Escola de Engenharia

**Universidade do Minho**

Mestrado Integrado em Engenharia Informática

**Unidade Curricular de**

**Laboratórios de Informática IV**

Ano Lectivo de 2016/2017

**“**

***M***

***n***

***han***

***-***

***mnha***

***n***

**,**

**a**

**petece**

**-**

**me comer…algo”**

**Carlos Pereira, a61887; Diogo Silva, a76407;**

**Maria Ana de Brito, a73580; Rafael Costa, a61799**



**LI**

**4**

Março, 2017

|  |  |
| --- | --- |
| Data de Recepção |  |
| Responsável |  |
| Avaliação |  |
| Observações |  |

**“*Chewie*, apetece-me comer… algo”**

**Carlos Pereira, a61887; Diogo Silva, a76407; Maria Ana de Brito, a73580; Rafael Costa, a61799**

Março, 2017

**Resumo**

O projeto consiste no desenvolvimento de uma aplicação que compreenda todas as fases de desenvolvimento de uma peça de software. Assim, um projeto viável deve apresentar uma metodologia, processo e técnicas de desenvolvimento assentes num conjunto bem definido de etapas. Assim, este relatório foca-se na primeira etapa do desenvolvimento de *Chewie*: fundamentação, incluindo a análise dos requisitos da aplicação.

Inicialmente devemo-nos certificar em definir o contexto, expondo o caso de estudo. Após a caraterização do problema deve-se definir a motivação e quais os objetivos pretendidos atingir aquando a finalização do projeto. Para tal ser possível, é necessário justificar o sistema (salientando o contacto com o utilizador), expondo a utilidade da aplicação e identidade do projeto.

Por fim, diminuindo o nível de abstração, procede-se com uma breve definição da maqueta do sistema, permitindo uma primeira visualização do funcionamento da aplicação.

**Área de Aplicação:** Engenharia de Software, Base de Dados, Aplicações *Mobile*, Aplicações *Desktop*

**Palavras-Chave:** *Chewie*, utilizador, gastronomia, pesquisa, restaurante, avaliação, comentários, distância, sistema de recomendação

**Índice**

[1. Introdução 4](#_Toc481169126)

[1.1. Contextualização 4](#_Toc481169127)

[1.2. Apresentação do Caso de Estudo 4](#_Toc481169128)

[1.3. Motivação e Objetivos 4](#_Toc481169129)

[1.4. Justificação do Sistema 5](#_Toc481169130)

[1.5. Utilidade do Projeto 6](#_Toc481169131)

[1.6. Identidade do Projeto 6](#_Toc481169132)

[1.7. Recursos Necessários 6](#_Toc481169133)

[1.8. Maqueta do Sistema 7](#_Toc481169134)

[1.9. Conjunto de Medidas de Sucesso 8](#_Toc481169135)

[1.10. Planeamento do Desenvolvimento 9](#_Toc481169136)

[1.11. Levantamento de Requisitos 9](#_Toc481169137)

[1.11.1 Pesquisa por texto 9](#_Toc481169138)

[1.11.2 Pesquisa por voz 10](#_Toc481169139)

[1.11.3 Apresentação dos resultados 10](#_Toc481169140)

[1.11.4 Filtragem de resultados 10](#_Toc481169141)

[1.11.5 Informações a exibir 10](#_Toc481169142)

[1.11.6 Histórico de pesquisa 10](#_Toc481169143)

[1.11.7 Partilha numa rede social 10](#_Toc481169144)

[1.11.8 Comentários sobre o prato pesquisado 11](#_Toc481169145)

[1.11.9 Sugestões de restaurantes a visitar 11](#_Toc481169146)

[1.11.10 Tempo de resposta da aplicação 11](#_Toc481169147)

[1.11.11 Memória ocupada pelos dados 11](#_Toc481169148)

[1.11.12 Facilidade de utilização 11](#_Toc481169149)

[2. Análise de Requisitos 12](#_Toc481169150)

[2.1. Interface Layer 12](#_Toc481169151)

[2.1.1 Input 12](#_Toc481169152)

[2.1.1.1 Requisitos Funcionais 12](#_Toc481169153)

