



Universidade do Minho

Mestrado Integrado em Engenharia Informática
Licenciatura em Ciências da Computação

Unidade Curricular de Laboratórios de Informática IV

Ano Lectivo de 2018/2019

CHELI4

**Henrique Faria, João Marques, José Pereira,
Ricardo Petronilho**

Junho, 2019

LIV

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

CHELI4

**Henrique Faria, João Marques, José Pereira,
Ricardo Petronilho**

Junho, 2019

Resumo

Este relatório é relativo ao processo de desenvolvimento de um assistente de cozinha para um qualquer utilizador que pretenda realizar os seus 'cozinhados' preferidos bem como gerir uma ementa de refeições.

No relatório é mostrada a Fundamentação do processo de desenvolvimento para a nossa aplicação em questão e a Especificação dos requisitos levantados bem como o Sistema de Base de Dados e o Sistema de Interfaces a implementar.

Área de Aplicação: Desenvolvimento de um Sistema de Software.

Palavras-Chave: Aplicação, Cheli4, Assistente Pessoal, Multiplataforma, Personalização, Base de Dados, Receitas.

Índice

| | |
|-------------------------------------------------------|----|
| Resumo | 1 |
| Índice | 4 |
| Índice de Figuras | 6 |
| 1. Fundamentação da aplicação | 7 |
| 1.1. Contextualização | 7 |
| 1.2. Motivação e Objetivos | 7 |
| 1.3. Definição de identidade do sistema a desenvolver | 8 |
| 1.4. Justificação, viabilidade e utilidade do sistema | 8 |
| 1.5. Identificação dos recursos necessários | 9 |
| 1.6. Modelo do sistema a implementar - Maquete | 10 |
| 1.7. Definição de medidas de sucesso | 11 |
| 1.8. Plano de desenvolvimento | 11 |
| 2. Especificação da Aplicação | 13 |
| 2.1. Levantamento de Requisitos | 13 |
| 2.2. Análise de Requisitos | 13 |
| 2.3. Modelo Domínio | 14 |
| 2.4. Uses cases especificados | 17 |
| 2.4.1. Pesquisar receita | 18 |
| 2.4.2. Interpretar comando de voz | 18 |
| 2.4.3. Selecionar receita | 20 |
| 2.4.4. Realizar receita | 21 |
| 2.4.5. Quiz - Questionário | 22 |
| 2.5. Sistema da base de dados | 23 |
| 2.6. Sistema da interface gráfica | 24 |
| 2.7. Diagrama de estados | 29 |
| 3. Implementação | 31 |
| 3.1. Servidor de base de dados | 31 |
| 3.2. Arquitetura da aplicação | 31 |
| 3.3. Interface Gráfica | 32 |

| | |
|-----------------------------|----|
| 3.4. Assistente Pessoal | 33 |
| 4. Segurança | 35 |
| 5. Utilização da Aplicação | 36 |
| 6. Validação do Projeto | 37 |
| Conclusão | 38 |
| Lista de Siglas e Acrónimos | 40 |

Índice de Figuras

| | |
|----------------------------------------|----|
| Figura 1: Maquete da aplicação CHELI4. | 10 |
| Figura 2: Diagrama de Gantt. | 11 |

1. Fundamentação da aplicação

1.1. Contextualização

A empresa Continente é uma insígnia de hipermercados pertencente à Sonae MC. Foi a primeira cadeia de hipermercados em Portugal, e mantém-se até a data como uma referência no sector de retalho alimentar do país.

Com o objetivo de se destacar das restantes empresas, decidiu expandir a sua área de negócios e entrar no mercado das receitas culinárias, através de uma app grátis que fornece um serviço de leitura de receitas e controlo automático ativado por voz (hands off).

Querendo obter rapidamente muitos utilizadores, esta especificou que pretendia uma forma de chegar a cada utilizador sem que importasse o tipo de dispositivo que este estivesse a usar.

A aplicação tem incorporada outras funcionalidades secundárias também importantes que serão mais à frente especificadas.

1.2. Motivação e Objetivos

O projeto realizado teve como objetivo a criação de uma aplicação Web para ajudar um utilizador de qualquer idade a preparar o seu 'cozinhado' preferido sem grande dificuldade, tendo sido criada uma interface de fácil utilização.

Os motivos que levaram a equipa de trabalho a realizar tal projeto foram primeiramente a geração de lucro e ao mesmo tempo proporcionar maior eficiência a todos os utilizadores quando concretizam os seus 'cozinhados' preferidos sem as dificuldades que anteriormente encontravam, como por exemplo: o utilizador pode rever a receita sem a necessidade de manusear o dispositivo e ter que parar para o conseguir fazer.

Em suma, a realização deste assistente de cozinha tem como objetivo tornar o processo de cozinhar mais prático, acessível e divertido.

1.3. Definição de identidade do sistema a desenvolver

A aplicação é denominada por CHELI4, propriedade do grupo Sonae MC.

A plataforma é universal e por isso foi implementada num Web Browser. A aplicação centrasse na culinária Portuguesa. A idade recomendada para o uso da app é 16 anos ou superior devido à maturidade do indivíduo.

A app comunica com o utilizador através de texto e/ou áudio e o utilizador comunica com a mesma através de voz ou clique. O idioma é português no entanto, para trabalho futuro, pretendesse generalizar o idioma.

O objetivo primário da aplicação é melhorar a eficácia na consulta e elaboração de receitas. De forma a aumentar o lucro gerado da empresa implementaram-se outras funcionalidades secundárias como:

- a aplicação de desconto em vendas no grupo Sonae MC;
- possibilidade de agendamento semanal das refeições;
- mapas com localização de postos de venda de consumíveis e como lá chegar;
- armazenamento de preferências de cada utilizador para um serviço mais personalizado.

Por último, não sendo uma funcionalidade para o utilizador, mas mais uma forma de lucro para a empresa foi implementada a apresentação de publicidades.

1.4. Justificação, viabilidade e utilidade do sistema

A implementação da aplicação num Browser torna o seu acesso universal e por isso abrange um maior número de utilizadores. Esta implementação permite também uma maior escalabilidade e manutenção facilitada uma vez que se for necessário adicionar/remover uma funcionalidade do sistema apenas se altera numa só plataforma, caso contrário se a aplicação fosse, por exemplo, implementada distintivamente no iOS e no Android as alterações teriam de ser realizadas em ambas as plataformas.

Sendo o Cheli4 uma aplicação hands off permite ao seu utilizador não tocar no dispositivo enquanto faz o seu cozinhado para não correr o risco de danificar o seu dispositivo. Outra vantagem é que não necessita de ir a sites externos procurar informações sobre o cozinhado que está a preparar porque em caso de dúvida sobre alguma expressão a própria aplicação consegue explicar ao utilizador como resolver. Para além disso até a pessoa mais preguiçosa conseguiria fazer o seu preparado preferido sem grande dificuldade pois a aplicação Cheli4 usa um vocabulário simples e sucinto para fácil compreensão de todos os passos e expressões que suscitem alguma dúvida. Como se já não bastasse todos os passos vão ter

associados as suas durações sendo que depois de lido o passo a aplicação começa a contar o tempo e avisa o utilizador quando o tempo chega ao fim para este ter uma maior noção do tempo que passou desde que inicializou o passo. Com alguns preparados realizados qualquer um poderá ser o próximo “Master Chef”.

Todos os dados serão armazenados externamente ao dispositivo do cliente, desta forma o espaço disponível de armazenamento do dispositivo do utilizador não condiciona o uso da aplicação.

