



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

Friendcesinha – Francesinha em Braga

Grupo 5

Ana Murta, Ana Henriques, Leonardo Freitas,

Rui Coelho, Tiago Carneiro

novembro, 2021

Data de Recepção	
Responsável	
Avaliação	
Observações	

Friendcesinha – Francesinha em Braga

Grupo 5

Ana Murta, Ana Henriques, Leonardo Freitas,

Rui Coelho, Tiago Carneiro

novembro, 2021

Resumo

O aumento significativo de confecção de Francesinhas instigou à criação de uma associação, a Confraria da Francesinha de Braga, cujo objetivo centra-se em querer apresentar as diversas confeções deste prato na cidade de Braga. De acordo com a confraria, existe uma lacuna em termos de informação relevante e estruturada acerca da oferta bracaraense, pelo que requereu o desenvolvimento de uma plataforma que respondesse eficazmente a esta falta de informação. Como tal, nasce o projeto ***Friendcesinha – Francesinha em Braga***, um software de apoio que se equipara a um amigo, que objetiva ajudar a localizar os restaurantes bracaraenses que servem uns dos pratos mais emblemáticos da região norte, as Francesinhas. O desenvolvimento desta plataforma passou por uma fase de conceção teórica, que procurou, em conjunto com o cliente, definir de forma clara os vários requisitos aos quais a plataforma deveria responder. Este processo de especificação é de elevada importância, uma vez que marca a *baseline* para o trabalho a ser construído. Uma vez bem definidos, e com recurso à *Unified Modelling Language*, o trabalho centrou-se em modelar concetualmente a plataforma a desenvolver. Através de um Modelo de Domínio e de um Diagrama de *Use Cases*, foi fornecida uma aproximação estrutural, embora abstrata, da aplicação a desenvolver. Esta modelação permite, essencialmente, mostrar quais são os principais componentes da aplicação a desenvolver, e que interação existe, e pode existir, entre estes componentes. Adicionalmente, com a modelação da aplicação, surge também a necessidade de descrever o modo como os dados, sobre os quais a plataforma trabalha, serão armazenados. Assim sendo, para o sistema de dados, foi prevista a utilização de uma Base de Dados Relacional. Visando a construção correta deste sistema, foi fornecido um Modelo Concetual e um Modelo Lógico, que pretendem descrever não só os dados considerados relevantes, mas também a forma como estes devem estar estruturados e organizados.

Área de Aplicação: Engenharia de *Software*, Desenho e Arquitetura de Sistemas de Bases de Dados, Sistemas de Recomendação, Guia para Locais de Interesse

Palavras-Chave: Planeamento de *software*, Bases de Dados Relacionais, *Unified Modelling Language*, Plataforma, Locais de Interesse

Índice

RESUMO	I
ÍNDICE	II
ÍNDICE DE FIGURAS	IV
ÍNDICE DE TABELAS	V
1. INTRODUÇÃO	1
1.1. CONTEXTUALIZAÇÃO	1
1.2. MOTIVAÇÃO E OBJETIVOS.....	2
1.3. IDENTIDADE DO PROJETO	3
1.4. RECURSOS NECESSÁRIOS.....	3
1.5. MAQUETA DO SISTEMA	4
1.6. MEDIDAS DE SUCESSO.....	5
1.7. PLANO DE DESENVOLVIMENTO.....	6
2. LEVANTAMENTO E ANÁLISE DE REQUISITOS	7
2.1. ESTRATÉGIA, MÉTODO E VALIDAÇÃO	7
2.2. REQUISITOS.....	7
2.2.1 <i>Requisitos funcionais</i>	7
2.2.2 <i>Requisitos não funcionais</i>	8
3. ESPECIFICAÇÃO E MODELAÇÃO DO SOFTWARE.....	10
3.1. APRESENTAÇÃO GERAL DA ESPECIFICAÇÃO.....	10
3.2. MODELO DE DOMÍNIO	10
3.3. DIAGRAMA DE USE CASES.....	11
4. CONCEÇÃO DO SISTEMA DE DADOS.....	13
4.1. MODELO CONCETUAL.....	13
4.2. MODELO LÓGICO	15
5. ESBOÇO DE INTERFACES DO SISTEMA.....	16
5.1. CARACTERIZAÇÃO DAS INTERFACES	16
6. CONCLUSÕES E TRABALHO FUTURO	19
REFERÊNCIAS	20
LISTA DE SIGLAS E ACRÓNIMOS	21
ANEXOS	22

I.	USE CASES	23
II.	MOCKUPS	30

Índice de Figuras

FIGURA 1 – LOGOTIPO DA APLICAÇÃO <i>FRIENDCESINHA</i>	3
FIGURA 2 – MAQUETA DO SISTEMA A DESENVOLVER.	5
FIGURA 3 – DIAGRAMA DE GANTT DO PROJETO.....	6
FIGURA 4 – MODELO DE DOMÍNIO DO SISTEMA.	11
FIGURA 5 – DIAGRAMA DE <i>USE CASES</i>	12
FIGURA 6 – MODELO CONCETUAL DA BD.	14
FIGURA 7 – MODELO LÓGICO DA BD.....	15
FIGURA 8 – <i>MOCKUP</i> : PÁGINA INICIAL DA PLATAFORMA.....	16
FIGURA 9 – <i>MOCKUP</i> : PÁGINA INICIAL DO CLIENTE.	17
FIGURA 10 – <i>MOCKUP</i> : MAPA COM RESTAURANTES.....	17
FIGURA 11 – <i>MOCKUP</i> : PERFIL DO RESTAURANTE.	18
FIGURA 12 – <i>MOCKUP</i> : EDIÇÃO DO PERFIL DO CLIENTE.	30
FIGURA 13 – <i>MOCKUP</i> : AVALIAÇÃO DO RESTAURANTE PELO CLIENTE.	30
FIGURA 14 – <i>MOCKUP</i> : PEDIDO DE RESERVA.	31
FIGURA 15 – <i>MOCKUP</i> : PÁGINA COM CÓDIGOS DE DESCONTO.	31

Índice de Tabelas

TABELA 1 – DICIONÁRIO DE RELACIONAMENTOS.....	14
TABELA 2 – USE CASE: REGISTRAR NA PLATAFORMA.....	23
TABELA 3 – USE CASE: CONSULTAR RESTAURANTE.	23
TABELA 4 – USE CASE: TERMINAR SESSÃO.....	24
TABELA 5 – USE CASE: EFETUAR AUTENTICAÇÃO.	24
TABELA 6 – USE CASE: CONSULTAR CÓDIGOS PROMOCIONAIS.....	25
TABELA 7 – USE CASE: EFETUAR UMA RESERVA.	25
TABELA 8 – USE CASE: REALIZAR AVALIAÇÃO.....	26
TABELA 9 – USE CASE: ALTERAR DEFINIÇÕES DE CONTA.	26
TABELA 10 – USE CASE: VALIDAR REGISTO.....	26
TABELA 11 – USE CASE: ALTERAR INFORMAÇÃO DO RESTAURANTE.....	27
TABELA 12 – USE CASE: CONSULTAR AVALIAÇÕES.....	27
TABELA 13 – USE CASE: REMOVER REGISTO DE RESTAURANTE.	27
TABELA 14 – USE CASE: GERAR CÓDIGO PROMOCIONAL.	28
TABELA 15 – USE CASE: EFETUAR PEDIDO.....	28
TABELA 16 – USE CASE: CANCELAR RESERVA.....	28
TABELA 17 – USE CASE: REGISTRAR RESTAURANTE.	29

1. Introdução

1.1. Contextualização

O nome “Francesinha” pouco significa para aqueles que nunca visitaram Portugal. Mesmo assim, para aqueles que cá habitam, existem diversos que desconhecem o característico sabor deste prato, característico da cidade do Porto, com elevada representatividade no norte do país (Ribeiro, 2014).

De acordo com a lenda, foi por volta de 1950 que Daniel David Silva concebeu a primeira Francesinha na cervejaria “A Regaleira”, no Porto (“Francesinha”, 2021). Conta-se que, tendo sido um emigrante em França, inspirou-se numa das sanduíches francesas mais típicas, a *Croque-Monsieur*, e decidiu adaptá-la aos sabores fortes e quentes dos portugueses (Bessa, 2020).

