
	ID do Restaurante	CHAR(9)	9
	Username do Cliente	CHAR(14)	14
Total			306

Tabela 18: Espaço ocupado por cada entrada de Avaliação

Assim, a tabela seguinte demonstra o espaço total gasto por uma entrada em cada tabela.

Entidade	Espaço em disco (em bytes)
Restaurante	401
Comida	51
Cliente	225
Avaliação	306
Total	983

Tabela 19: Espaço em Disco Total para uma Entrada de cada

## 5. Esboço das interfaces do sistema

### 5.1. Estrutura geral

De forma a prever a parte gráfica da aplicação, procedeu-se à criação de *mockups*, isto é, imagens que refletem o *design*, esquema de cores, *layouts*, entre outros aspetos visuais do programa. Torna-se assim essencial esta etapa, porque se antecipa o aspeto final da aplicação, permitindo a construção de uma versão o mais agradável possível. Um utilizador será mais propenso a utilizar uma aplicação da qual goste do aspeto.

Para além disso, os *mockups*, permitem recriar instâncias em que o utilizador poderá interagir com o sistema, logo, a sua construção fornece uma ajuda vital para que estas possam ser planeadas de forma a torná-las as mais práticas e simples possíveis de forma a serem acessíveis a todo o tipo de utilizadores.

Desta forma, foram construídos *mockups* para os seguintes cenários:

- Ecrã de início da aplicação;
- Versão *WebMobile* do ecrã de início.
- Criação de conta;
- Autenticação;
- Página principal da aplicação após autenticação;
- Definições;
- Informação de um restaurante;
- Avaliar um restaurante;
- Procurar um restaurante;
- Mostrar uma lista de restaurantes;