A existência de uma conta de utilizador permite registar preferências e dados pessoais dos utilizadores, proporcionando uma experiência mais pessoal e personalizada

1.5. Identificação dos recursos necessários

Os recursos necessários ao desenvolvimento da aplicação são os seguintes:

Recursos Humanos:

- 1 Gestor de Projeto (Gere projeto e distribui trabalho pelos trabalhadores)
- 2 Engenheiros de Software (Calculam custos, receitas e viabilidade do projeto - software, mão de obra)
- 4 Programadores:
 - 1 criação da base de dados, povoamento e manutenção
 - 1 criação e desenvolvimento do assistente de voz
 - 1 pesquisa de locais em mapas, direções e agenda de refeições
 - 1 criação do site e ligação do mesmo à base de dados

Softwares a utilizar:

Software de gestão:

- Microsoft Project (Gestão das tarefas atribuídas a cada elemento da equipa)

Software de análise de funcionamento da app:

- Visual Paradigm (diagramas UML para representação do funcionamento da aplicação)

Software para a base de dados:

- Microsoft SQL Server (base de dados relacional em SQL)

Software de desenvolvimento de funcionalidades da app:

- Microsoft Visual Studio (C#)
- Bing Maps Platform (mapas localizações e direções)

- Microsoft Speech Platform (reconhecimento de voz - direção do utilizador para o dispositivo)
- Microsoft Cortana (assistente de voz - direção do dispositivo para o utilizador)

Software Relatório:

- Microsoft Office Word

Tempo estimado do Projeto (14 semanas):

- Fundamentação (2 semanas)
- Especificação (4 semanas)
- Construção (7 semanas)
- Testes (1 semana)

Receitas:

- Coleção de livros de receitas de Manuela Freitas, ACTIVA

1.6. Modelo do sistema a implementar - Maquete

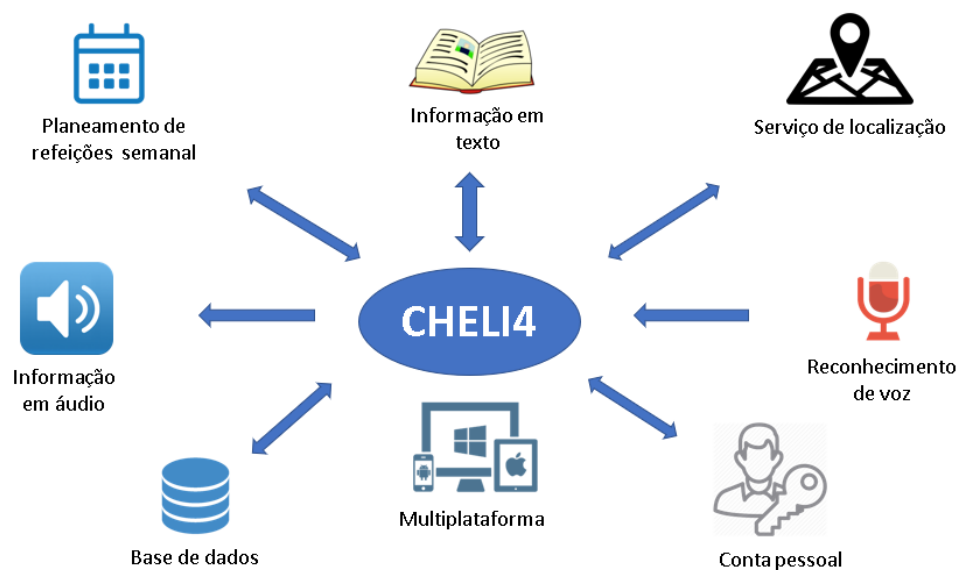


Figura 1: Maquete da aplicação CHELI4.

As setas bidirecionais indicam input e output enquanto as setas unidirecionais indicam input ou output.

A aplicação recorre a um Servidor de Base de Dados para armazenar informação relativa a receitas e utilizadores.

As informações relacionadas com a conta pessoal são armazenadas também na Base de Dados.

1.7. Definição de medidas de sucesso

Ao utilizar a app é gerado um código de desconto aplicável em lojas do grupo Sonae MC, desta forma é possível aumentar o número de clientes que usam a aplicação e publicitar as próprias empresas do grupo.

Para além disso, existe a possibilidade de incorporar anúncios de empresas terceiras de forma a maximizar os lucros gerados pela aplicação.

A aplicação fornece um serviço de escolha seletiva de postos de venda pré-selecionados das empresas com parceria no ramo alimentício da Sonae. A apresentação da localização usa a política de maior proximidade.

Todos os tópicos referidos acima, são medidas de sucesso para a empresa que solicitou a elaboração da aplicação CHELI4, pelo que, os principais objetivos, são o aumento das vendas, e publicidade dos seus produtos.

1.8. Plano de desenvolvimento

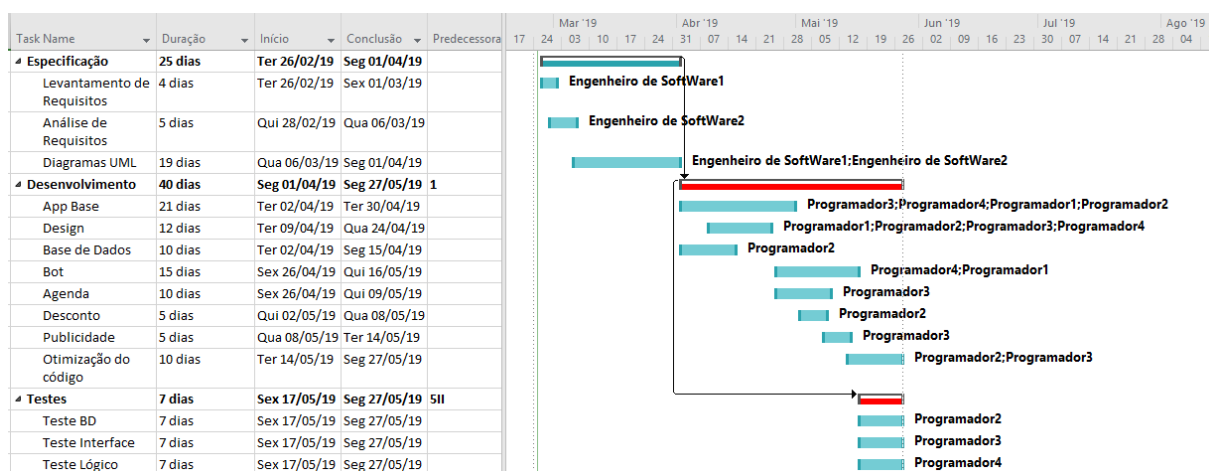


Figura 2: Diagrama de Gantt.

A fase especificação teve uma duração de 25 dias, pelo que foi feito o levantamento e análise de requisitos e a elaboração dos diagramas UML.

A fase desenvolvimento, teve uma duração de 40 dias, e é composta pelas tarefas de App Base, Design, base de dados, funcionalidades(bot, agenda, desconto, publicidade) e otimização de código.

Por fim e não menos importante a fase de Testes, é constituída pelos testes da base de dados, interface e lógica da aplicação.

É importante referir que a fase de desenvolvimento foi iniciada apenas após a conclusão da primeira fase, enquanto que a fase de testes foi elaborada paralelamente com a fase de desenvolvimento.

2. Especificação da Aplicação

2.1. Levantamento de Requisitos

O processo de levantamento de requisitos, foi elaborado com algumas reuniões com a equipa da SONAE, onde foram discutidas as funcionalidades que os mesmos queriam ver realizadas na aplicação, e através de inquéritos/questionários realizados aos clientes dos hipermercados do grupo SONAE, para perceber o tipo de dificuldades que eles encontram na hora de cozinhar e o que os levaria a usar uma aplicação com um assistente de voz.

Assim, chegou-se ao conjunto de requisitos, explicados abaixo, organizados em requisitos de sistema e utilizador.

2.2. Análise de Requisitos

Requisitos de sistema:

- Guardar/Registar informação sobre o utilizador, como nome, username, password, data nascimento, email, altura, peso, estado da conta (se foi apagada ou não).
- Autenticar utilizadores através de username e password.
- Apresentar receitas sem e com filtros.
- Sem filtro: Através das predefinições do utilizador.
- Com filtro: São apresentadas apenas as que fazem match com os mesmos (Carne, Peixe, vegetariano, vegan, fast-food, pequeno almoço, prato, etc...;Filtrar por produto.)
- Efetuar receitas, explicando/ditando passo a passo cada processo para a realização da mesma, através de texto ou do assistente pessoal.
- Assistente pessoal, presente na realização da receita para:
 - Ditar/explicar os passos;
 - Ditar determinadas expressões que utilizador não tenha entendido (exemplo: explicar o que significa claras em castelo);
 - Controlar os tempos para execução da receita e passos;
 - Realizar questionário, perguntas relacionadas com a dificuldade na realização da receita e do resultado final.
- Gerar/guardar/consultar código de desconto.
- Gerar/guardar/consultar agenda semanal do utilizador.
- Gerar publicidade.
- Calcular a localização dos supermercados mais próximos (grupo SONAE) do Cliente.
- Calcular o número de utilizadores registados.

Requisitos de Utilizador:

- Utilizador Cliente:
- Registar-se na aplicação.
- Autenticar-se na aplicação.
- O utilizador pode pesquisar por receitas, com filtros ou sem filtros, onde é apresentada uma lista de receitas respetivamente.
- O utilizador realiza receitas com e sem assistente de voz.
- Gerar automaticamente uma agenda semanal de receitas consoante as suas preferências pessoais.
- Pedir a localização de supermercados mais próximos do grupo SONAE para obter ingredientes em falta.
- Obter códigos de desconto, para ser utilizado nas lojas SONAE.

Requisitos de Administrador:

- Verificar o número de Utilizadores registados.
- Verificar avaliações gerais das receitas.
- Verificar número de receitas feitas por utilizador
- Apresentar e gerir publicidades.