Inicialmente, a composição da Francesinha revelava-se mais simples, comparativamente com as diversas opções disponíveis na atualidade: continha apenas carnes e queijo, sendo apenas diferenciável das restantes sandes pelo seu molho característico. Nos tempos mais atuais, encontram-se diversas ramificações, existindo variedades com ovo, batata ou até mesmo com uma vertente vegetariana (Bessa, 2020).

A essência deste prato reside no molho: as receitas variam de casa para casa, havendo restaurantes que criam molhos de excelência, cuja combinações de ingredientes parece ser um segredo bem guardado (Ribeiro, 2014). A título de exemplo, a diversidade e quantidade de bebidas alcoólicas utilizadas afetam de forma determinante não só o sabor geral do prato, mas também a sua acidez ou doçura.

A Francesinha tornou-se rapidamente uma aposta de sucesso na gastronomia (Várzea, 2020), por um lado devido ao seu sabor intenso, por outro pelo facto de a sua confeção ser barata, impulsionando a propagação da receita por diversos restaurantes espalhados pelo país. Atualmente, esta sanduíche característica da gastronomia nacional continua a ter um enorme sucesso, tendo sido eleita uma das 10 melhores sanduíches do mundo pelo *site AOL Travel*, bem como pelo *site The Culture Trip* e pela prestigiada revista espanhola *Condé Nast Traveller* (“Francesinha”, 2021).

Conferindo foco à cidade de Braga, não é passível de dizer que se caracteriza pelas suas Francesinhas. No entanto, atendendo ao aumento significativo de confeção de Francesinhas em Braga, os restaurantes bracarenses têm procurado oferecer alternativas à tradicional Francesinha do Porto (Almeida, n.d.). Estes acontecimentos instigaram a criação de uma associação, a Confraria da Francesinha

de Braga (CFB), cujo objetivo central resume-se a querer dar a conhecer as confeções deste prato na cidade de Braga, procurando mobilizar turistas e entusiastas da gastronomia em busca dos variados sabores de Francesinha.

1.2. Motivação e Objetivos

A Francesinha, tal como foi supramencionado, constitui uma parte importante da cultura gastronómica do Norte, particularmente da cidade do Porto. Como tal, é natural que o número de aficionados por Francesinhas tenha aumentado com o tempo, sendo este aumento acompanhado pelo desenvolvimento de plataformas e/ou de documentação acerca dos restaurantes que incorporam este prato nos seus menus.

No que diz respeito à cidade de Braga, e tal como foi referido anteriormente, são diversos os restaurantes que incluem este prato na sua ementa. No entanto, o problema que surge quando diversas opções existem reside na escolha do lugar mais adequado para experimentar uma Francesinha. A adequação aqui é a chave da escolha, pois diferentes indivíduos concebem o que é “bom” de modos distintos, podendo optar por restaurantes em função da sua localização, do preço, das opções e variações disponibilizadas em termos de Francesinhas ou, ainda, pelo *rating* atribuído por outros indivíduos.

Assim sendo, e pela lacuna que se faz sentir no mercado neste âmbito, a CFB requereu a criação de uma plataforma que apresente a Francesinha de Braga. O desenvolvimento desta plataforma procura, primeiramente, colocar Braga no mapa como sendo um ponto de referência gastronómico de Francesinhas, impulsionando o fluxo de clientes nestes estabelecimentos de restauração. Adicionalmente, a plataforma deve ser capaz de dar a conhecer não só a Francesinha, em termos de constituição e classificação atribuída pela CFB, mas também o restaurante – horário de funcionamento, preço e localização. Com o desenvolvimento desta aplicação, pretende-se ainda que os utilizadores disfrutem de uma experiência gastronómica de qualidade, impulsionando, deste modo, a adesão de novos utilizadores à aplicação e, consequentemente, o aumento significativo de clientes nestes estabelecimentos. Por fim, a aplicação deve ser capaz de dar fornecer ao utilizador uma lista dos restaurantes de interesse geograficamente mais próximos, disponibilizando rotas para os mesmos – considerando-se que estes restaurantes são aqueles que respeitam um determinado conjunto de características definidas pelo utilizador.

1.3. Identidade do projeto

Tal como o nome sugere, o *Friendcesinha – Francesinha em Braga* é um software de apoio que se equipara a um amigo, cujo objetivo centra-se em ajudar a localizar os restaurantes bracarenses que servem uns dos pratos mais emblemáticos da região norte, as Francesinhas.

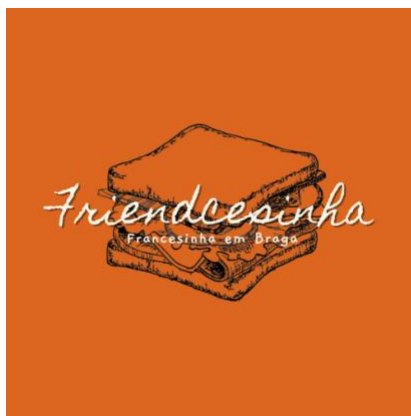


Figura 1 – Logotipo da aplicação *Friendcesinha*.

1.4. Recursos necessários

Atendendo à informação previamente prestada, e considerando que o presente projeto se cinge ao domínio da restauração, são precisos recursos que se apliquem especificamente a este ramo, em conjunto com recursos já intrínsecos à criação de *software*. Estima-se que o desenvolvimento da plataforma tenha uma duração aproximada de 4 meses, com um custo total de 30 000€.

De modo a desenvolver de forma metódica e correta a aplicação proposta no presente projeto, desde a sua especificação até à sua construção, é imperativo que a equipa seja formada por três programadores, um analista e um gestor de projetos – onde todos os elementos integrantes devem deter experiência em programação, para que seja promovido um envolvimento ativo por todos nas várias etapas do projeto. Adicionalmente, são necessárias instalações para reuniões, de modo a permitir que sejam realizadas sessões com o cliente para que este possa acompanhar o estado de desenvolvimento da aplicação.

Conferindo foco ao domínio sob o qual a aplicação será desenvolvida, é importante o levantamento dos estabelecimentos bracarenses que pretendem integrar a aplicação, i.e., é necessário efetuar uma listagem dos mesmos e criar protocolos com os seus proprietários, para que estes registem e efetuem a gestão da informação dos seus restaurantes na aplicação.

Do ponto de vista do desenvolvimento de *software*, será indispensável a utilização de ferramentas de geolocalização, que possibilitem a obtenção de direções para os restaurantes indexados na aplicação.

Finalmente, atendendo ao facto de que o mundo se encontra cada vez mais digital, é indispensável acolher um profissional de *Marketing* à equipa de trabalho, com a finalidade de investir na divulgação da aplicação, quer seja para potenciar novos utilizadores, quer para cativar os proprietários de estabelecimentos de restauração a integrar o seu restaurante na aplicação.

1.5. Maqueta do sistema

O sistema a desenvolver pode ser decomposto em duas componentes distintas: *Front-End* e *Back-End*. O *Front-End* deverá ser apresentado através de uma versão *web*:

- O *website* deve permitir que o utilizador usufrua de um conjunto de funcionalidades, como por exemplo a possibilidade de seleccionar um restaurante, obter direções para o estabelecimento e efetuar reservas;
- No *website* questões relacionadas com a trajetória e localização dos estabelecimentos deve contar com um serviço de mapeamento, como, a título de exemplo, a *Application Programming Interface (API)* do *Bing Maps*;
- A lógica de negócio deve ser responsável por processar a informação enviada para o *website* e por gerir a lógica de utilizadores, permitindo que estes se registem e se autenticem.

O *Back-End* corresponde ao servidor que sustenta o *website*, cuja principal função não só incide no armazenamento e a manutenção de todos os dados dos utilizadores e dos serviços, mas também na gestão das funcionalidades internas do sistema.

A Figura 2 apresenta uma representação esquemática da maqueta do sistema.

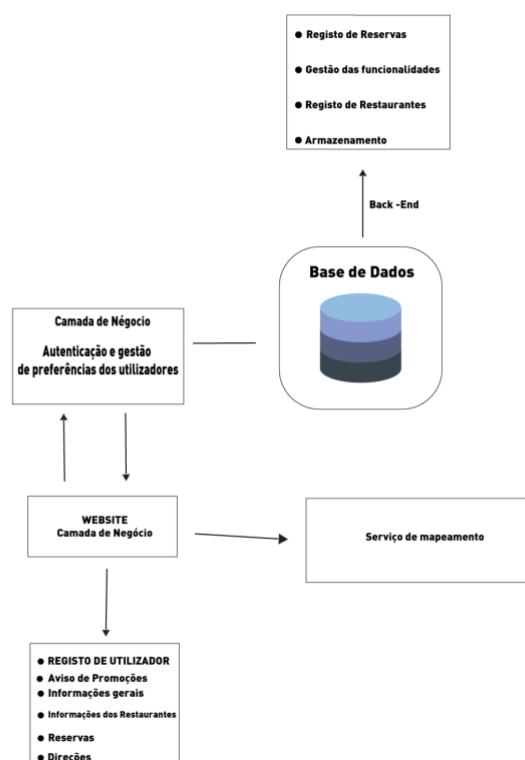


Figura 2 – Maqueta do sistema a desenvolver.