## 5.2. Caracterização

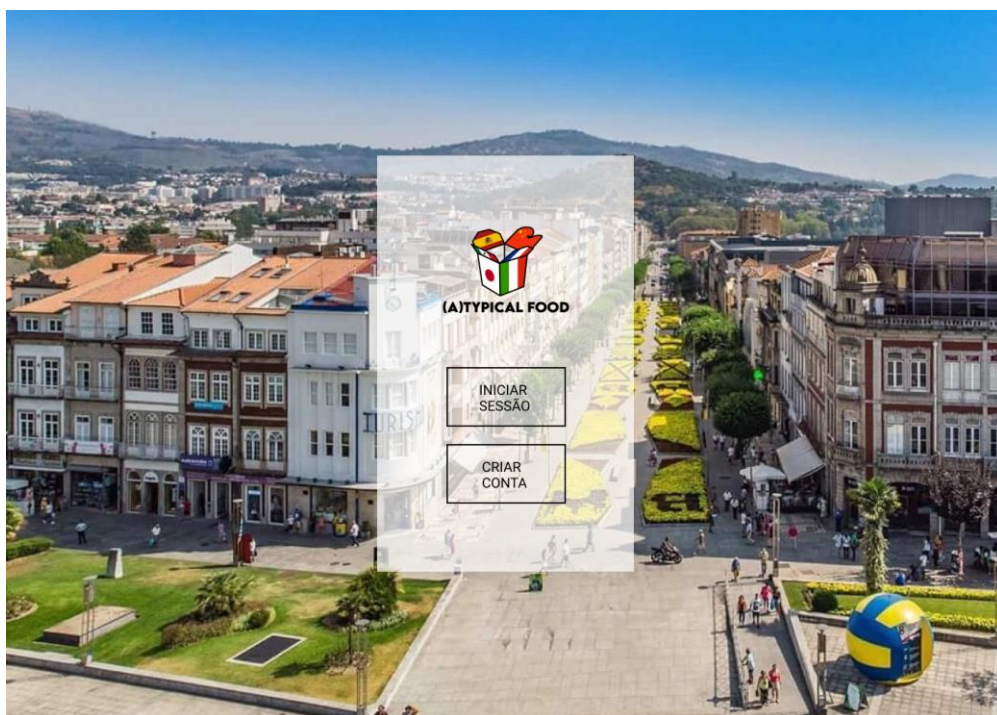


Figura 10: Ecrã Inicial da Aplicação

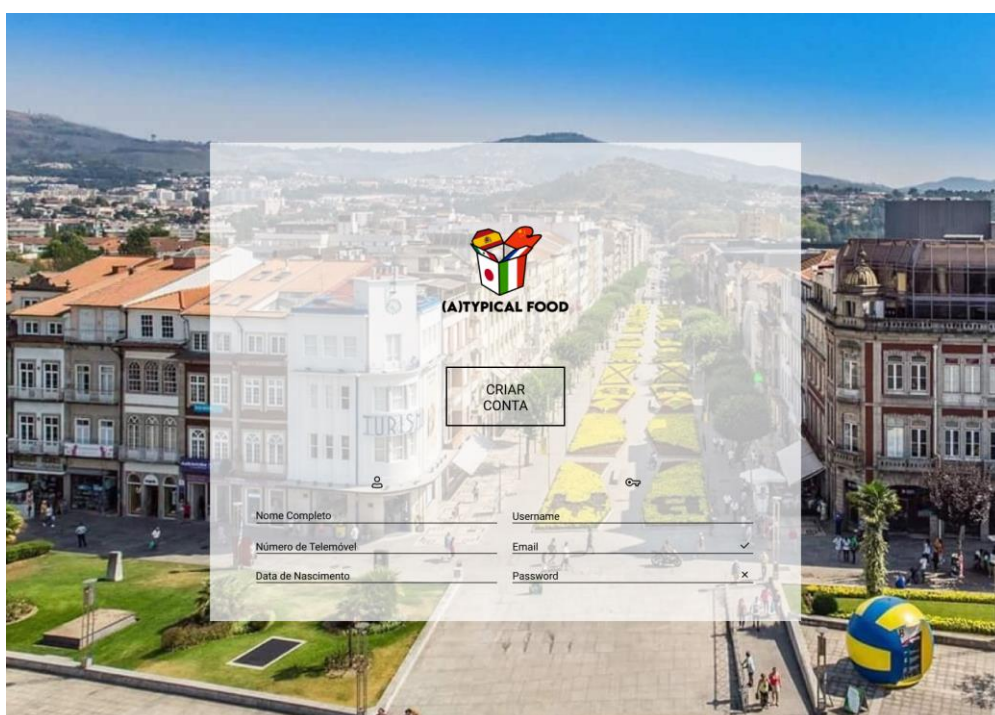


Figura 11: Menu de Criação de Conta





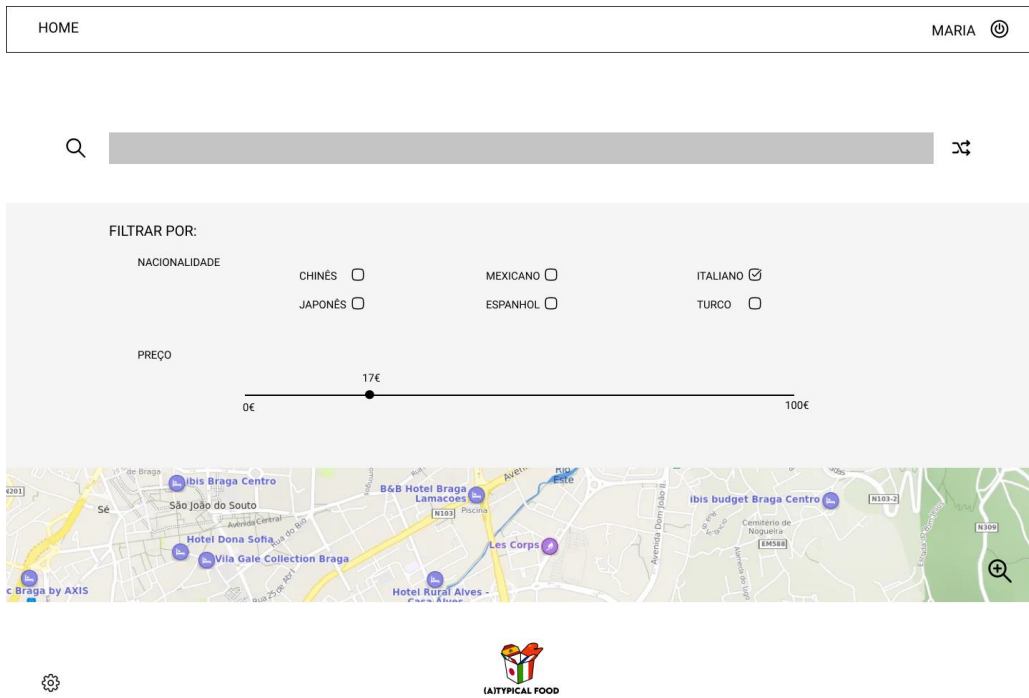


Figura 14: Pesquisa por Restaurante

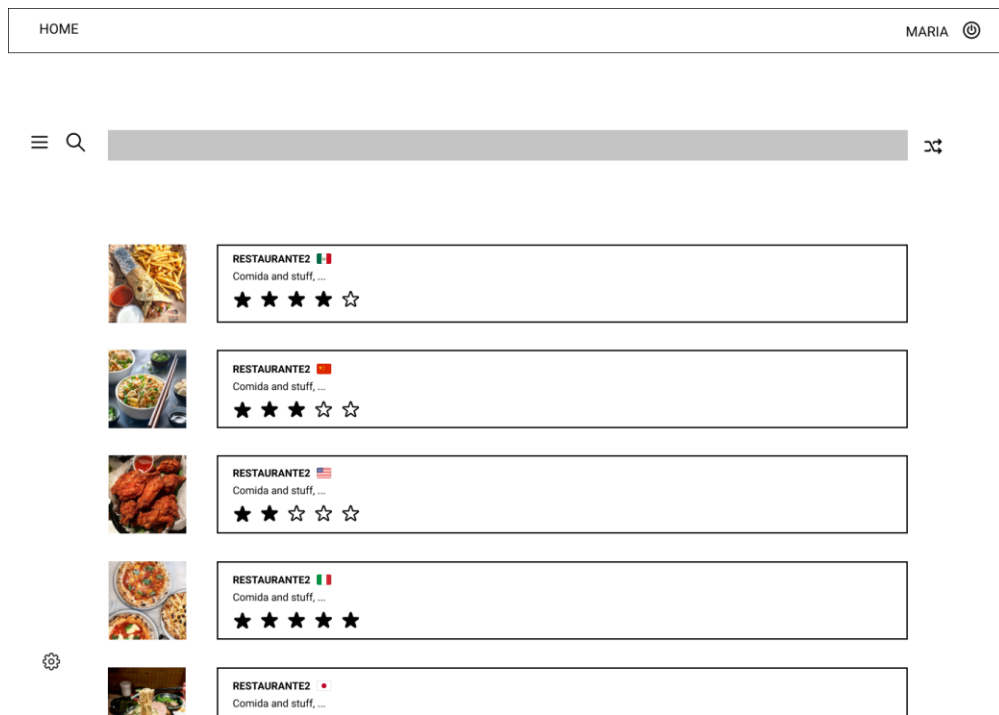


Figura 15: Lista de Restaurantes




INFORMAÇÃO

MARIA

Q

Mikado



## MIKADO

O Restaurante Mikado é uma referência na cozinha de fusão asiática, Chinesa e Japonesa.

Se está à procura de um programa, seja para almoçar ou jantar, então o Mikado é sem dúvida a aposta ideal.

Saboreie e exprima sentimentos com o nosso variado menu, que contém diversas iguarias típicas das regiões asiáticas.

Deverá efetuar a sua marcação, contudo, poderá ir na mesma até ao restaurante sem a chamada efectuada, pois será igualmente bem recebido(a). Se preferir, poderá adquirir o nosso takeaway e degustar num local a seu gosto.

O Mikado possui um Conceito exclusivo dentro de portas, o 1st Floor. Este prima pelo atendimento detalhado, pela explosão de sabores e pelo conforto da sala, com o intuito de lhe proporcionar uma magnífica experiência.

Traga a curiosidade, nós tratamos do resto. Venha comprovar!

€€ - €€€

Ementa

AVALIAÇÕES

MARIO123

Não sei, foi mais ou menos!