2.3. Modelo Domínio

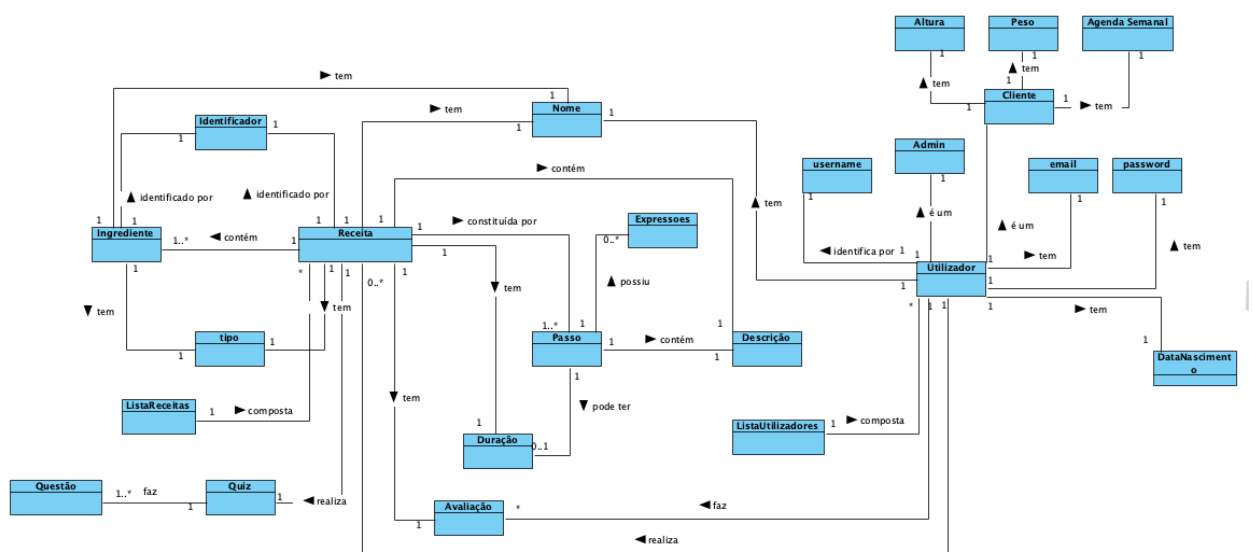


Figura 3: Modelo Domínio, visão geral.

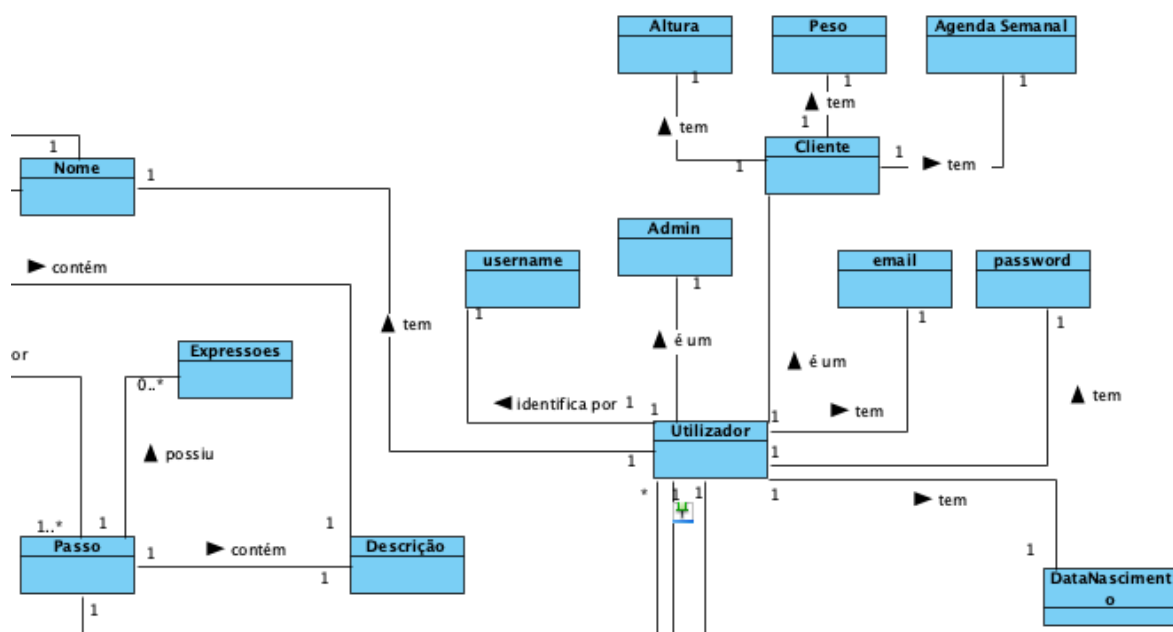


Figura 4: Modelo Domínio, parte do Utilizador.

A aplicação CHELI4 vai conter utilizadores, que podem ser de dois tipos: clientes, que utilizam a aplicação para realizar receitas e os administradores que irão gerir a mesma, tal como se pode ver na figura acima. A informação necessária para o **Utilizador** é o **nome**, **username**, **password**, **email**, **data nascimento**.

O cliente tem ainda três dados importantes, dois deles para estatística, que são a **altura** e o **peso**, e a **agenda semanal**, que é uma funcionalidade da aplicação, onde é gerada uma semana com receitas escolhidas pela aplicação. O utilizador **Admin**, não tem qualquer campo adicional, a sua classe apenas serve para distinguir do **Cliente**. Todos os campos referenciados anteriormente têm multiplicidade de 1 para 1.

Por fim existe uma lista de utilizadores, que tem multiplicidade de um para muitos.

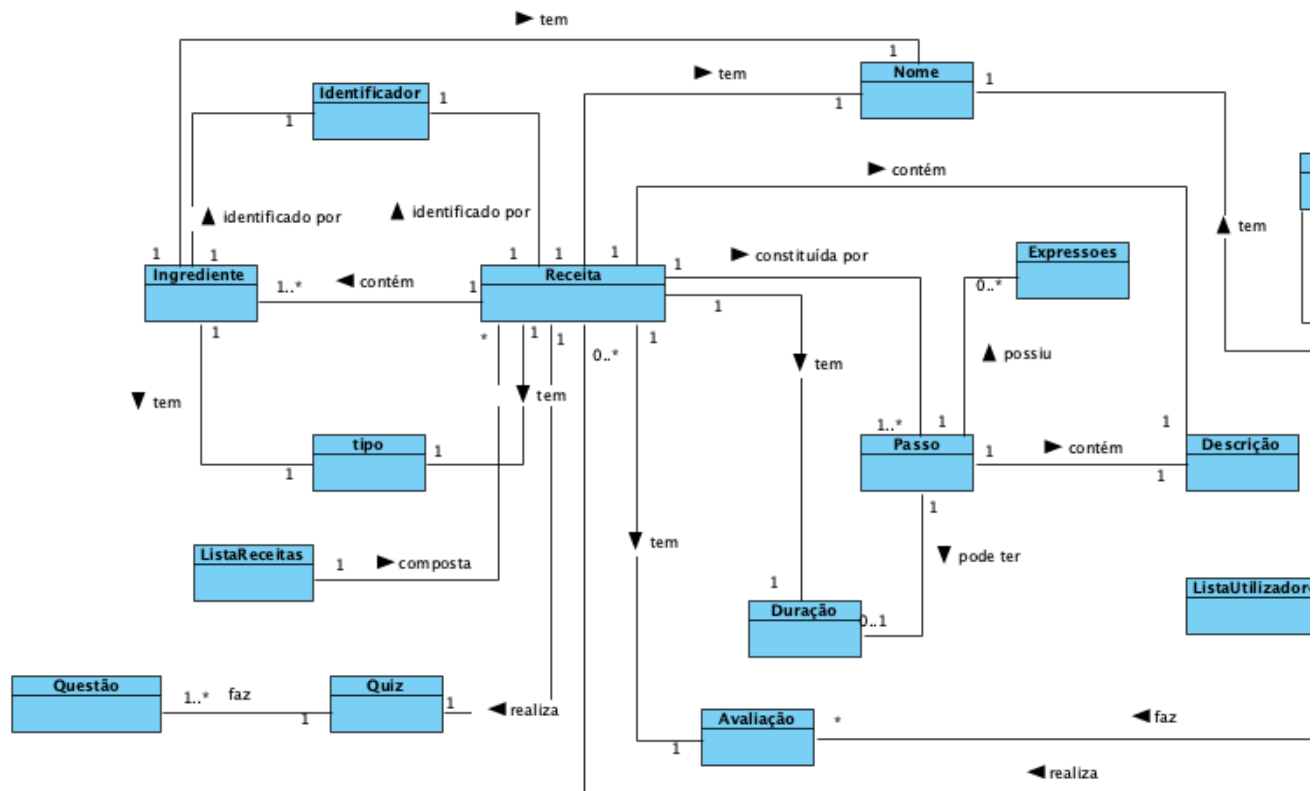


Figura 5: Modelo Domínio, parte da Receita.