1.6. Medidas de sucesso

Visando uma avaliação do sucesso da aplicação a desenvolver, em conjunto com a CFB foram explicitadas um conjunto de métricas que procuram determinar o êxito do projeto a desenvolver. Tal como foi supramencionado, existe uma lacuna no que diz respeito a plataformas que providenciem informação acerca dos diversos restaurantes que servem Francesinhas na cidade de Braga. Deste modo, uma primeira medida de sucesso da plataforma a desenvolver consiste numa elevada aderência à mesma, quer por parte dos restaurantes, quer relativamente ao número de utilizadores.

Especificamente, considera-se que a aplicação terá bons desempenhos se atingir o limiar mínimo de adesão de 70% dos restaurantes bracarenses, e cerca de 30 mil utilizadores registados no primeiro mês após o lançamento da plataforma. Associado ainda com a vasta aderência à aplicação, estima-se ainda que a plataforma seja capaz de conquistar o destaque na *front page* da *Google* ao pesquisar por soluções de restaurantes na área de Braga.

Outro parâmetro para determinar o sucesso do projeto consiste no rendimento da plataforma. Ambiciona-se que, ao fim de um ano, seja alcançada uma rentabilidade de cerca de 25% do investimento inicial, despendido na sua implementação. Considerando os restaurantes aderentes, estima-se um crescimento de 20% nas suas receitas totais anuais.

Por fim, no que diz respeito à plataforma, espera-se que os utilizadores usufruam de uma boa experiência de utilização, com uma interface simples e intuitiva, e que tal os motive a votar na plataforma para o prémio “Inovação 2021”, atribuído pelo Posto de Turismo da cidade de Braga.

1.7. Plano de desenvolvimento

O presente projeto faz-se acompanhar de um plano de desenvolvimento que visa dar a conhecer o planeamento efetuado para o desenvolvimento da aplicação. A concreta definição de um plano estruturado e organizado é imprescindível para a realização de um bom projeto, pois possibilita que o trabalho ocorra de forma faseada e permite efetuar avaliações periódicas do progresso alcançado.

A Figura 3 apresenta o Diagrama de Gantt desenhado para o desenvolvimento da plataforma *Friendcesinha*. Este diagrama contempla três fases distintas de elaboração do projeto: fundamentação, especificação e construção. Para cada fase, encontram-se discriminadas as tarefas a elaborar e a duração temporal correspondente.

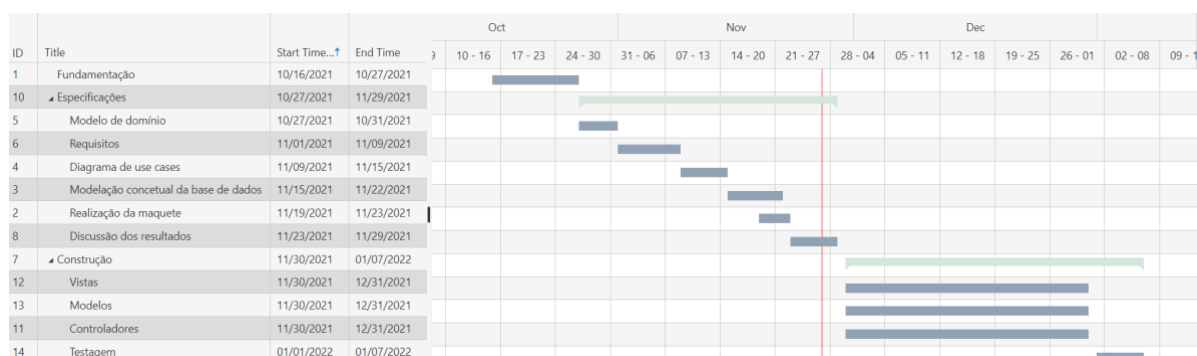


Figura 3 – Diagrama de Gantt do projeto.

2. Levantamento e Análise de Requisitos

2.1. Estratégia, método e validação

O levantamento dos requisitos da plataforma a desenvolver constitui uma parte crucial no processo de desenvolvimento de um sistema de *software*. Como tal, é imperativo que seja adotada uma estratégia de levantamento de requisitos estruturada, metódica e rigorosa, para que os requisitos levantados reflitam as características desejadas para a plataforma.

O acompanhamento por parte do cliente no decurso do desenvolvimento da plataforma é algo indispensável. Particularmente, o cliente detém um papel crucial na fase de especificação dos requisitos da aplicação, uma vez que estes definem as características, funcionais e não funcionais, às quais o sistema a desenvolver deve responder. Para tal efeito, foram elaboradas reuniões periódicas com membros da CFB, a fim de definir um conjunto de requisitos para a plataforma. Estas reuniões permitiram uma melhor compreensão das características da plataforma a desenvolver através do diálogo ativo entre a equipa e de desenvolvimento e o cliente.

Numa primeira reunião, os membros da CFB deram a conhecer a sua visão da aplicação a desenvolver, tendo esta visão servido de base para a plataforma a construir. Partindo desta construção inicial, a equipa de trabalho levantou alguns requisitos que julgavam corresponder às características pedidas pelo cliente. As sessões seguintes possibilitaram a reestruturação dos requisitos, em função das discussões geradas entre equipa e cliente. Assim sendo, os requisitos abaixo apresentados resultam deste debate constante existente nestas reuniões e, como tal, refletem o acordo final estabelecido entre a equipa e o cliente.

2.2. Requisitos

2.2.1 Requisitos funcionais

- i) Para criar conta na aplicação, o utilizador tem de inserir um *username* e um *e-mail* válidos, i.e., que ainda não se encontrem registados, e uma *password*;
- ii) Um utilizador pode ser um cliente, um proprietário ou o administrador da plataforma;
- iii) Para se registar como um proprietário, um cliente deve fornecer um comprovativo de propriedade do negócio, assim como as informações referentes ao mesmo;
- iv) O administrador da plataforma adiciona ou remove os restaurantes, em função dos pedidos recebidos;

- v) O administrador é responsável por validar os comprovativos de propriedade submetidos pelos proprietários;
- vi) A autenticação na aplicação deve ser feita com recurso ao *username* e a respetiva palavra-passe;
- vii) Um utilizador pode editar *username* e palavra-passe do seu perfil, não estando, contudo, autorizado a alterar o seu *e-mail*;
- viii) A aplicação permite que o cliente defina um raio correspondente à distância máxima de procura de restaurantes;
- ix) A aplicação permite que o cliente defina um número mínimo de estrelas para os restaurantes a procurar;
- x) A aplicação deve permitir que as preferências do cliente possam ser alteradas a qualquer momento;
- xi) Os resultados apresentados, sob a forma de uma lista, devem estar ordenados pelo seu número de estrelas, respeitando a configuração de preferências previamente definida.
- xii) Após a seleção do restaurante, o cliente pode consultar a informação relevante acerca do mesmo, como o menu das francesinhas e ingredientes que as constituem;
- xiii) O cliente pode efetuar reservas num restaurante;
- xiv) As reservas efetuadas devem ser descritas pela data e hora da reserva, juntamente com o número de pessoas;
- xv) O cliente pode, no momento em que efetua a reserva, fazer o seu pedido, para que este seja antecipadamente preparado;
- xvi) A aplicação deverá ser capaz de fornecer direções, com recurso ao *Global Positioning System* (GPS), até ao restaurante selecionado pelo cliente;
- xvii) Após o registo na plataforma, o cliente tem direito a um código *Quick Response* (QR) de boas-vindas, com desconto 10% que pode usufruir num dos restaurantes aderentes;
- xviii) Os proprietários podem gerar códigos QR promocionais para o seu estabelecimento;
- xix) Os clientes podem avaliar os restaurantes através de um sistema de estrelas (entre 0 e 5) e efetuando comentários;
- xx) O cliente pode consultar as avaliações feitas por outros clientes;
- xxi) O proprietário pode alterar as informações referente ao seu restaurante;

2.2.2 Requisitos não funcionais

- xxii) A plataforma deve estar funcional todos os dias da semana, 24 horas por dia;
- xxiii) A plataforma não deve necessitar de condições especiais para funcionar para além de conectividade à internet e um dispositivo com GPS;
- xxiv) A aplicação deverá ser acessível, com *layout* simples e intuitivo;

- xxv) A aplicação deverá ter uma boa performance em termos processamento, i.e., processamento rápido e curto tempo de resposta.