[2.1.2 Apresentação dos Resultados 12](#_Toc481169154)

[2.1.2.1 Requisitos Funcionais 12](#_Toc481169155)

[2.1.2.2 Requisitos Não Funcionais 14](#_Toc481169156)

[2.2. Business Layer 14](#_Toc481169157)

[2.2.1 Login 14](#_Toc481169158)

[2.2.1.1 Requisitos Funcionais 14](#_Toc481169159)

[2.2.2 Opinião 14](#_Toc481169160)

[2.2.2.1 Requisitos Funcionais 14](#_Toc481169161)

[2.2.3 Preferências 15](#_Toc481169162)

[2.2.3.1 Requisitos Funcionais 15](#_Toc481169163)

[2.2.4 Histórico 15](#_Toc481169164)

[2.2.4.1 Requisitos Funcionais 15](#_Toc481169165)

[2.3. Servidor 15](#_Toc481169166)

[2.3.1 Pesquisa 15](#_Toc481169167)

[2.3.1.1 Requisitos Não Funcionais 15](#_Toc481169168)

[2.3.2 Base de Dados 16](#_Toc481169169)

[2.3.2.1 Requisitos Não Funcionais 16](#_Toc481169170)

[3. Conclusões e Trabalho Futuro 16](#_Toc481169171)

**Índice de Figuras**

Figura 1 - Maqueta do Sistema 7

Figura 2 - Diagrama de Gantt 8

# Introdução

## Contextualização

*“Chewie*, apetece-me comer... algo" é um *software* que visa facilitar a procura de pratos de forma genérica no âmbito da região de Braga. Isto é, o utilizador será convidado a indicar o "algo" que pretende degustar e a procura indicará um conjunto de estabelecimentos num perímetro dentro do território em que o utilizador se insere. Este processo poderá ser efetuado via comunicação de voz ou introdução de texto de forma a tornar o procedimento mais rápido, conciso e eficaz.

Esta aplicação, possível de utilizar no *desktop*, também voltada à plataforma *mobile* poderá ser utilizada num contexto regional tendo os resultados naturalmente influenciados pela mesma.

*Chewie* pretende "compreender" e ajudar o utilizador a encontrar a melhor gastronomia possível, desta forma, os pratos oferecidos ao mesmo terão em conta a avaliação, as preferências de outros utilizadores e a distância ao mesmo. Pretende-se que seja encontrada a melhor solução para o utilizador.

## Apresentação do Caso de Estudo

*Chewie* é um sistema de recomendação de restaurantes com base na pesquisa de um prato gastronómico indicado pelo utilizador.

## Motivação e Objetivos

Suponhamos que temos um utilizador que desconhece a região onde se encontra, a nível gastronómico. Esse utilizador tem vontade de experimentar um prato local dessa região. Contudo existem certos ingredientes que não aprecia ou que não deve comer, mas não sabe se esses ingredientes são usados nessa região.

Imaginemos também, um outro utilizador faminto e que conhece a região onde se encontra. Contudo, este não quer pensar ou ter que fazer uma pesquisa morosa em pratos na sua zona.

Este projeto surge da necessidade de implementar um motor de busca gastronómico a nível da região de Braga, que tem em consideração as preferências do utilizador, apresentando-lhe os resultados que mais se adequam a ele. Oferecendo também ao mesmo tempo uma interface de fácil uso e funcionalidade, possibilitando a aceitação dos parâmetros de pesquisa por voz. Isto tudo porque não existe no mercado uma aplicação que consiga juntar ao mesmo tempo a pesquisa de pratos gastronómicos com as preferências do próprio utilizador e que seja de uso simples.

Pretendemos assim ter uma aplicação simples, que evite ter o utilizador a perder tempo fazendo pesquisas gastronómicas, pois será a nossa plataforma a fazer por ele. Também queremos que os resultados sejam apresentados de maneira fácil, indicando meios de contacto, avaliações, comentários e até direções ao estabelecimento que o utilizador selecionar de uma lista de resultados. Por fim almejamos em ter uma plataforma que seja rápida no tempo de resposta, visto ser um fator de grande sensibilidade no contexto da hora de refeição.