A receita contém os campos: **nome**, **tipo**, **identificador**, **duração** e **descrição**. Do mesmo modo, contém um ou mais **passos**, que explicam a receita, sendo que este contém **duração**, **descrição** e **expressões** (que podem ser explicadas, em caso de dúvida por parte do **Cliente**).

Do mesmo modo, a receita pode conter um ou vários **ingredientes**, sendo que estes contém **identificador**, **nome**, **tipo**.

No final de cada receita é realiza um Quiz/questionário, que coloca questões ao Cliente sobre a dificuldade e qualidade da receita.

2.4. Uses cases especificados

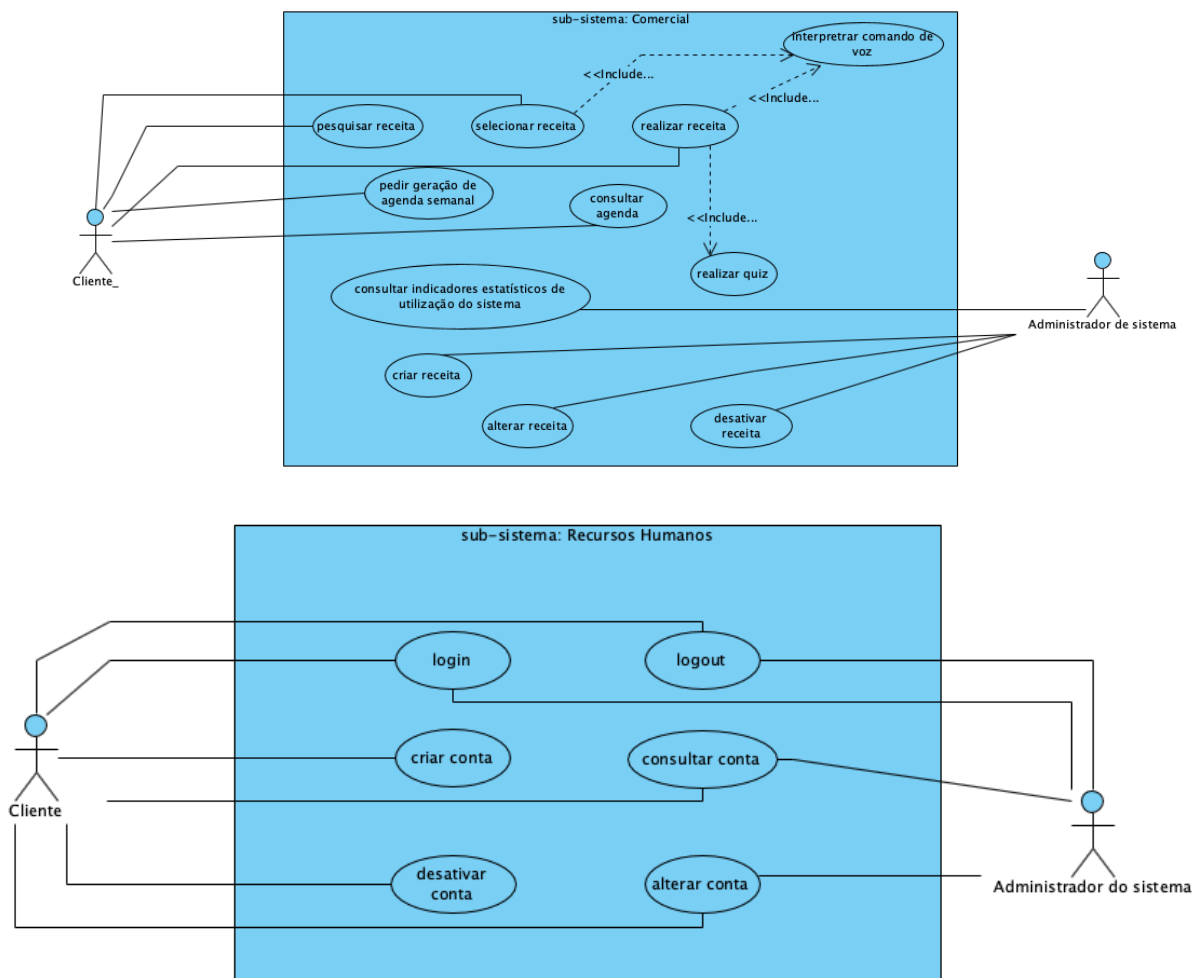


Figura 6 : Diagrama de use cases.

O diagrama de use cases foi dividido por subsistemas, **comercial** e **recursos humanos**. Deste modo, em relação ao subsistema comercial, o ator **Cliente**, que corresponde à pessoa que irá utilizar a aplicação, pode pesquisar, selecionar e realizar receitas, sendo que os dois últimos podem incluir o use case, **interpretador de comandos**, pois podem usar o assistente pessoal. Do mesmo modo, o use case realizar receita também inclui o **realizar quiz**, que consiste num questionário realizado no final da elaboração da receita.

Como um dos requisitos para a elaboração da aplicação é a geração de uma agenda semanal de forma automática, também foi necessário definir os use cases, denominados por **pedir geração de agenda semanal** e **consultar agenda**.

O administrador do sistema, é a pessoa responsável por gerir a aplicação, deste modo, os seus use cases são relacionados com **receitas**, **criar/editar/desativar** e consultar estatística. O use case remover receita não foi elaborado, pois a equipa achou importante não permitir essa funcionalidade, pois não se deve remover dados, apenas marca-se a receita como desativada. Do mesmo modo, no subsistema recursos humanos, também não se permite ao cliente **remover**

a sua conta, apenas **desativá-la**. O cliente, pode se **registrar**, **autenticar**, **consultar a sua conta**, tal como o administrador do sistema.

2.4.1. Pesquisar receita

Formato tabular

| | | |
|-------------------------------------------------|-------------------------------------|-------------------------------------------------------------------------|
| Use Case: | Procurar receitas com filtro | |
| Actor: | Cliente | |
| Pré condição: | Estar autenticado no sistema | |
| Pós condição: | accede a receita pretendida | |
| Cenário Normal | Actor input | System input |
| | | 1-Apresenta campo de filtros |
| | 2-Escolhe filtros pretendidos | |
| | 3- seleciona pesquisa | |
| | | 4-Realiza pesquisa com os filtros 5-Apresenta resultados de pesquisa |
| Alternativa 1 [não tem filtros] (passo 2) | 6-Seleciona receita pretendida | |
| | | 3.1 Obtêm preferências do cliente |
| | | 3.2- volta a 3 |
| Exceção 1 [não tem resultados] (passo 5) | 5.1-Não encontra receita pretendida | |
| | 5.2-Encerra processo. | |

Figura 7 - Especificação tabular do use case procurar receitas com filtro.

Neste use case após a apresentação dos filtros disponíveis, o cliente pode optar por escolher ele os filtros ou não escolher nenhum, caso não o faça, o sistema carrega as preferências deste, consoante as notas atribuídas às refeições por ele confeccionadas. Estas preferências servem de guia para o tipo de comida favorita do cliente, tornando assim a experiência de procurar uma comida que o satisfaça mais personalizada.

Caso este se tenha acabado de registrar e não tenha preferências registadas o sistema fornece uma lista de receitas variadas para que o cliente tenha diversas escolhas possíveis.