★ ★ ★ ★ ☆

MARIO321

Não sei, foi mais ou menos! Não sei, foi mais ou menos! Não sei, foi mais ou menos! Não sei, foi mais ou menos! Não sei, foi mais ou menos! Não sei, foi mais ou menos! Não sei, foi mais ou menos!

★ ★ ☆ ☆ ☆

MARI\_0123

Não sei, foi mais ou menos!

★ ★ ★ ★ ★

Figura 16: Menu de informação do Restaurante



**(A) TYPICAL FOOD**

Figura 18: Definições



(A) TYPICAL FOOD

Figura 18: Definições



Figura 19: Website em Mobile

## 6. Conclusões e Trabalho Futuro

### 6.1. Conclusões

Ao desenvolver o presente projeto, foi possível compreender a importância do devido planeamento de uma aplicação para o sucesso da sua implementação.

Apenas depois de contextualizarmos e de justificarmos a importância e utilidade da aplicação é que foi iniciado o seu desenvolvimento, decidindo-se uma estratégia desde cedo e enumerando-se primeiro os requisitos que esta deveria ser capaz de cumprir.

Com a parte referida anteriormente bem definida, foi intuitivo partir para a definição da estrutura e dos aspetos comportamentais da aplicação, criando-se um modelo de domínio e explicitando-se os casos de utilização.

Relativamente à base de dados, notou-se que o desenvolvimento de um modelo conceptual permitiu descrever claramente as relações entre as entidades antes de nos debruçarmos para o modelo lógico, realizado com *MySQL*.

Por fim, o desenvolvimento de interfaces foi importante para criarmos uma ideia da estética, aspeto e funcionalidade da interação com o utilizador, uma vez que muitos utilizadores priorizam a utilização de aplicações com bom aspeto e simples de utilizar.

## 6.2. Trabalho futuro

Além do trabalho inicial para o desenvolvimento do *(A)Typical Food*, pretende-se que este, ao longo do tempo e com a angariação de novos utilizadores, possa evoluir, apresentando novas funcionalidades aos utilizadores e, até mesmo, a expansão do seu conceito inicial.

Tendo isto em conta, pretendemos que, no futuro, a aplicação apresente a possibilidade de pesquisa de restaurantes multiculturais, não apenas em Braga, mas sim numa região que se venha a expandir ao longo do tempo.

Além disto, desejamos também que exista a opção de escolha de diversas línguas na interface do programa, para que, deste modo, se possa atrair possíveis clientes que não estejam aptos para falar português.

Por fim, também é nossa intenção, que possam ser implementados algoritmos com recurso a técnicas de *Machine Learning* para que possam ser sugeridos restaurantes aos clientes com base nos seus gostos.

## Referências

Sommerville, I., 2011. Software engineering. 10th ed. Boston: Addison-Wesley.

SEFSTAT. [online] Disponível em: <<https://sefstat.sef.pt/forms/distritos.aspx>>[Consultado a 15 de novembro de 2021].

Observador. [online] Disponível em: <<https://observador.pt/2021/06/23/estrangeiros-em-portugal-alcancam-em-2020-numero-mais-elevado-de-sempre/>>[Consultado a 15 de novembro de 2021]

GeeksForGeeks. [online] Disponível em: <<https://www.geeksforgeeks.org/non-functional-requirements-in-software-engineering/>> [Consultado a 18 de novembro de 2021]

Lucidchart. [online] Disponível em: <<https://www.lucidchart.com/pages/er-diagrams>>[Consultado a 18 de novembro de 2021]

Computer Science, University of Toronto. [online] Disponível em: <<http://www.cs.toronto.edu/~jm/340S/PDF2/CM2.pdf>>[Consultado a 18 de novembro de 2021]

MySQL. [online] Disponível em: <<https://dev.mysql.com/doc/refman/8.0/en/>> [Consultado a 19 de novembro de 2021]

Uxpin. [online] Disponível em <<https://www.uxpin.com/studio/blog/what-is-a-mockup-the-final-layer-of-ui-design/>>[Consultado a 20 de novembro de 2021]

## **Lista de Siglas e Acrónimos**

**LI4** Laboratórios de Informática IV

**SEFSTAT** Portal de Estatística do Serviço de Estrangeiros e Fronteiras

**ER** Entidade Relacionamento