## Justificação do Sistema

O sistema que nos propomos a desenvolver tem uma aplicação prática no quotidiano dos utilizadores, uma vez que vai ser possível pesquisar os melhores restaurantes para degustação de um prato indicado pelo utilizador, segundo os seus critérios. Desta forma, quando o utilizador pesquisa um certo prato, os restaurantes à escolha ser-lhe-ão mostrados por uma ordem que conjuga vários fatores, que visam garantir a sua preferência. Se, por outro lado, pretender uma pesquisa personalizada, poderá definir quais as suas preferências de procura.

Esta aplicação também promove a mobilidade dos utilizadores, pois estes poderão realizar a pesquisa por comandos de voz. Assim, caso pretendam, poderão apenas interagir por voz com a aplicação, descartando a necessidade de introduzir dados manualmente. Esta funcionalidade é útil em situações em que o utilizador não tem qualquer possibilidade de indicar à mão qual o prato que pretende degustar. Se, por exemplo, alguém estiver a conduzir, não pode quebrar a sua atenção para introduzir os dados manualmente, correndo o risco de causar um acidente rodoviário. No entanto, o utilizador poderá usar esta aplicação, uma vez que lhe é oferecido um serviço de voz, tornando a sua condução segura.

O cliente, além de ler os comentários de um restaurante, também poderá, sempre que entender, deixar os seus próprios comentários nos restaurantes que visitou ou partilhar em direto a sua experiência gastronómica.

Além disso, a aplicação terá uma utilidade real no mercado, dado que não existe nenhuma outra que conjugue estes fatores, pois mais nenhuma permite fazer uma pesquisa de um prato gastronómico, de acordo com os critérios pessoais dos utilizadores, promovendo, também, a mobilidade dos mesmos.

## Utilidade do Projeto

O principal intuito da aplicação *Chewie* é facilitar ao cliente a procura de um prato específico. Um utilizador pode então obter várias opções de locais que oferecem o seu prato desejado independentemente do local onde se encontra. A lista de resultados é filtrada de acordo com a qualidade do restaurante e o prato procurado, bem como as preferências do utilizador. Desta maneira, se um utilizador não gostar de um prato servido de uma certa forma (com algum acompanhante, especiaria, molho, etc.), então na lista de resultados não estará incluído o restaurante que serve este prato. Os resultados são acompanhados pela morada do restaurante e respetivo trajeto até este, sendo bastante proveitoso especialmente em casos que o utilizador se encontra em cidades que não conhece.

A aplicação mantém um histórico dos restaurantes visitados por um utilizador possibilitando um rápido acesso aos resultados de uma pesquisa anterior. É possível, também, registar uma opinião de um dado restaurante ou prato numa rede social.

## Identidade do Projeto

O projeto a ser desenvolvido trata-se de um sistema de recomendação que, após a indicação de um prato gastronómico, pesquisa quais os melhores locais para realizar a sua degustação. O resultado da procura deverá mostrar uma lista dos melhores restaurantes selecionados de acordo com as preferências do utilizador, bem como as avaliações efetuadas por parte de outros utilizadores e a distância ao local em que se encontra num dado momento.

Além disso, a aplicação também se caracteriza por se enquadrar no mundo moderno, em que a maior parte dos utilizadores está ligada a uma rede social e partilha o que está a fazer ou o que sente em tempo real. Desta forma, é possível partilhar a sua experiência gastronómica num determinado restaurante, numa rede social à sua escolha ou, então, fazer um *tweet* à medida que degusta o prato escolhido.

Em síntese, a aplicação *Chewie* encontra-se direcionada para os utilizadores, de forma a proporcionar-lhes a melhor pesquisa de restaurantes, continuamente atualizada e fiável.

## Recursos Necessários

Para o desenvolvimento deste projeto é necessária uma base de dados relacional povoada com dados provenientes de alguns *sites* de opinião sobre gastronomia. Os *sites* selecionados devem fornecer as seguintes informações sobre os restaurantes: nome do estabelecimento, morada, contacto, foto, bem como comentários e avaliações feitas por utilizadores.