2.4.2. Interpretar comando de voz

| | | |
|--------------------------------------------------------------------------------------------------------------------|--------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Use Case: | Interpretar o comando de voz | |
| Actor: | Cliente | |
| Pré condição: | Ator está autenticado no sistema | |
| Pós condição: | Interpretou o comando de voz | |
| Cenário Normal | Actor input | System response |
| | 2. ator diz "Hey Cheli". | 1. sistema está à escuta de uma expressão. |
| | | 3. sistema valida expressão escutada verificando se corresponde à expressão "Hey Cheli". |
| | | 4. sistema escuta próximo comando de voz. |
| | 5. ator diz comando de voz. | 6. sistema interpreta comando de voz. |
| Exceção 1 [expressão não corresponde a "Hey Cheli"] (passo 3) | | 3.1. Encerra processo. |
| Alternativa 1 [ator não se encontra no último passo e comando reconhecido é "Go Next"] (passo 6) | | 6.1. o passo seguinte é apresentado no ecrã. 6.2. o passo seguinte é ditado em voz alta. 6.3. Encerra processo. |
| Alternativa 2 [ator não se encontra no passo 1 e comando reconhecido é "Go Next"] (passo 6) | | 6.1. o passo anterior é apresentado no ecrã. 6.2. o passo anterior é ditado em voz alta. |
| Alternativa 3 [comando reconhecido é "Repeat"] (passo 6) | | 6.1. o passo atual é ditado em voz alta. 6.2. Encerra processo. |
| Alternativa 4 [comando reconhecido é "Help"] (passo 2) | | 6.1. é apresentada a lista de comandos por voz. 6.2. a lista de comandos por voz é ditada em voz alta. 6.3. Encerra processo. |
| Alternativa 5 [ator encontra-se na seleção da receita - passo 0 - e comando reconhecido é "Start"] (passo 6) | | 6.1. o primeiro passo é apresentado no ecrã. 6.2. o primeiro passo é ditado em voz alta. 6.3. Encerra processo. |
| Alternativa 6 [comando reconhecido é "Tell me expressions"] (passo 6) | 6.3. ator diz o número da expressão. | 6.1. sistema diz: "From the following expressions point me the number which you did not understand: 1 - ... 2 - ... N - ...". 6.2. sistema fica à escuta do número da expressão. 6.4. sistema interpreta ditado do utilizador verificando se corresponde a um número (e caso seja, se corresponde a uma expressão). 6.5. <<include>> pesquisa expressão 6.6. Encerra processo. |
| Alternativa 6.1 [sistema não reconheceu o número da expressão] (passo 6.4) | | 6.4.1. sistema informa que não reconheceu o a expressão ditada. 6.4.2. Encerra processo. |
| Alternativa 7 [sistema não reconheceu o comando de voz] (passo 6) | | 6.1. sistema informa que não reconheceu o comando de voz. 6.2. Encerra processo. |

Figura 8: Formato tabular do use case - interpretar comando de voz

Este use case descreve a interação entre o utilizador e o assistente pessoal.

Para ativar o assistente pessoal o utilizador **tem de dizer “Hey Cheli” de seguida diz o comando pretendido**. Tanto a expressão ativadora como o comando pretendido são imediatamente **validados** por parte do sistema (**passos 3 e 6 respetivamente**), confirmando se realmente o utilizador disse “Hey Cheli” e se o comando consta na lista de comandos possíveis.

Consoante o comando ditado é realizado a tarefa associada evidenciada no formato tabular acima. O utilizador pode pedir ao assistente pessoal para:

- **avançar e ditar passo seguinte em voz alta** (Alternativa 1);
- **recuar e ditar passo anterior em voz alta** (Alternativa 2);
- **ditar passo atual em voz alta** (Alternativa 3);
- **Informar quais os comandos de voz existentes** (Alternativa 4);
- **Iniciar a realização de uma receita** (Alternativa 5);
- **pesquisar possíveis expressões não entendidas** (Alternativa 6);

Note-se que este use case é incluído nos use case's: selecionar receita e realizar receita uma vez que é possível tomar proveito do assistente pessoal nos mesmos.

A implementação do assistente pessoal será utilizando uma **Thread** uma vez que o assistente pessoal tem de estar num processo à parte e constante de escuta pela expressão ativadora - "Hey Cheli".

2.4.3. Selecionar receita

Formato tabular:

| | | |
|---------------------------------------------------------|----------------------------------------------------|----------------------------------------------------------------------------|
| Use Case: | Selecionar Receita | |
| Actor: | Cliente | |
| Pré condição: | Ter Pesquisado Receita | |
| Pós condição: | Pronto para Iniciar Receita | |
| Cenário Normal | Actor input | System response |
| | | 1. «include» Interpretar Comando de Voz |
| | | 2. Mostra Descrição da Receita e respetivos Ingredientes |
| | 3. Informa que quer Iniciar a Receita | 4. Inicia a Receita |
| Alternativa 1 [Faltam ingredientes] (passo 3) | 3.1. Informa que faltam Ingredientes | 3.2. Obtém localização do Utilizador e calcula supermercados mais próximos |
| | | 3.3. Mostra supermercados mais próximos |
| | 3.4. Informa que pode prosseguir com a Receita | |
| | | 3.5. Volta a 4. |
| Exceção 1 [Não quer Receita] (passo 3 ou 3.4) | 3.1/ 3.4.1 Informa que não quer realizar Receita | 3.2/ 3.4.2. Cancela Realização de Receita |
| | | |
| Exceção 2 [Não fornece Localização] (passo 3.2.2) | 3.2.2.1. Informa que não quer fornecer Localização | 3.2.2.2. Cancela Realização de Receita |
| | | |
| Alternativa 2 [Não tem GPS] (passo 3.2) | | 3.2.1. Questiona o Utilizador qual a sua Localização |
| | 3.2.2. Fornece Localização | 3.2.3. Volta a 3.3. |

Figura 9: Formato Tabular do use case Selecionar Receita

No use case Selecionar Receita o cliente já realizou a sua pesquisa e no momento em que clica na receita por ele escolhida inicia este use case, onde é inicializado os comandos por voz para poder iniciar o seu cozinheiro sem voltar a clicar no seu dispositivo.

Depois de iniciado o interpretador de comando de voz é mostrado ao cliente os ingredientes da receita, a duração e uma imagem ilustrativa da mesma. Caso tenha todos os

ingredientes o utilizador pode inicializar a sua receita. No caso de a receita levar demasiado tempo a preparar ou não ser bem o que o cliente procurava este pode cancelar a seleção voltando para o menu de pesquisa (**Exceção 1**). Caso seja a receita procurada pelo cliente e este não tenha todos os ingredientes pode indicá-lo e ser-lhe-ão mostrado os 5 supermercados mais próximos do grupo SONAE se for possível obter a sua localização (**Alternativa 1**), neste momento o cliente se não tiver tempo de ir comprar os ingredientes cancela a execução da receita e volta para o menu de pesquisa (**Exceção 1**). Sendo impossível a obtenção da sua localização é pedido ao cliente que forneça a sua localização para poderem ser mostrado os supermercados. Neste momento o cliente pode recusar-se a fornecer a sua localização e como tal volta para o menu de pesquisa (**Exceção 2**), caso forneça a localização são lhe mostrados os supermercados para depois poder prosseguir com a sua receita (**Alternativa 2**).

2.4.4. Realizar receita

| | | |
|----------------------------------------------------------------------------------------------------------|--------------------------------|----------------------------------------------|
| Use Case: | Realizar receita | |
| Actor: | Cliente | |
| Pré condição: | Ator selecionou receita | |
| Pós condição: | Realizou a receita com sucesso | |
| Cenário Normal | Actor input | System response |
| | | 1. <<include>> interpretar comando de voz |
| | | 2. Passo 1 é apresentado no ecrã. |
| Alternativa 1 [ator não se encontra no último passo e carrega no botão "Avançar"] (passo 3) | 3. Ator carrega num botão. | 3.1. o passo seguinte é apresentado no ecrã. |
| | | 3.2. o processo é encerrado. |
| | | |
| Alternativa 2 [ator não se encontra no passo 1 e carrega no botão "Recuar"] (passo 3) | | 3.1. o passo anterior é apresentado no ecrã. |
| | | 3.2. o processo é encerrado. |
| | | |
| Alternativa 3 [ator encontra-se no último passo e carrega no botão "Finish"] (passo 3) | | 3.1. <<include>> quiz |
| | | 3.2. regressa ao menu principal. |
| | | |
| Alternativa 4 [ator carrega no botão numa expressão para pesquisa] (passo 3) | | 3.1. <<include>> pesquisa expressão |
| | | 3.2. o processo é encerrado. |
| | | |

Figura 10: Formato tabular do use case Realizar Receita

Este use case sucede-se ao use case: pesquisar receita e selecionar receita.

Como referido anteriormente o use case: **interpretar comando de voz** é evocado logo no início uma vez que durante a realização da receita o utilizador pode tomar proveito do assistente pessoal, sendo extramente útil uma vez que **pode não ser conveniente** ao utilizador manusear o dispositivo durante a realização da mesma.

Por outro lado, o utilizador pode optar por manipular a aplicação de forma tradicional, clicando fisicamente no respetivo botão para:

- **avançar passo** (Alternativa 1);
- **recuar passo** (Alternativa 2);
- **indicar a conclusão da receita** (Alternativa 3);
- **pesquisar possíveis expressões não entendidas** (Alternativa 4);

Após a indicação da conclusão da receita é apresentado ao utilizador um questionário sendo evocado o use case: quiz; referenciado mais á frente.