3. Especificação e Modelação do Software

3.1. Apresentação geral da especificação

O sistema a desenvolver deve ser capaz de fornecer informação de interesse acerca dos restaurantes que servem Francesinhas na cidade de Braga, permitindo, entre outras funcionalidades, obter direções para os mesmos e efetuar reservas. De modo a assegurar que o *software* a desenvolver cumpre as funcionalidades desejadas pelo cliente, o processo de desenvolvimento incorpora a modelação do sistema, para que possa ser determinada uma representação das interações existentes entre as diversas entidades que o incorporam. Esta representação, embora seja abstrata, permite de um modo simples compreender a composição do sistema e as relações que nele ocorrem.

A concretização desta modelação foi elaborada com recurso à *Unified Modeling Language* (UML), através de um Modelo de Domínio e de um Diagrama de *Use Cases*. O Modelo de Domínio corresponde a uma apresentação concetual do sistema que pretende, de uma forma simples, dar a conhecer as interações ocorrentes no mesmo. Por sua vez, o Diagrama de *Use Cases*, pretende descrever o modo como os diversos atores interagem com o sistema – onde cada interação é denominada de *Use Case*.

3.2. Modelo de Domínio

O Modelo de Domínio construído toma como base os requisitos funcionais previamente levantados. Partindo destas especificações, o processo de desenvolvimento do modelo iniciou-se pelo registo das entidades e das suas relações. A criação do modelo ocorreu sob a forma de um processo incremental e iterativo, tendo a sua versão final sido validada pelo cliente. A Figura 4 apresenta o Modelo de Domínio gerado para a plataforma.

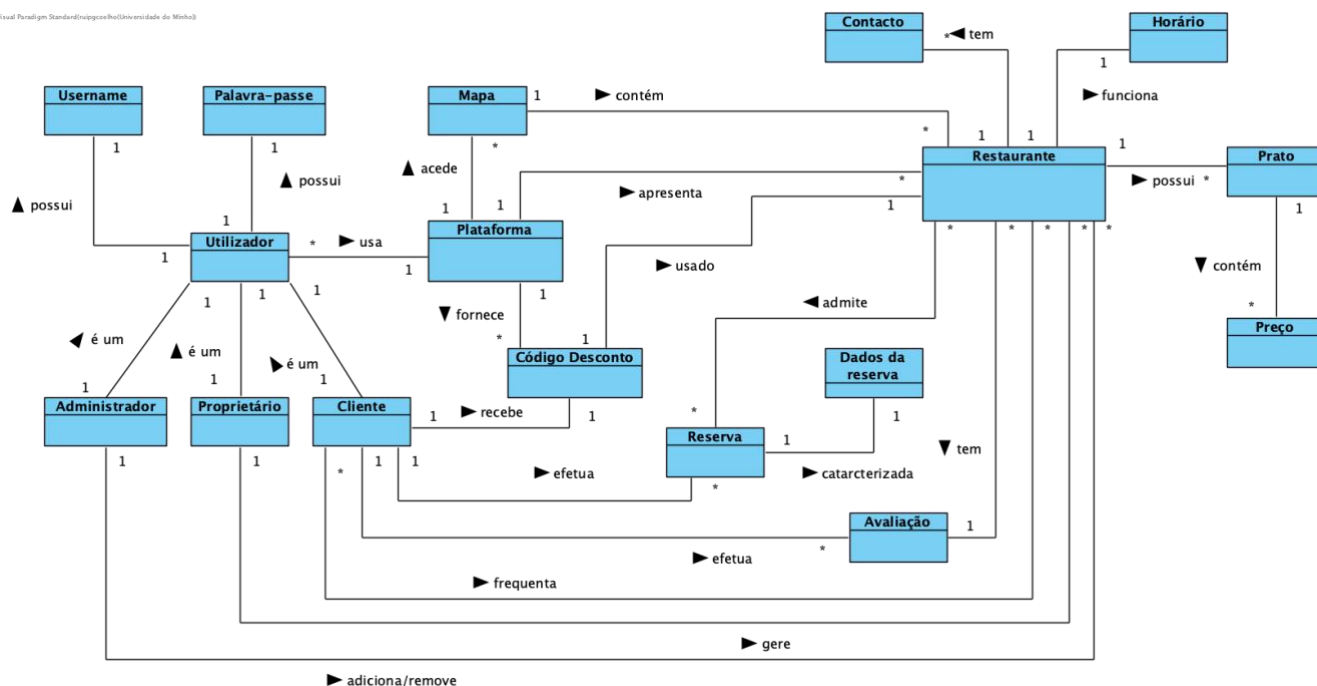


Figura 4 – Modelo de domínio do sistema.

3.3. Diagrama de Use Cases

Tal como foi supramencionado, o Diagrama de Use cases pode ser visto como a descrição de um conjunto de ações, *Use Cases*, que a plataforma a desenvolver deve executar quando interage com utilizadores – atores. Como tal, o Diagrama de *Use Cases* proposto procura efetuar esta correspondência, definindo de forma clara o poder de ação de cada ator com o sistema.

Para a definição do diagrama, tal como para a criação do Modelo de Domínio, tonou-se como base os requisitos levantados e validados pelo cliente. Em função das especificações definidas, e tendo em conta a problemática a ser trabalhada, foram identificados os seguintes atores: administrador, proprietário e cliente – podendo estes atores ser generalizados para um ator denominado de “utilizador”. Assim sendo, foram derivados os Use Cases, cf. Anexo I, que originaram o Diagrama de Use Cases¹ apresentado na Figura 5.

¹ Os *Use Cases* levantados representam os comportamentos significativos que a plataforma deve exibir.