## Maqueta do Sistema

A aplicação tem uma arquitetura Cliente-Servidor, sendo que existem dois tipos de Cliente que o Servidor terá de servir (**Figura 1**). Os Clientes podem ser:

* Cliente *Mobile* - clientes em plataformas móveis, como *Smartphones* e *Tablets*;
* Cliente *Desktop* - clientes em ambiente *Desktop*.

Neste sistema temos um Servidor que responde aos Clientes pelas vias de comunicação apropriadas (através de *sockets)*, sendo este capaz de servir mais do que um ao mesmo tempo. Assim, desta forma, será possível ter a aplicação em dispositivos/aparelhos que poderão não ter a capacidade de aguentar os seus requisitos. Sendo assim, o Servidor vai estar ligado a uma base de dados que irá armazenar toda a informação necessária para responder aos pedidos dos vários Clientes. Para além disso, é ainda, da responsabilidade do servidor autenticar um Cliente através do seu nome de utilizador e da sua palavra-passe.

Podemos dividir o Cliente em duas camadas fundamentais: a *Interface Layer* e a *Business Layer.* A *Interface Layer* trata da interação com o utilizador. É responsável pelo tratamento de todos ospossíveis *inputs* do utilizador (texto ou voz) e pela apresentação de todos os resultados pretendidos. Já a *Business Layer* trata da extração de todos os campos relevantes de uma qualquer pesquisa, armazenamento de dados que possam ser visualizados por um utilizador (estabelecimentos, histórico de um utilizador, sugestões, …) e envia pedidos ao Servidor. É importante referir que estas duas camadas mencionadas são ligadas através de um módulo *Façade.*

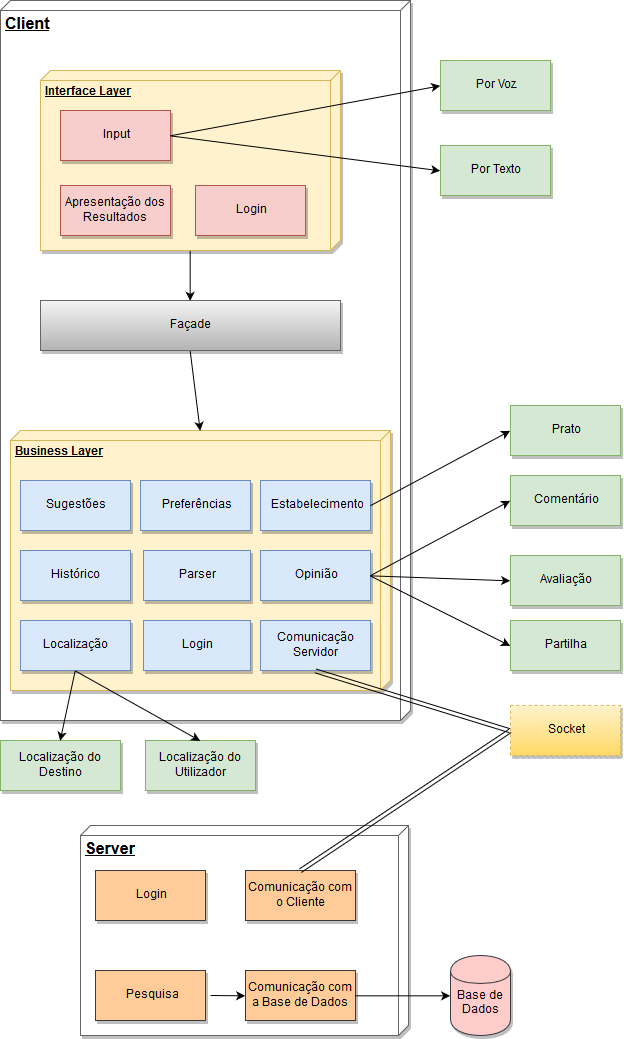


Figura 1 - Maqueta do Sistema

## Conjunto de Medidas de Sucesso

Enumera-se, de seguida, um conjunto de medidas necessárias para o sucesso deste projeto.