2.4.5. Quiz - Questionário

| | | |
|--------------------------------------------------------|------------------------|------------------------------------------------------|
| Use Case: | Quiz | |
| Actor: | Cliente | |
| Pré condição: | Autenticado no sistema | |
| Pós condição: | Responder ao Quiz | |
| Cenário Normal | Actor input | System response |
| | | 1. Recolher questões para efetuar |
| | | 2. Coloca questão, onde se avalia de 0-10 |
| | 3. Indica pontuação | |
| | | 4. Validar pontuação |
| | | 5. Registar resposta |
| Alternativa 1 (passo 3) [não é a última questão] | | 6. Terminar processo |
| | | 3.1. Validar pontuação |
| | | 3.2. Registar resposta |
| Alternativa 2 (passo 3.1) [pontuação inválida] | | 3.3. Regressar a 2 (próxima questão) |
| | | |
| Alternativa 3 (passo 4) [pontuação inválida] | | 3.1.1. Informa que a pontuação inserida não é válida |
| | | 3.1.2. Regressa a 3 (volta a pedir pontuação) |
| | | 4.1. Informa que a pontuação inserida não é válida |
| | | 4.2. Regressa a 3 (para a mesma questão) |

Figura 11: Formato Tabular do use case Quiz

O use case Quiz, é responsável por apresentar um questionário ao cliente, após a conclusão da receita, com questões relacionadas com a dificuldade encontrada e avaliação do resultado final, sendo que as respostas são através de um número entre o intervalo de zero a dez, onde zero corresponde a muito mau, e dez a muito bom.

As questões colocadas ao cliente podem variar, isto é, estas vão ser guardadas na base de dados, podendo a administração atualizar as mesmas, não sendo questões estáticas. Assim, a especificação deste use case, começa com a recolha das questões para uma lista. De seguida é colocada a questão ao cliente, recolhida da lista, e este responde com um número de zero a dez. Uma alternativa para o passo três, é a **Alternativa 1**, que ocorre sempre que não se chegou à última questão, ou seja, ao fim da lista de questões.

A alternativa, começa com a validação da resposta dada pelo cliente, pois este pode não introduzir um número entre o intervalo (0-10), ou nem sequer responder com um número. Caso a resposta dada seja inválida, passa-se para a **Alternativa 2**, que informa que a resposta é inválida e regressa ao passo 3, para voltar a responder, e insistir até a resposta ser válida, sendo que é importante referenciar que ele regressa ao passo 3, para não ser recolhida outra questão enquanto ele não responde corretamente a esta questão. Caso a resposta dada seja válida, regista-se a mesma e volta-se para o passo 2, para passar à próxima questão.

Por fim, quando se chegar à última questão, ou seja, o último elemento da lista, não se entra na **Alternativa 1**, mas segue-se o **cenário normal**, onde se valida a resposta, e caso esta seja inválida, passa-se à **Alternativa 3**, que é bastante semelhante à 2, que insiste na inserção até a resposta ser válida. Se for válida, regista-se a resposta e termina-se o processo.

2.5. Sistema da base de dados

Depois de todos os requisitos levantados e analisados foi possível realizar o modelo lógico para a base de dados necessária para a App a desenvolver. O modelo desenvolvido pela equipa é o seguinte:

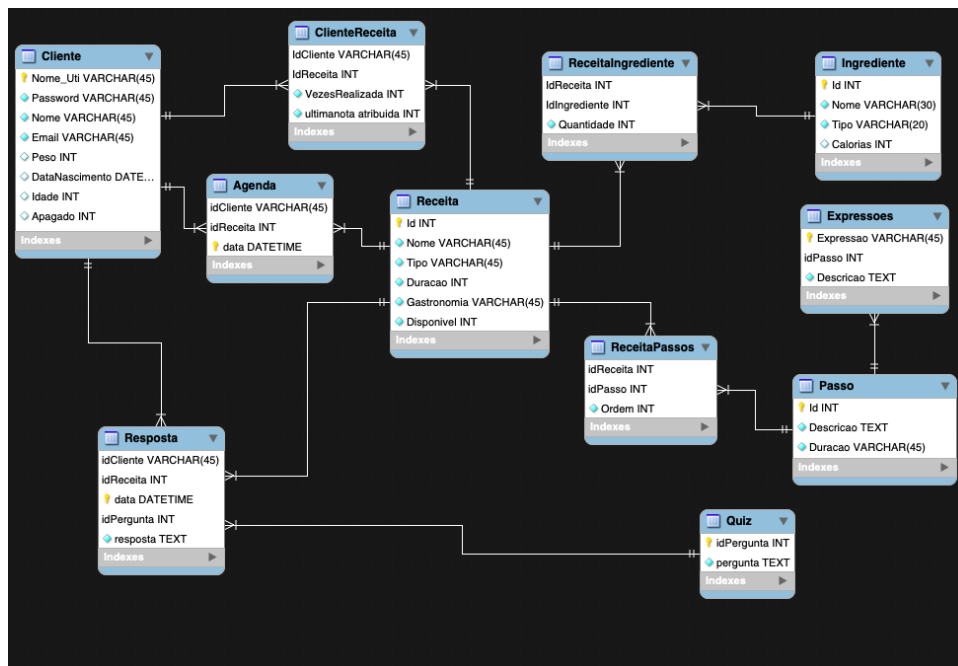


Figura 12: Modelo Lógico

Com este modelo é possível responder a todos os requisitos referidos pelo grupo SONAE para os dados necessários guardar bem como as relações entre os mesmos.

2.6. Sistema da interface gráfica

Nesta secção mostramos as interfaces gráficas mais importantes da App a realizar, as restantes, também elas importantes, mas não tanto serão mostradas em anexo.

Optou-se por uma interface simples e prática, para não complicar o processo da procura e realização da receita e visto que a simplicidade reina na atualidade, sendo o principal objetivo, agradar ao cliente.

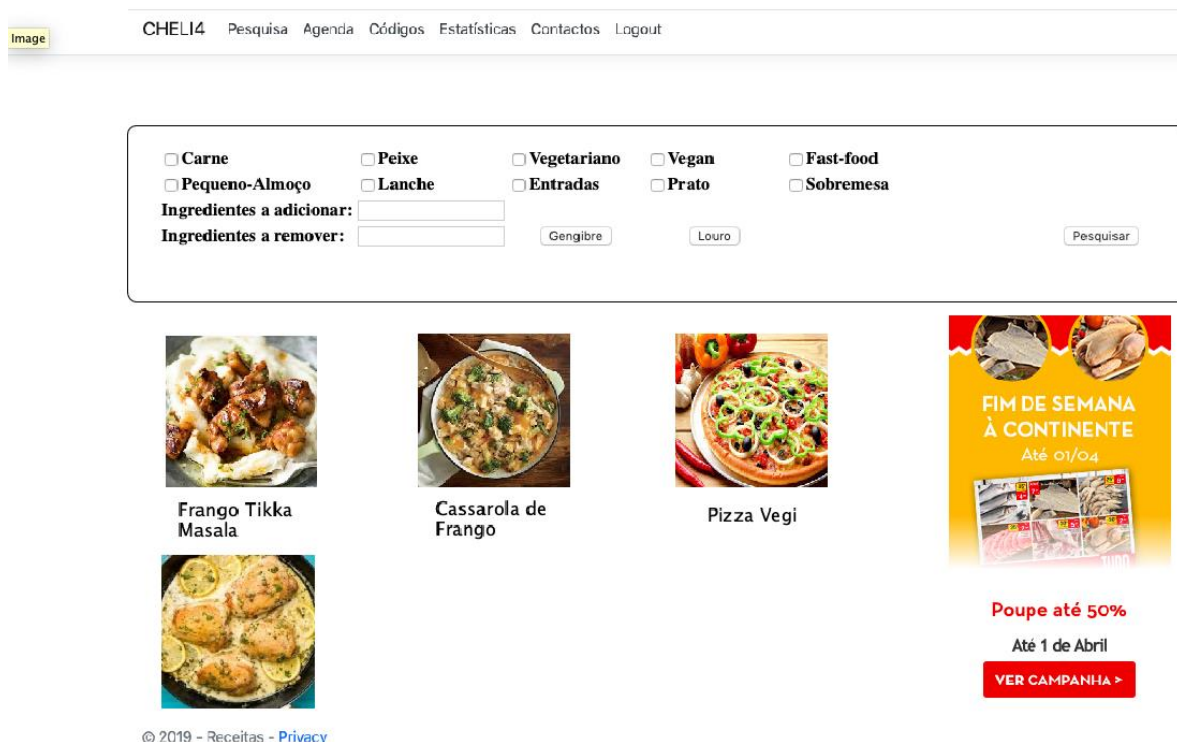


Figura 13: Interface Pesquisar Receita (interface principal)

A interface gráfica pesquisar receita é considerada a principal, pois é a que ocorre quando o Cliente se autentica na aplicação. Esta interface contém os filtros possíveis para a pesquisa de uma receita e o resultado dessa pesquisa aparece abaixo, tal como se pode verificar na figura acima.

Os filtros são variados e automáticos, ou seja, os mesmos são recolhidos da base de dados, portanto podem ser atualizados facilmente pela administração.

Esta interface ainda conta com a opção de procurar receitas que contenham um determinado produto, ou pela negativa, procurar receitas que não tenham um determinado produto, como se pode ver acima, procurou-se por receitas, que não tenham os ingredientes gengibre e louro.