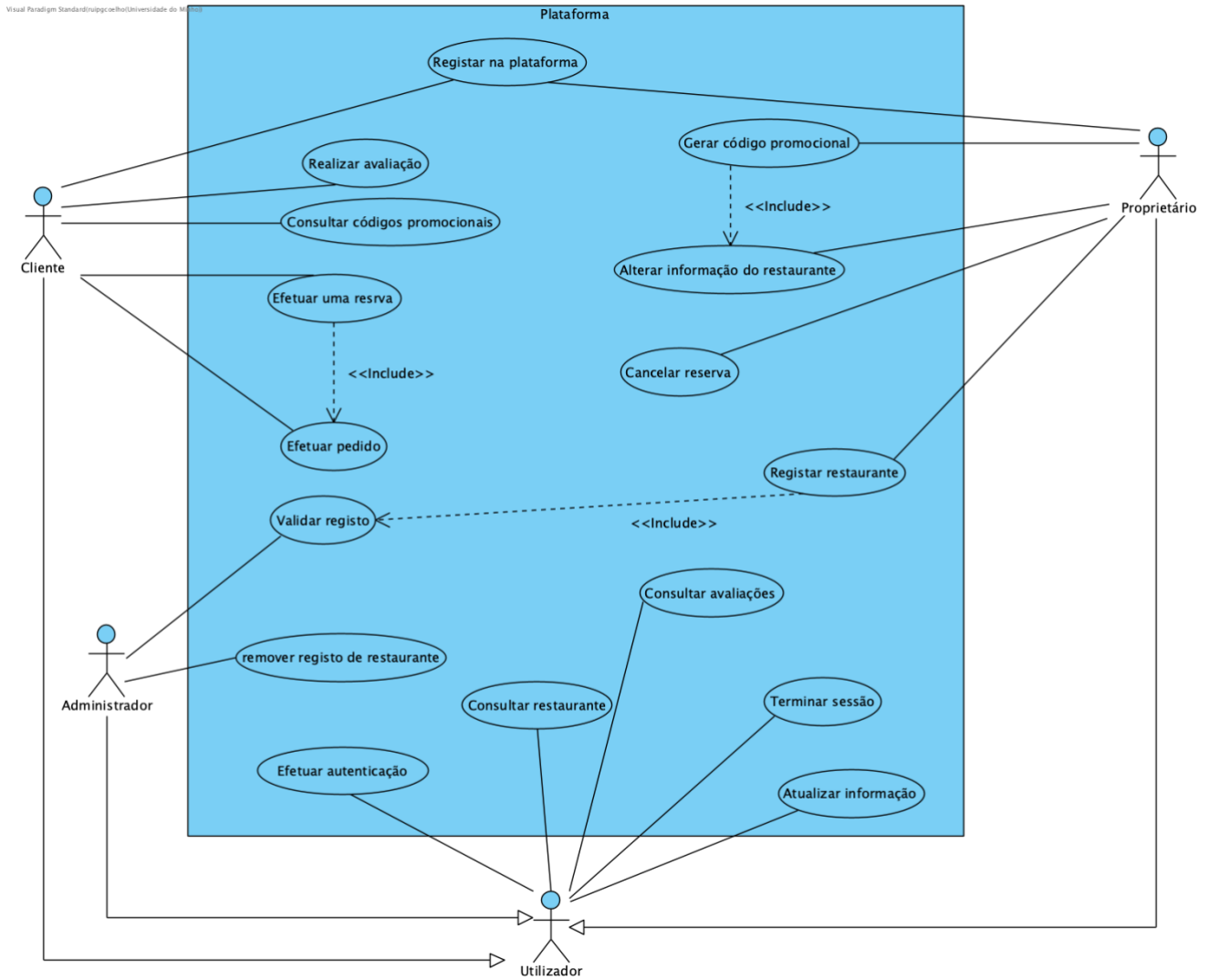


Figura 5 – Diagrama de *Use Cases*.

4. Conceção do Sistema de Dados

A construção de uma Base de Dados (BD) que armazenará a informação relevante a ser usada pela plataforma é um processo com elevada importância no desenvolvimento do projeto. De modo a proporcionar uma BD estruturada, a sua estruturação inicia-se pela criação de um modelo concetual. A criação deste modelo pretende dar a conhecer a primeira abordagem para como a BD deve ser estruturada, permitindo avaliar a sua adequação em função dos requisitos especificados e dos dados relevantes que devem ser armazenados. Assim sendo, a modelação elaborada resulta de um processo crítico e pretende capturar de forma clara as entidades e relacionamentos que permitem responder aos requisitos definidos anteriormente.

Uma vez definido o modelo concetual, segue-se o desenvolvimento do modelo lógico, que toma como base o modelo concetual definido. Este modelo apresenta informação adicional acerca da implementação da base de dados, fornecendo mais informação acerca da organização dos dados.

4.1. Modelo concetual

A especificação da BD a criar iniciou-se pela definição do seu modelo concetual, tal como foi supramencionado. Para a criação deste modelo foram consideradas as seguintes entidades:

- **Ciente:** termo que designa os clientes que se encontram registados na BD da plataforma. Cada cliente possui características próprias, como por exemplo o nome de utilizador e palavra-passe, sendo inequivocamente identificado pelo seu Número de Identificação Fiscal (NIF);
- **Proprietário:** termo que designa o dono de um restaurante. Cada proprietário possui NIF que os descreve de forma inequívoca;
- **Restaurante:** termo que refere um estabelecimento. Os restaurantes podem ser caracterizados de forma única pelo seu nome, sendo ainda descritos, por exemplo, pela sua morada;
- **Reserva:** termo que designa a reserva efetuada por um cliente num dado restaurante. A reserva é definida de forma inequívoca pelo seu Identificador (ID), tendo ainda como características a data em que foi efetuada e o número de pessoas;
- **Avaliação:** termo que refere uma apreciação elaborada por um cliente. Cada avaliação possui um ID único, tendo ainda um número de estrelas associado e um comentário;
- **Prato:** termo que designa um prato servido num restaurante. O nome do prato é um identificador inequívoco do mesmo, podendo ainda ser caracterizado pelo seu preço.

Após a especificação das entidades, foi necessário efetuar a identificação dos relacionamentos que ocorrem entre estas. Após uma análise cuidada dos requisitos e das entidades levantadas

anteriormente, foi criada uma tabela, Tabela 1, que apresenta os relacionamentos registrados, e a sua cardinalidade, entre as diversas entidades.

Entidade	Multiplicidade	Relacionamento	Multiplicidade	Entidade
Cliente	1..1	Efetua	1..N	Reserva
Cliente	1..1	Realiza	1..N	Avaliação
Restaurante	1..1	Possui	1..N	Avaliação
Restaurante	1..N	Pertence	1..1	Proprietário
Restaurante	1..N	Serve	1..N	Prato
Restaurante	1..1	Aceita	1..N	Reserva
Reserva	1..N	Engloba	1..N	Prato

Tabela 1 – Dicionário de relacionamentos.

A definição, completa, do modelo concetual² para a BD encontra-se presente na Figura 6.

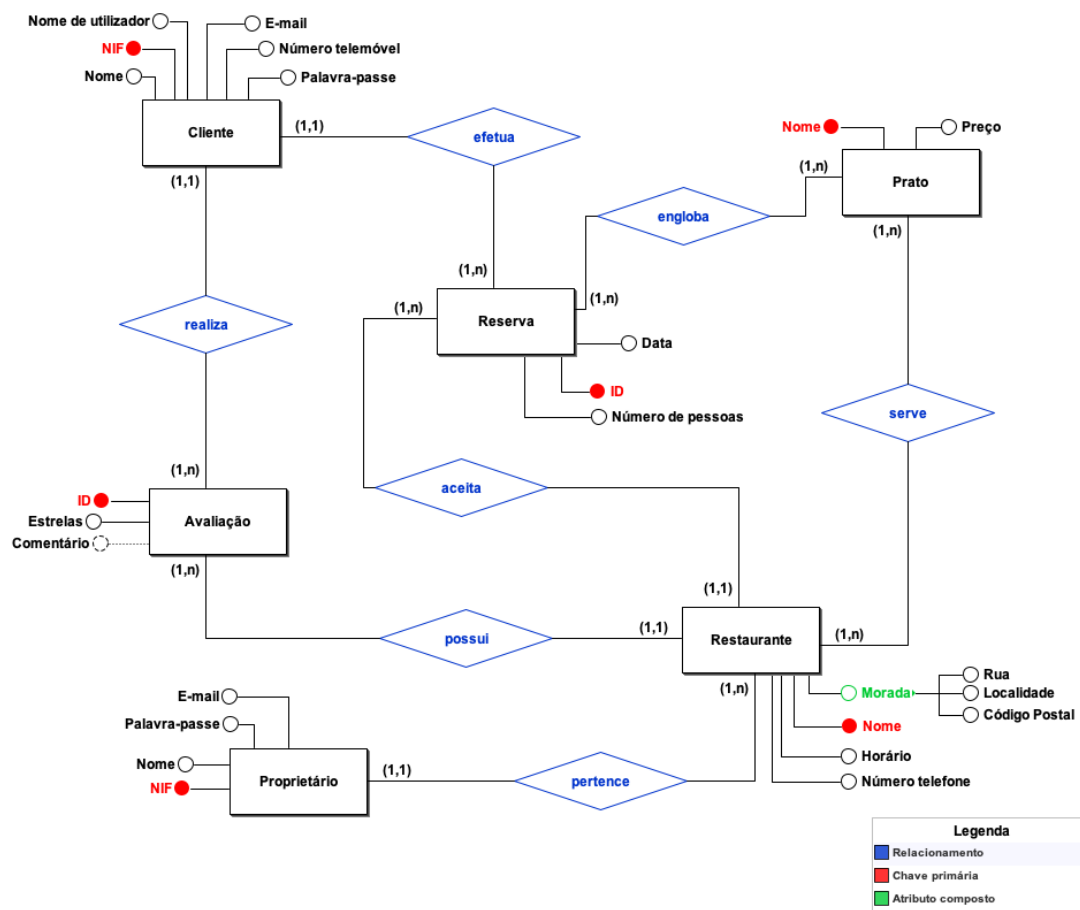


Figura 6 – Modelo concetual da BD.

² No modelo, os atributos opcionais encontram-se representados a tracejado.