As pesquisas realizadas devem ser efetuadas de uma forma eficiente, em que os resultados obtidos vão de acordo com as preferências do utilizador e os locais indicados apresentam a melhor qualidade possível. A *User Interface* deve ser bastante intuitiva, simples e com um aspeto cativante. O desenvolvimento do projeto deverá ser bem planeado de modo a reduzir o número de horas gastas, bem como o custo de produção deste.

Para um bom funcionamento da aplicação devem-se realizar um conjunto de testes para verificar cada uma das suas funcionalidades em separado. Para a pesquisa, pode-se definir um domínio finito (um local específico) e verificar a validade dos dados obtidos para esse domínio. De modo a garantir a acessibilidade da *User Interface,* a aplicação deverá ser testada por um determinado conjunto de utilizadores que não tenham conhecimento da implementação desta. Para complementar esta informação, podemos realizar um questionário simples acerca da qualidade da aplicação, facilidade de utilização, rapidez dos resultados e consistência dos mesmos.

Todos os requisitos, apresentados nas secções abaixo, devem estar bem implementados dentro do prazo previsto.

## Planeamento do Desenvolvimento

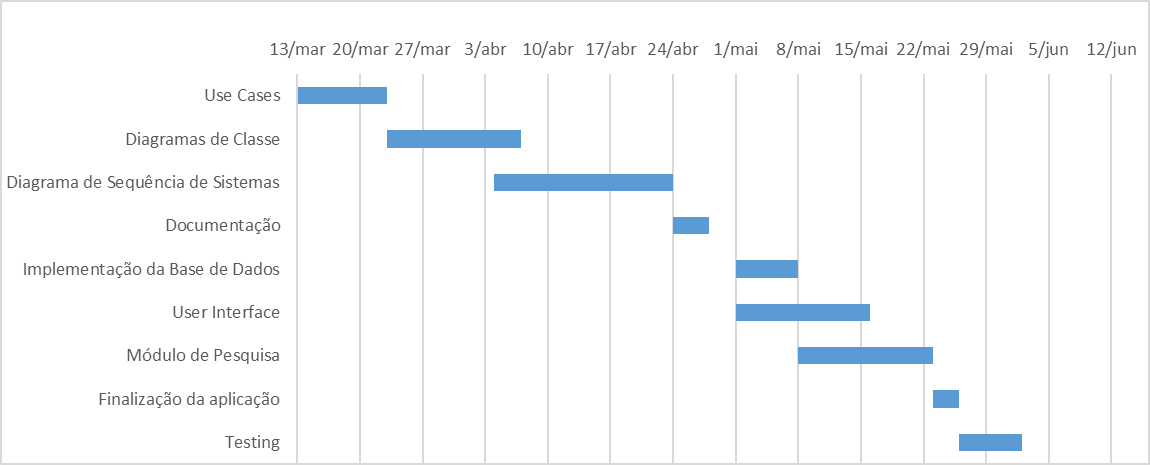


Figura 2 - Diagrama de *Gantt*

No diagrama acima é apresentado o planeamento previsto para as fases futuras deste projeto. Assim, a fase 2 será desenvolvida desde do dia 13 de março até ao dia 28 de abril.

Por outro lado, a fase 3 do projeto terá início no dia 1 de maio e será concluída entre o dia 29 de maio e 5 de junho.

## Levantamento de Requisitos

### Pesquisa por texto

O utilizador tem a possibilidade de introduzir os parâmetros da pesquisa, manualmente, através de um teclado que lhe será apresentado (ou que esteja incorporado na máquina). Aqui, o utilizador apenas introduzirá o prato gastronómico que pretende, podendo também adicionar elementos que ache necessários. Por exemplo, pesquisar por “pastéis de nata sem sabor a limão” ou “pão de ló de ovos moles”.

### Pesquisa por voz

Com esta funcionalidade queremos promover a mobilidade do cliente da aplicação, permitindo-lhe realizar uma pesquisa através da voz. Para isso, o utilizador dispõe de um botão reservado para iniciar a pesquisa. Ou seja, ao clicar neste botão, a aplicação sabe que vai procurar pelos parâmetros que vai receber por voz.