Ainda nesta interface, pode-se aceder aos botões presentes na barra superior que contém outras funcionalidades, como estatística, agenda, códigos de desconto do utilizador, contactos e fazer logout da conta.

Por fim, é importante realçar a presença de publicidades à direita da interface.



Preparação: 25 min; Tempo de cozedura: 1hour 15 minutes; Pronto em: 1 hour 40 minutes;

Ingredientes:

1 frango;
3 limões;
2 laranjas;
1 colher de sopa pimentão doce;
1 colher de sopa sal;
1 colher de sopa azeite;

Faltam Ingredientes

Instruções para usar a CHELI:

HEY CHELI - Comando de reconhecimento de instruções;
START - Começa a ler passo atual da receita;
NEXT - Avança para o próximo passo da receita;
BACK - recuar para o passo anterior;

Start



Figura 14: Interface Selecionar Receita

A interface Selecionar Receita aparece depois de o cliente clicar na receita pretendida na interface Pesquisar Receita, aqui o cliente pode iniciar a receita passando para a interface Realizar Receita, ou pode indicar que lhe faltam ingredientes e nesse caso a aplicação irá sugerir supermercados do grupo SONAE para os comprar, ou mesmo ir para outras interfaces como mostrar os seus códigos promocionais ou ver as suas estatísticas pessoais.



Pré-aquecer o forno a 190° C.

Anterior

Próximo



Poupe até 50%

Até 1 de Abril

VER CAMPANHA >



Marca da Semana

Até 31 de Dezembro

VER CAMPANHA >

Figura 15: Interface Realizar Receita

É nesta interface que acontece grande parte da interação entre o cliente e a App, pois é nesta interface que é possível ter a maior interação hands off com a mesma, aqui o cliente pode ir avançando nos passos da receita clicando no botão “Próximo” ou comunicando com a Cheli4 dizendo “Hey Cheli” seguido de “Next”, ou mesmo voltar para o passo anterior dizendo “Previous”.

Agenda de Alimentação CHELI4:

| Maio 2019 | | | | | | | |
|-----------|----|----|----|----|----|----|----|
| Nº | Se | Te | Qu | Qu | Se | Sá | Do |
| 18 | | | 1 | 2 | 3 | 4 | 5 |
| 19 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| 20 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
| 21 | 20 | 21 | 22 | 23 | 24 | 25 | 26 |
| 22 | 27 | 28 | 29 | 30 | 31 | | |

Escolha a semana em que pretende criar a agenda:

- ☒ Semana 1
- ☐ Semana 2
- ☐ Semana 3
- ☐ Semana 4
- ☐ Semana 5

Mês Anterior

Próximo Mês

Criar Agenda

Figura 16: Interface Agenda Semanal.

Esta é a interface na qual o cliente pode pedir para lhe ser gerado uma agenda semanal com receitas variadas baseadas nas suas preferências. Sendo possível escolher qual a semana para a qual quer gerar a sua agenda.

2.7. Diagrama de estados

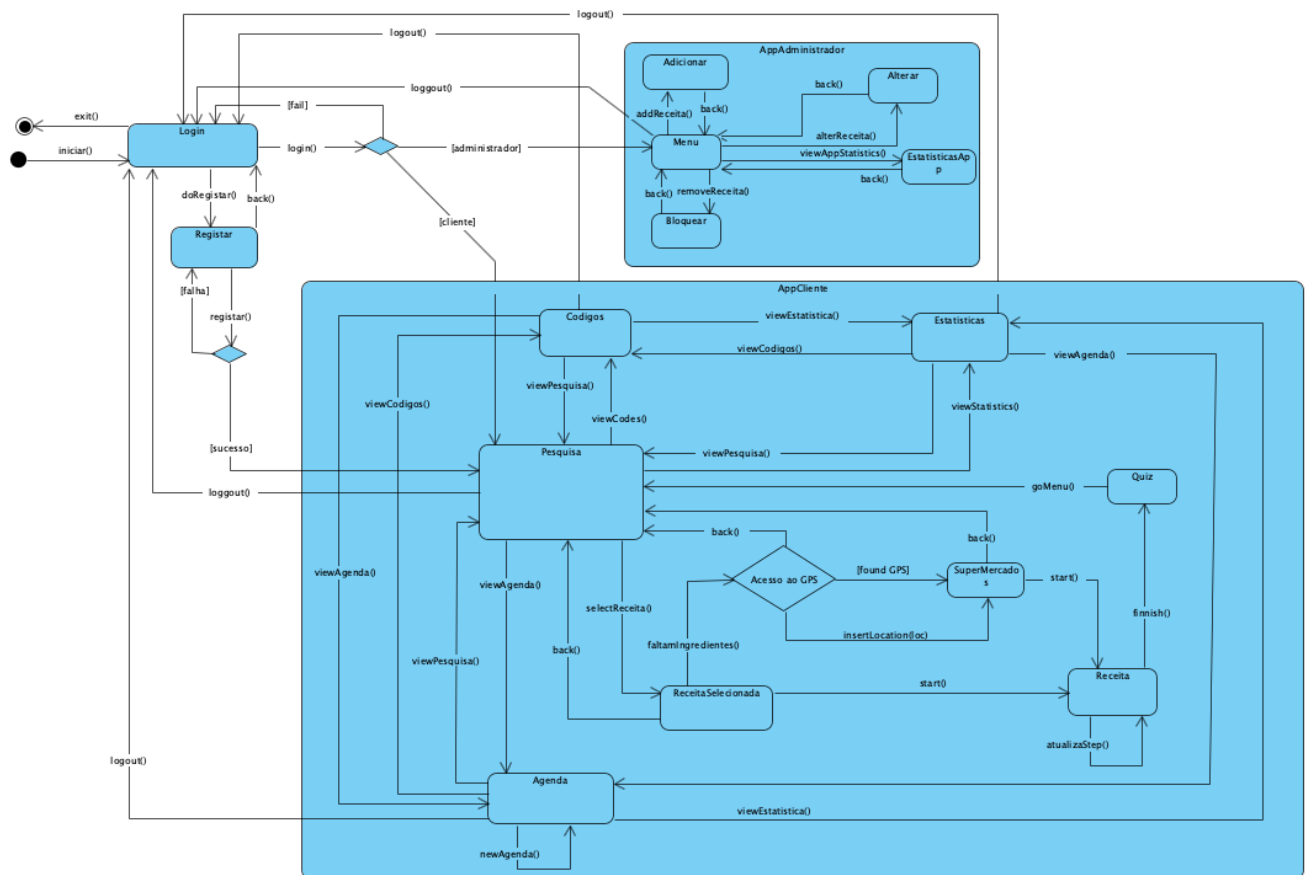


Figura 17: Diagrama de estado.

Quando inicializada a aplicação o primeiro estado é o **Login**, de seguida é possível autenticar havendo dois casos alternativos, ator ser Cliente ou Administrador. Caso seja administrador é encaminhado para o super-estado **AppAdministrador**, caso contrário será para o **AppCliente**.

O cliente pode registrar-se no sistema, permanecendo no estado **Registrar**. Caso tenha sucesso no registo é encaminhado para o super-estado **AppCliente**.

Esse estado é inicializado no **Pesquisa**, sendo possível aceder aos estados: **Códigos**, **Estatísticas**, **Agenda** e recuar a qualquer momento.

Caso selecione uma receita previamente pesquisada, encontra-se no estado **ReceitaSelecionada**, onde pode inicializar a receita passando para o estado **Receita**. Por outro lado, o cliente pode não ter os ingredientes suficientes para iniciar a receita assim, o sistema para calcular os supermercados mais próximos do cliente sendo necessário acesso ao GPS. Caso não seja possível aceder ao GPS é pedido ao cliente para introduzir a sua localização, no entanto se a mesma não seja possível de se obter volta para o estado **Pesquisa**. Caso haja sucesso a fornecer a localização passa para o estado **SuperMercados** sendo possível inicializar a receita

quando o cliente obtiver os ingredientes passando finalmente para o estado **Receita**. Quando finalizada a receita é efetuado um questionário ao cliente passando para o estado **Quiz**. Finalizando o questionário volta novamente para o estado **Pesquisa**.

3. Implementação

3.1. Servidor de base de dados

A implementação da base de dados foi elaborada com a utilização da ferramenta *SQL Server*, na qual aplicou-se o modelo lógico apresentado anteriormente.

Apesar de a equipa se ter baseado no modelo lógico apresentado inicialmente, *foram* feitas algumas alterações face às necessidades que foram surgindo, das quais como guardar o nome das imagens das receitas na tabela **Receita**.