4.2. Modelo Lógico

A derivação deste modelo, tal como foi anteriormente referido, parte do modelo concetual acima descrito. Para cada entidade do modelo foi criada uma relação, ou tabela, onde foram inseridos os diversos atributos que descrevem as entidades. Para a construção do modelo, considerou-se que os identificadores inequívocos de cada entidade do modelo representam, no modelo lógico, chaves primárias da relação. Adicionalmente, com vista a ser possível referenciar corretamente dados de relações diferentes, foram integradas, ainda, chaves estrangeiras nas tabelas. A título de exemplo, na relação “Reserva” foram inseridas, como chaves estrangeiras, as chaves primárias “fk_Cliente_NIF” e “fk_Restaurante_Nome” para referenciar as relações “Cliente” e “Restaurante”, respetivamente. Deste modo, é possível relacionar os dados e, por exemplo, dado um ID de reserva, obter informação referente ao cliente que efetuou a reserva e ao restaurante onde a reserva foi realizada. Adicionalmente, foram ainda discriminados os tipos de dados para cada uma das entradas das tabelas.

A Figura 7 apresenta o modelo lógico desenvolvido para a BD,

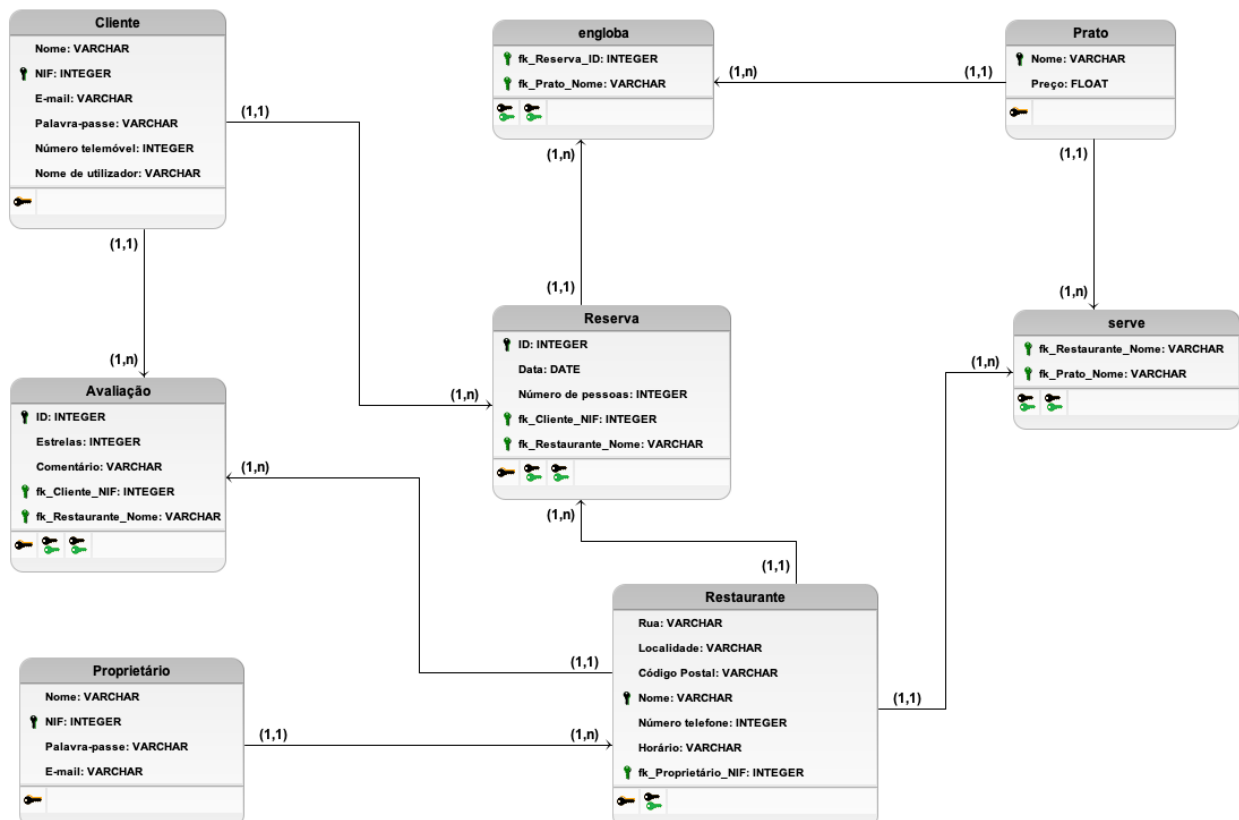


Figura 7 – Modelo lógico da BD.

5. Esboço de Interfaces do Sistema

O presente capítulo pretende dar a conhecer alguns esboços, vulgo *mockups*, criados para fornecer uma ideia genérica da interface de comunicação entre o utilizador e a plataforma. A esquematização desta interface é, agora, possível uma vez que foi conduzido um trabalho cuidadoso de especificação do *software* a desenvolver. Nesta fase, encontra-se claramente definido o domínio de ação da plataforma, assim como os requisitos a que deve responder. Adicionalmente, o esboço da estrutura da BD é também importante para a determinação da interface de comunicação, uma vez que esta representa a ligação entre o utilizador, que fornecerá dados, e a plataforma, que armazenará e trabalhará sobre esses dados.

5.1. Caracterização das interfaces

A página inicial da plataforma, esboço apresentado na Figura 8, encontra-se responsável pela autenticação e pelo registo de utilizadores. Uma vez autenticados, os diferentes utilizadores interagem com interfaces distintas, em função das permissões associadas a cada um, e.g., um proprietário terá autorização para editar as informações do seu restaurante e, como tal, a interface deve refletir a permissão dessa ação para este tipo de utilizador.

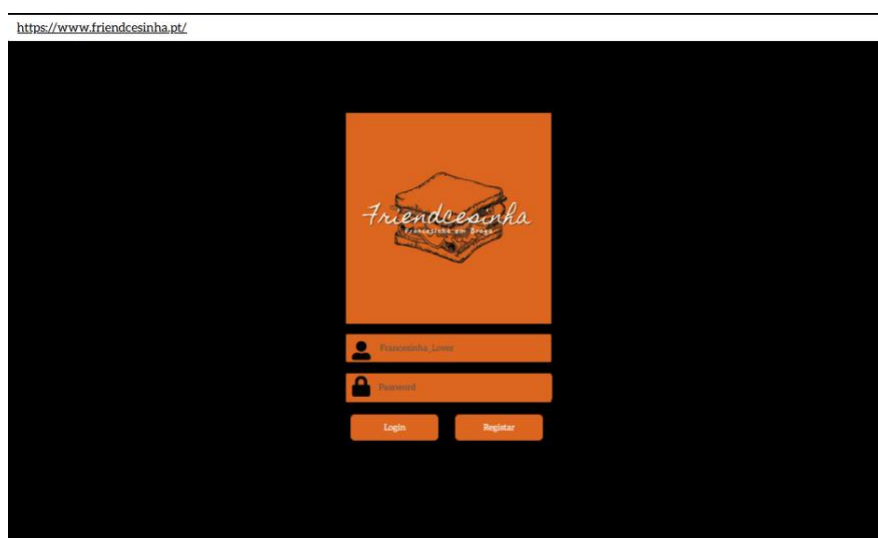


Figura 8 – *Mockup*: página inicial da plataforma.

Os esboços seguintes demonstrados centram-se na ótica do cliente, i.e., os *mockups* refletem uma possível interface de utilização específica para um utilizador do tipo cliente. Após a autenticação, o

cliente é conduzido à sua página inicial, Figura 9, que permite a seleção de diversos serviços. Entre estes serviços, pode efetuar-se o destaque para o esboço da interface do mapa, Figura 10, que permite ao cliente observar os restaurantes que se encontram registados na plataforma.



Figura 9 – Mockup: página inicial do cliente.

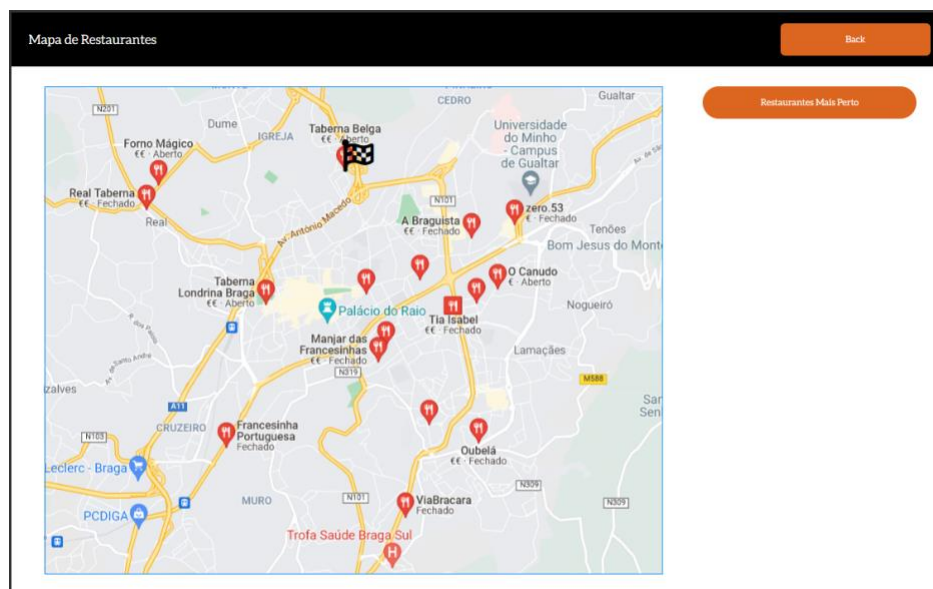


Figura 10 – Mockup: mapa com restaurantes.