### Apresentação dos resultados

Feita a pesquisa, a ordem de apresentação dos restaurantes conjuga vários fatores, sendo eles: o número de estrelas (avaliação), o número de comentários feitos por utilizadores e a distância entre o estabelecimento e o cliente. A relevância dos resultados surge da combinação destes três critérios, dando-se maior importância à avaliação dos mesmos. Considera-se então que a avaliação e o número de comentários terão um peso positivo na ordem da apresentação. Por outro lado, quanto maior for a distância ao local a partir da posição do cliente, maior será o impacto negativo na apresentação, logo mais baixo será o seu lugar na lista.

### Filtragem de resultados

Enquanto que os fatores descritos no ponto anterior são por defeito, o utilizador poderá refinar a apresentação de resultados, indicando novas opções de filtragem. Por exemplo, número de estrelas mínimo e/ou distância máxima até ao local.

### Informações a exibir

Para cada restaurante que se encontre na lista de resultados, serão apresentadas as seguintes informações ao cliente: nome, foto e avaliação. Desta forma, o utilizador poderá selecionar um estabelecimento com base na sua preferência. Feita a seleção, as informações serão expandidas, passando a incluir: descrição, contactos, comentários, horário de funcionamento e o trajeto ao destino.

### Histórico de pesquisa

Cada pesquisa efetuada pelo utilizador será guardada pela aplicação, de forma a que este possa reconsultar pesquisas anteriores e utilizá-las novamente

### Partilha numa rede social

Acompanhando as tendências tecnológicas e sociais, o cliente poderá partilhar numa rede social o restaurante e o prato que vai consumir. Assim, irá manter os seus seguidores atualizados acerca das suas escolhas de degustação.

### Comentários sobre o prato pesquisado

De forma a contribuir para a avaliação de um restaurante que visitou, o utilizador poderá comentar sobre a sua experiência gastronómica naquele estabelecimento. Assim, outros clientes poderão ter uma melhor compreensão acerca desse local.

### Sugestões de restaurantes a visitar

Com o intuito de valorizar e distinguir esta aplicação em relação a outras plataformas, o sistema fornece uma lista de restaurantes atualmente mais populares na região onde o utilizador se encontra. Assim, caso ainda não saiba o que pretende degustar, o cliente poderá selecionar a opção Sugestões que lhe irá sugerir os restaurantes com melhor avaliação e que ainda não teve a oportunidade de visitar.

### Tempo de resposta da aplicação

É esperado que o tempo de resposta seja impercetível ao utilizador, uma vez que um sistema deste tipo não pode permitir esperas demoradas.

### Memória ocupada pelos dados

Importa que os dados armazenados localmente ao utilizador (*Mobile* ou *Desktop*) ocupem o mínimo possível. Isto porque não pretendemos que o dispositivo do cliente fique sobrecarregado pela aplicação.

### Facilidade de utilização

A *User Interface* deve estar preparada para qualquer tipo de utilizador, independentemente do seu nível de experiência com a plataforma que estão a utilizar.

# Análise de Requisitos

## *Interface Layer*

### Input

### Requisitos Funcionais

* **Pesquisa por texto -** Quando o utilizador pretender realizar a sua pesquisa textualmente, ser-lhe-á apresentada uma barra branca na janela da aplicação. Aqui o cliente deverá introduzir os parâmetros da sua procura. Dependendo do tipo de cliente, a introdução é feita de maneira diferente. Se a pesquisa for feita num dispositivo desktop, o teclado está naturalmente presente na máquina. Por outro lado, caso seja feita num dispositivo mobile, será utilizado o teclado que se tem por defeito nesses sistemas. Introduzidos os parâmetros, o cliente deve selecionar o botão apropriado para que a pesquisa seja iniciada.
* **Pesquisa por voz -** Caso o utilizador não queira ou não consiga fazer a pesquisa por texto, será dada a possibilidade de indicar o que pretende através da sua voz. Para dar início à pesquisa, deverá premir o botão apropriado. Ao selecionar este botão, a aplicação entra em modo de captura de voz, permanecendo neste estado por um período de tempo curto. Aqui, tudo que for dito ao dispositivo será convertido para texto que por sua vez é utilizado para efetuar a pesquisa. A captura de voz termina automaticamente após o utilizador não indicar nada por alguns instantes de tempo.

### Apresentação dos Resultados

### Requisitos Funcionais

* **Apresentação dos resultados** - Efetuada uma pesquisa, é apresentado ao utilizador uma lista de restaurantes que satisfazem os parâmetros introduzidos. A ordem dos resultados é influenciada pelos seguintes critérios: avaliação do estabelecimento, número de comentários feitos e distância ao local de onde foi efetuada a pesquisa.

Destes fatores, distinguem-se os que exercem uma influência positiva na ordem de apresentação: avaliação e número de comentários. Entre estes, a avaliação do estabelecimento é a que tem mais influência. Quanto maior for o número de comentários, pressupõe-se que a avaliação do estabelecimento é mais fidedigna. Assume-se que entre um restaurante com uma avaliação de 4/5 e 100 comentários e um restaurante com a mesma avaliação, mas apenas com 10 comentários, a avaliação do primeiro será a mais fiável. Por esta razão conjuga-se a avaliação com o número de comentários.

Por outro lado, quanto maior for a distância ao local pior será a sua posição na lista de resultados. Assim, evita-se que os restaurantes mais longínquos ocupem os primeiros lugares da lista.

* **Histórico de pesquisa** - Sempre que um utilizador selecionar a aba histórico, ser-lhe-á apresentada uma lista de pesquisas ordenadas da mais recente para a mais antiga. Ao selecionar uma pesquisa já feita, o cliente é encaminhado para o respetivo estabelecimento. Devido ao número elevado de procuras que poderão ocorrer, o utilizador terá de fazer *scroll down* de forma a visualizar o histórico mais antigo. Do histórico podem ser eliminados estabelecimentos que foram visitados pelo utilizador.
* **Sugestões de restaurantes a visitar** - É dado ao utilizador a hipótese de ser a aplicação a sugerir-lhe o próximo estabelecimento que deverá visitar. A opção *sugestões* apresenta os cinco melhores estabelecimentos que este ainda não tenha visitado. A ordem de apresentação é por avaliação.
* **Filtragem de resultados** - Apresentados os resultados da pesquisa, o utilizador tem a possibilidade de refiná-los, selecionando as opções de interesse. Para tal, o cliente tem de selecionar um botão que vai apresentar-lhe um menu para definir as opções do refinamento.

O aperfeiçoamento dos resultados pode ser feito na distância máxima ao local, na avaliação mínima do estabelecimento e/ou se encontra em horário de funcionamento no momento. Introduzidos os campos de personalização da pesquisa, o utilizador seleciona o botão designado para registá-los. Logo de seguida, são apresentados os resultados que cumprem os fatores pretendidos.

* **Informações a exibir** - A lista dos resultados contém informações mais genéricas para cada um dos restaurantes. Aqui apresenta-se: nome, foto e avaliação. Consideram-se estas as informações mais relevantes para que o utilizador possa formar uma primeira impressão sobre o estabelecimento.

Selecionado o restaurante, acrescentam-se às informações iniciais: descrição do estabelecimento, contactos, *website*, horário de funcionamento, comentários e trajeto ao destino.

Quantos aos comentários, o utilizador poderá navegar sobre a lista apresentada. O local é apresentado numa janela da *Windows Maps*, em que ao seleciona-la é calculado o percurso ao destino.

### Requisitos Não Funcionais

* **Facilidade de utilização** - A *User Interface* é intuitiva ao ponto de ser possível a um utilizador, sem experiência na área da informática, conseguir efetuar uma pesquisa, navegar e refinar os resultados, selecionar um estabelecimento e inclusive partilhar numa rede social.

## *Business Layer*

### Login

### Requisitos Funcionais

* ***Login* no sistema** - O Cliente pode registar-se na aplicação ou, caso já o tenha feito, efetuar o *login*. Isto serve para que o Cliente possa trocar entre dispositivos (*mobile* ou *desktop*), sem perder as suas preferências de pesquisa ou histórico, como também não permitir que avalie/comente o mesmo estabelecimento mais que uma vez. Para que uma pesquisa possa ser feita, o Cliente tem que estar autenticado.