Uma vez selecionado um restaurante, o cliente é levado para a página do mesmo, onde se encontra, entre outras opções, a possibilidade de obter direções ou de efetuar reservas. A Figura 11 representa o esboço dessa interface.



Figura 11 – *Mockup*: perfil do restaurante.

Os restantes esboços efetuados para a plataforma *Friendcesinha* podem ser consultados no Anexo II.

6. Conclusões e trabalho futuro

A primeira fase do trabalho desenvolvido centrou-se na fundamentação do projeto. Atendendo ao seu grau de importância, a fundamentação foi elaborada de forma cuidadosa, procurando fornecer informação que sustente e justifique o desenvolvimento do projeto.

A especificação de requisitos sucedeu a fundamentação do projeto. Tal como foi referido ao longo do presente relatório, a listagem dos requisitos, funcionais e não funcionais, é extremamente relevante, uma vez que determina a *baseline* para a criação da plataforma. Particularmente nesta fase, o papel do cliente é de elevado grau de importância, e, como tal, o seu envolvimento é imprescindível, pois é ele quem valida as especificações finais da aplicação. Assim sendo, o processo de levantamento dos requisitos resultou de uma análise cuidada das expectativas demonstradas pelo cliente para a plataforma a desenvolver, tendo o resultado final sido obtido após várias iterações e reuniões com o mesmo. Posto isto, a continuação do desenvolvimento do projeto é viável, uma vez que as especificações levantadas se encontram em conformidade com as expectativas de funcionamento da plataforma. Ainda assim, é passível, e plausível, que possam ser considerados requisitos adicionais, de modo a disponibilizar uma série de funcionalidades adicionais para a plataforma.

A modelação apresentada para a aplicação foi elaborada com igual cuidado. A criação de um modelo abstrato para a plataforma possibilita, em primeiro lugar, aprimorar o conhecimento do problema em questão. Adicionalmente, permite fornecer de um modo visual e simples a estrutura geral da aplicação, não sendo necessário conhecimento técnico para a sua compreensão. Por um lado, o fornecimento de uma modelação abstrata simples confere liberdade à equipa de desenvolvimento em termos de implementação. Por outro, o problema em análise é suscetível de ser descrito com um modelo pouco complicado, o que facilita a sua compreensão. Neste sentido, considera-se que o modelo apresentado é simples, mas completo.

A conceção do sistema de dados revelou-se uma tarefa desafiante. Anteriormente foi referido que os dados seriam mantidos numa BD relacional e, como tal, desenvolveu-se um Modelo Concetual e um Modelo Lógico para estruturar o sistema de dados. Para tal efeito, e tendo como base os requisitos, efetuou-se um levantamento dos dados relevantes para a plataforma, que possibilitaram a identificação de entidades e de relacionamentos que integrariam a base de dados. Os modelos propostos parecem ser uma boa abordagem inicial para a manutenção dos dados. Contudo, apesar de apresentar alguma robustez, as relações, vulgo tabelas, apresentadas podem, e devem, ser melhoradas, pois refletem uma primeira abordagem ao problema, carecendo, assim, posterior reflexão.

O trabalho de concetualização termina-se com a apresentação de esboços de interface. Os *mockups* apresentados são simples porque, primeiramente, pretende-se que o *layout* da plataforma seja *user friendly* e de manuseamento fácil e intuitivo. Adicionalmente, a sua simplicidade confere alguma liberdade à equipa de desenvolvimento no que diz respeito ao *design* aplicacional.

Findo o trabalho de concetualização, a equipa considera que o projeto se encontra, agora, em boas condições para prosseguir com o desenvolvimento do seu *software*.

Referências

Almeida, H. (n.d.). *As Melhores Francesinhas De Braga – Roteiro Gastronómico*.
<https://webraga.pt/blog/as-melhores-francesinhas-de-braga/>

Bessa, C. (2020, agosto 18). *A história da francesinha (e bons restaurantes para experimentar)*.
Porto secreto. <https://portosecreto.co/francesinha/>

Connolly, T. M., & Begg, C. E. (2005). *Database systems: a practical approach to design, implementation, and management*. Pearson Education.

Francesinha. (2021, maio 8). In *Wikipedia*. <https://pt.wikipedia.org/wiki/Francesinha>

Ribeiro, S. (2014, janeiro 22). *História da Francesinha*. Viaje comigo.
<https://www.viajecomigo.com/2014/01/22/historia-da-francesinha/>

Várzea, M. (2020, setembro 25). *O ABC da Francesinha: o que é, como se faz e onde comer o ex-libris gastronómico do Porto*. <https://magg.sapo.pt/comida/artigos/o-abc-da-francesinha-o-que-e-como-se-faz-e-onde-comer-o-ex-libris-gastronomico-do-porto>

Lista de Siglas e Acrónimos

API	<i>Application Programming Interface</i>
BD	Base de Dados
CFB	Confraria da Francesinha de Braga
ID	Identificador
GPS	<i>Global Positioning System</i>
NIF	Número de Identificação Fiscal
QR	<i>Quick Response</i>
UML	<i>Unified Modeling Language</i>

Anexos

I. Use Cases

Use Case		
Use Case	Registar na plataforma	
Ator	Cliente ou Proprietário	
Descrição	Permite ao utilizador registar-se na plataforma	
Pré-condição	Cliente acede à plataforma	
Pós-condição	Uma nova conta é criada na plataforma	
	Ator	Sistema
Cenário Normal	1. Utilizador seleciona opção de se registar	
		2. Sistema requisita nome de utilizador, endereço de <i>e-mail</i> e palavra-passe
		3. Sistema regista nova conta na plataforma
Fluxo alternativo (1): [Registo como Proprietário] (Passo 1)		1.1. Sistema requisita um comprovativo de propriedade
		1.2. Regressa ao passo 2
Fluxo de exceção (2): [Credenciais inválidas] (Passo 3)		3.1 Sistema informa que as credenciais fornecidas já se encontram registadas no sistema

Tabela 2 – Use Case: registar na plataforma.

Use Case		
Use Case	Consultar restaurante	
Ator	Utilizador	
Descrição	Utilizador consulta informação relevante sobre o restaurante	
Pré-condição	Utilizador encontra-se autenticado	
Pós-condição	Utilizador acede à informação sobre o restaurante	
	Ator	Sistema
Cenário Normal	1. Utilizador seleciona um restaurante	
		2. Sistema redireciona o utilizador para a página com as informações do restaurante

Tabela 3 – Use Case: consultar restaurante.

Use Case		
Use Case	Terminar sessão	
Ator	Utilizador	
Descrição	Permite ao utilizador sair da sua conta na plataforma	
Pré-condição	Utilizador encontra-se autenticado	
Pós-condição	Sessão do utilizador é terminada	
	Ator	Sistema
Cenário Normal	1. Utilizador seleciona opção de sair da conta	
		2. Sistema termina sessão do utilizador

Tabela 4 – Use Case: terminar sessão.

Use Case		
Use Case	Efetuar autenticação	
Ator	Utilizador	
Descrição	Permite ao utilizador autenticar-se na plataforma	
Pré-condição	Utilizador tem uma conta já registada	
Pós-condição	Utilizador entra na plataforma	
	Ator	Sistema
Cenário Normal	1. Utilizador seleciona opção de se autenticar	
	2. Utilizador fornece as suas credenciais	
		3. Sistema valida as credenciais
Fluxo alternativo (1): [Utilizador não se recorda da palavra-passe] (Passo 2)	2.1. Utilizador pede para recuperar palavra-passe	
		2.2. Sistema requisita <i>e-mail</i> para recuperação da conta
	2.3. Utilizador insere <i>e-mail</i>	
		2.4. Sistema envia <i>e-mail</i> de recuperação da palavra-passe
Fluxo de exceção (2): [Credenciais inválidas] (Passo 3)		3.1. Sistema informa que as credenciais são inválidas

Tabela 5 – Use Case: efetuar autenticação.

Use Case		
Use Case	Consultar códigos promocionais	
Ator	Cliente	
Descrição	Acesso e consulta de códigos promocionais	
Pré-condição	Cliente está autenticado	
Pós-condição	Cliente consulta os códigos promocionais	
	Ator	Sistema
Cenário Normal	1. Cliente pede acesso aos seus códigos promocionais	
		2. Sistema fornece os códigos ao cliente
Fluxo de exceção (1): [Códigos promocionais indisponíveis] (Passo 2)		2.1. Sistema informa que o cliente não tem códigos promocionais para utilizar

Tabela 6 – Use Case: consultar códigos promocionais.

Use Case		
Use Case	Efetuar uma reserva	
Ator	Cliente	
Descrição	Cliente efetua uma reserva num restaurante	
Pré-condição	Cliente encontra-se autenticada	
Pós-condição	Restaurante regista a reserva	
	Ator	Sistema
Cenário Normal	1. Cliente seleciona restaurante	
		2. Sistema fornece página do restaurante
	3. Cliente regista dados sobre a reserva	
		4. Sistema notifica cliente de que a reserva foi efetuada com sucesso
Fluxo alternativo (1): [Cliente efetua pedido] (Passo 4)	4.1. Cliente deseja efetuar pedido	
	4.2. <<include>> Efetuar pedido	
	4.3. Regressa ao passo 4	
Fluxo de exceção (2): [Reserva indisponível] (Passo 4)		4.1. Sistema informa o cliente de que não há disponibilidade para efetuar reservas

Tabela 7 – Use Case: efetuar uma reserva.

Use Case		
Use Case	Realizar avaliação	
Ator	Cliente	
Descrição	Cliente efetua uma avaliação de um restaurante	
Pré-condição	Cliente encontra-se autenticado	
Pós-condição	A lista de avaliações do estabelecimento é atualizada	
	Ator	Sistema
Cenário Normal	1. Utilizador seleciona restaurante	
		2. Sistema fornece página do restaurante
	3. Cliente redige avaliação	
		4. Sistema atualiza as avaliações do restaurante

Tabela 8 – Use Case: realizar avaliação.

Use Case		
Use Case	Atualizar informação	
Ator	Utilizador	
Descrição	Utilizador altera as definições da sua conta	
Pré-condição	Utilizador encontra-se autenticado	
Pós-condição	Alterações são registadas na plataforma	
	Ator	Sistema
Cenário Normal	1. Utilizador seleciona menu das definições	
	2. Utilizador insere alterações a efetuar	
		3. Sistema atualiza informações da conta

Tabela 9 – Use Case: alterar definições de conta.

Use Case		
Use Case	Validar registo	
Ator	Administrador	
Descrição	Administrador analisa e valida registos de propriedade do restaurante	
Pré-condição	Administrador encontra-se autenticado na plataforma	
Pós-condição	Processo de validação do registo terminado	
	Ator	Sistema
Cenário Normal	1. Administrador acede a lista de registos por validar	
	2. Administrador seleciona registo	
	3. Administrador valida registo	
		4. Sistema regista conclusão do processo de validação

Tabela 10 – Use Case: validar registo.

Use Case		
Use Case	Alterar informação do restaurante	
Ator	Proprietário	
Descrição	O proprietário altera informação associada ao restaurante	
Pré-condição	Proprietário encontra-se autenticado na plataforma	
Pós-condição	Informação sobre o restaurante é atualizada	
	Ator	Sistema
Cenário Normal	1. Proprietário entra na página do restaurante	
	2. Proprietário edita informação relevante	
		3. Sistema guarda novas informações

Tabela 11 – Use Case: alterar informação do restaurante.

Use Case		
Use Case	Consultar avaliações	
Ator	Utilizador	
Descrição	Utilizador consulta as avaliações efetuadas a um restaurante	
Pré-condição	Utilizador encontra-se autenticado na plataforma	
Pós-condição	Consulta concluída com sucesso	
	Ator	Sistema
Cenário Normal	1. Utilizador seleciona restaurante	
		2. Sistema devolve comentários feitos ao restaurante

Tabela 12 – Use Case: consultar avaliações.

Use Case		
Use Case	Remover registo de restaurante	
Ator	Administrador	
Descrição	Administrador remove restaurante da plataforma	
Pré-condição	Restaurante encontra-se registado	
Pós-condição	Restaurante não se encontra registado	
	Ator	Sistema
Cenário Normal	1. Administrador acede à lista de pedidos de remoção pendentes	
	2. Administrador seleciona restaurante	
		3. Sistema remove restaurante da plataforma

Tabela 13 – Use Case: remover registo de restaurante.

Use Case		
Use Case	Gerar código promocional	
Ator	Proprietário	
Descrição	Proprietário gera código promocional para ser usado no seu restaurante	
Pré-condição	Proprietário encontra-se autenticado na plataforma	
Pós-condição	Código promocional é inserido na informação do restaurante	
	Ator	Sistema
Cenário Normal	1. Proprietário cria código de desconto	
	2. <<include>> Alterar informação do restaurante	

Tabela 14 – Use Case: gerar código promocional.

Use Case		
Use Case	Efetuar pedido	
Ator	Cliente	
Descrição	Cliente efetua o pedido desejado do menu	
Pré-condição	Cliente encontra-se autenticado na plataforma	
Pós-condição	Pedido concluído com sucesso	
	Ator	Sistema
Cenário Normal	1. Cliente especifica o pedido	
	2. Cliente confirma pedido	
		3. Sistema confirma sucesso do pedido

Tabela 15 – Use Case: efetuar pedido.

Use Case		
Use Case	Cancelar reserva	
Ator	Proprietário	
Descrição	Cancelamento de reservas efetuadas pelos clientes	
Pré-condição	Proprietário encontra-se registado	
Pós-condição	Reserva encontra-se cancelada	
	Ator	Sistema
Cenário Normal	1. Proprietário consulta a lista de reservas	
	2. Proprietário seleciona reserva da lista	
	3. Proprietário cancela reserva	
		4. Sistema atualiza informação sobre as reservas
		5. Sistema notifica cliente sobre cancelamento da reserva

Tabela 16 – Use Case: cancelar reserva.

Use Case		
Use Case	Registrar restaurante	
Ator	Proprietário	
Descrição	O proprietário registra o seu restaurante na plataforma	
Pré-condição	Proprietário encontra-se autenticado na plataforma	
Pós-condição	Restaurante encontra-se registado	
	Ator	Sistema
Cenário Normal	1. Proprietário efetua o pedido para registar o restaurante	
		2. Sistema requisita informação sobre o restaurante
	3. Proprietário insere informação sobre o restaurante	
	4. Proprietário fornece comprovativo de propriedade	
		5. Sistema regista pedido
		6. <<include>> Validar registo
		7. Sistema regista restaurante e notifica proprietário

Tabela 17 – Use Case: registar restaurante.

II. Mockups

Meu Perfil

Editar perfil [Back](#)

Username:

Text Email:

Telemóvel:

Trocar password

Password Atual *

Password Nova *

Repetir Nova *

Filtro de Distância

Procuo locais perto de mim em Km

[Save](#)

[LogOut](#)

Figura 12 – *Mockup*: edição do perfil do cliente.

Avaliação Editor

Nome:

Número de Estrelas:

Faça a sua avaliação

Coloque a sua opinião aqui

Excelente!

Visitamos pela hora de almoço, ambiente tranquilo e acolhedor bem como muita simpatia por parte dos funcionários!

A francesinha era muito boa e a carne de vaca muito tenra.

[Confirmar](#) [Cancelar](#)

Figura 13 – *Mockup*: avaliação do restaurante pelo cliente.

Reserva Editor

Nome:
 Data:
 Hora:
 Número de Pessoas:

Detalhes do Pedido

Entrada:
 Prato Principal:
 Bebida:
 Acompanhamento:
 Sobremesa:

Figura 14 – Mockup: pedido de reserva.

Códigos Promocionais

[Back](#)

Desconto 10% Boas Vindas



Desconto Taberna Belga



Figura 15 – Mockup: página com códigos de desconto.