### Opinião

### Requisitos Funcionais

* **Partilha numa rede social** - No *Chewie*, selecionado um estabelecimento, o utilizador pode escolher a opção de partilha, sendo-lhe fornecido um conjunto de redes sociais disponíveis. Depois, este é redirecionado para a aplicação respetiva da rede social ou para o *browser.*
* **Comentários sobre o estabelecimento visitado** - Sempre que um utilizador selecionar um estabelecimento, é-lhe dada a opção de efetuar um comentário sobre o prato gastronómico e/ou estabelecimento. É apresentada uma janela ao utilizador para que possa inserir o seu comentário. Um utilizador apenas pode comentar um estabelecimento uma única vez.
* **Avaliação sobre o estabelecimento visitado** - Sempre que um utilizador selecionar um estabelecimento, é-lhe dada a opção de avaliar o estabelecimento visitado. A avaliação é um número inteiro compreendido entre 1 e 5, que é guardada diretamente na Base de Dados da aplicação. Uma janela é apresentada para que o utilizador faça a sua avaliação, podendo apenas dar uma única avaliação a um estabelecimento.

### Preferências

### Requisitos Funcionais

* **Informação para as preferências do utilizador** - Um utilizador indica numa pesquisa o prato pretendido e, caso queira, um conjunto de parâmetros adicionais que representam o seu gosto pessoal. Ao indicar que visitou um estabelecimento, é guardada na Base de Dados, para aquele prato, os parâmetros adicionais que inseriu anteriormente. Estes representam as suas preferências, no que diz respeito ao prato pesquisado. Quanto maior o número de vezes que pesquisar o mesmo prato com esses parâmetros, maior será a sua relevância numa pesquisa futura, para aquele prato.

### Histórico

### Requisitos Funcionais

* **Histórico de Pesquisa** - Na aplicação existe uma aba chamada histórico que armazena todas as pesquisas efetuadas. Para cada utilizador são guardados os estabelecimentos que este indica como visitados, como também a data.

## Servidor

### Pesquisa

### Requisitos Não Funcionais

* **Tempo de resposta da aplicação** - Pretende-se que o tempo de resposta da aplicação seja o mais curto possível, uma vez que é um fator de grande importância na hora da refeição. Assim, a aplicação deverá apresentar uma lista de resultados num período de tempo inferior a cinco segundos. Desta forma, o tempo de resposta não interfere de maneira negativa na experiência que o utilizador tem com a aplicação.
* **Limitação da região** - A pesquisa de pratos gastronómicos é limitada à região de Braga.

### Base de Dados

### Requisitos Não Funcionais

* **Memória ocupada pelos dados** - A aplicação tem uma arquitetura Cliente-Servidor. O espaço ocupado pelos dados no Cliente é mínimo, isto porque o volume de informação a ser acedido pela aplicação é muito elevado, ou seja, o Servidor é quem guarda e tem acesso direto aos dados. O Cliente terá guardado o seu *username* e a respetiva *password*, de forma a poder ser identificado no momento de *login*.

# Conclusões e Trabalho Futuro

O intuito da etapa de especificação do projeto consiste na análise (com bastante detalhe) de cada componente do projeto. Este processo é bastante útil para aumentar a robustez, segurança e versatilidade de um projeto. Para além disso permite uma resolução antecipada de inúmeros problemas ou *bugs* que poderiam vir a decorrer no processo de construção. No fundo, este método serve para aumentar a taxa de sucesso de um projeto.

Baseando-nos (o máximo possível), na nossa especificação, a próxima etapa consiste na construção do produto final. Tendo em conta o prazo previsto de entrega do mesmo, dar-se-á prioridade às funcionalidades básicas para o bom funcionamento da aplicação *Chewie,* sendo estas, a povoação da base de dados, implementação da pesquisa e obtenção dos resultados e construção da *User Interface*. Depois de todas estas funcionalidades básicas estarem completamente operacionais serão acrescentas à aplicação o maior número de funcionalidades extras que se conseguir.