A razão pela qual foi necessário guardar o nome das imagens, foi porque a equipa pensou em tirar proveito do **protocolo HTML**, o qual puxa todas as referências necessárias de ficheiros como imagens, entre outros, sendo que apenas necessitou-se de guardar o nome dos ficheiros das mesmas.

As conexões à base de dados são realizadas e asseguradas pela própria aplicação com um package da Microsoft (*EntityFrameworkCore*), caso o nome e tipos das variáveis das classes sejam os mesmos que os atributos da base de dados o processo de manipular e aceder à base de dados torna-se facilitado uma vez que a associação é automática, no entanto caso tenham nomes diferentes também é possível fazer a associação indicando o nome que a variável tem na base de dados.

O povoamento das receitas foi feito com alguma pesquisa em *websites* culinários, sobre essencialmente receitas portuguesas.

3.2. Arquitetura da aplicação

A aplicação foi dividida em três packages, *Models*, *Views* e *Controllers* (*MVC*).

O package *Controllers* é onde se define os diferentes métodos para os comportamentos de uma determinada página ou determinada entidade, sendo estes: *ClienteViewController*, *PesquisarReceitaViewController*, *PreReceitaViewController*, *RealizarReceitaViewController* e *HomeController*.

O *ClienteViewController*, é referente às janelas associadas à entidade utilizador, sendo estas o *Login* e *Register*.

O *HomeController* é responsável pelo ficheiro *index.html* que é inicializado por defeito em qualquer página web, sendo este controlador responsável por verificar se o utilizador já está autenticado, e em resposta a isto, redirecioná-lo para a página correta.

O *PesquisarReceitaViewController* contém os comportamentos, pesquisar uma receita por filtros e por nome, sendo que na primeira, são apresentadas na mesma página as receitas que fazem *match* com os filtros selecionados pelo utilizador, enquanto que a segunda redireciona para o controlador *PreReceitaViewController*, para apresentar a receita com mais detalhe através do nome inserido.

O *PreReceitaViewController* é responsável por apresentar a informação com mais detalhe da receita, que possa ter sido escolhida na pesquisa por nome ou filtro, do controlador referido anteriormente.

Por fim, o *RealizarReceitaViewController*, tal como o nome sugere, é responsável pelo comportamento da realização da receita, onde pode haver uma interação do utilizador com o assistente pessoal explicada com detalhe mais á frente.

O package *Models*, é subdividido em dois subsistemas que anteriormente foram explicados, **Recursos Humanos** e **Comercial**, onde se definem as classes das diferentes entidades desta aplicação, como o **Cliente**, **Receita**, **Passo**, **Ingrediente**, **Expressao**, etc, e as classes das relações entre entidades, relações estas que vêm da base de dados (N para N e 1 para N).

O package *Views*, está de certa forma ligado ao package *Controllers*, pelo que cada *Controller* tem uma diretoria no package *Views*, com os ficheiros *.cshtml* que controla

Na verdade, a arquitetura da aplicação é composta por uma quarta diretoria ou package, no entanto, não se inclui-o nas anteriores para se manter o modelo MVC.

A definição desta diretoria é *shared*, pelo que, tal como o nome indica, tem as classes *<Entidade>Handling.cs*, as quais são responsáveis por aceder ou manipular os dados no servidor da base de dados, da respetiva entidade, sendo partilhadas por todas as outras classes, onde se necessita dos dados, daí o nome *shared*.

3.3. Interface Gráfica

A nível de interface gráfica, tentou-se ao máximo seguir as interfaces propostas anteriormente, pelo que em determinados casos, achou-se melhor fazer algumas melhorias ou por impossibilidade de elaboração, tornando as mesmas mais simples o que era pretendido.

A elaboração das interfaces gráficas foi feita com a utilização da biblioteca **CSS bootstrap** e com algumas definições **CSS** feitas pelo grupo.

Uma das melhorias efetuadas às interfaces iniciais foi a remoção de publicidades excessivas, pelo que, se tornava saturante a quantidade das mesmas em algumas janelas. Desta forma, atualmente a aplicação apenas tem um rótulo mínimo no final da página, onde apenas se vê caso o cliente faça *scroll*.

3.4. Assistente Pessoal

Um dos requisitos e objetivos desta aplicação era a sua característica *handsfree*, no entanto a equipa concluiu que se precipitou na determinação deste requisito sem ter a certeza de que havia capacidade e conhecimento por parte do grupo para o fazer.

No entanto, não foi possível completar essa funcionalidade, pois não se conseguiu colocar o assistente sempre à escuta visto que a aplicação passa por uma transição entre *Controllers* e *Views*.

A solução ótima seria colocar o assistente sempre à escuta em uma *thread*, no entanto a equipa, experimentou essa possibilidade e não conseguiu obter resultados positivos, pelo que tentou-se solucionar o problema de outra forma.

A forma, para ultrapassar esta dificuldade, foi a colocação de um botão para conseguir falar com o assistente, e deste modo, deixou-se de ter o problema, pois apenas se ativa o assistente quando realmente o utilizador quer falar com o mesmo.

Obviamente a equipa percebe que não é a solução ótima face ao requisito, visto que, não é *handsfree* e que se necessita de uma interação entre o utilizador e o dispositivo, mas foi a única solução funcional que se conseguiu encontrar.

Os comandos inicialmente propostos para interagir com o assistente pessoal eram:

- ~~Hey cheli~~
- Go back
- Go Next
- Repeat
- Help
- ~~Start~~
- Tell me Expressions (alterado)

O comando “Hey Cheli” **não foi implementado** pois, como foi referido anteriormente, o assistente pessoal não está sempre á escuta, logo já não é necessário um comando para ativar o mesmo, sendo o mesmo ativado através de um botão.

O comando “Start” pelo mesmo motivo também **não tem sentido existir**, pois apenas era útil caso o cliente pudesse inicializar a receita através da voz, no entanto como tem de carregar num botão para inicializar o assistente pessoal, seria redundante ter um comando para inicializar a receita, é mais eficiente ter um botão para inicializar diretamente a mesma.

O comando “Tell me Expressions” **foi implementado**, no entanto com uma **alteração** que no ponto de vista da equipa não desvalorizou a eficiência do mesmo apesar de se reconhecer que não ficou cumprido o prometido. O efeito do comando proposto inicialmente era enumerar em voz alta as possíveis expressões que o cliente podia não saber ou compreender, de seguida o cliente indicava o número da expressão e o assistente ditava o significado da mesma. No entanto por motivos que a equipa não chegou a assimilar por falta de tempo para testar a implementação, quando o cliente ditava o número da expressão, o assistente não reconhecia o ditado em maior parte dos casos. Por exemplo, caso o cliente dissesse-se em voz alta “zero” (em inglês) o assistente, por vezes, reconhecia “weather” entre outras palavras que foneticamente não têm qualquer semelhança, sendo o resultado do reconhecimento **não fiável** para ser implementado. **Esta imprecisão confundiu a equipa uma vez que todos os outros comandos são reconhecidos perfeitamente.** Para solucionar tal problema decidiu-se listar os significados de todas as expressões possivelmente não compreendidas em baixo do passo atual.

Para implementar o reconhecimento fiável dos comandos pretendidos foi criada a classe Reconhecimento no package Model.Comercial. O que torna o reconhecimento dos comandos tão **fiável** é o facto de cada **frase ditada pelo utilizador ser filtrada**, sendo procurada na mesma, palavra a palavra, uma que corresponda a um comando.

Note-se que **nem sempre o comando ditado pelo utilizador é reconhecido de forma ideal** ao ponto de corresponder precisamente à expressão esperada, isto é, por vezes o cliente pretende dizer: “Go Next”; e o assistente reconhece: “hey connect” ou “you’re next”, no entanto a equipa desenvolveu o método de reconhecimento e identificação de comandos a englobar tal facto, sendo a frase ditada pelo utilizador testada para várias expressões foneticamente semelhantes, **tornando o reconhecimento bastante mais fiável.**

4. Segurança

Para assegurar que quaisquer dados críticos introduzidos pelo utilizador são transferidos de forma segura entre o dispositivo e o servidor da aplicação, foi criada a classe **Encriptação** no package shared.

A mesma é responsável criar um código Hash utilizando o algoritmo **MD5 checksum** que mapeia qualquer texto num conjunto de 32 caracteres únicos.

Desta forma, por exemplo caso o utilizador no momento de login ou registo introduza a password: "password"; a mesma é mapeada para o seguinte código: "5f4dcc3b5aa765d61d8327deb882cf99"; sendo o mesmo transferido para o servidor e não a password original realmente introduzida pelo cliente **evitando assim que qualquer dispositivo malicioso que se encontre no meio da comunicação fique a conhecer a password original do cliente.**

Note-se que esta implementação foi apenas utilizada para transmitir a password do cliente, no entanto é genérica e está preparada para ser utilizada para proteger qualquer outro tipo de dados críticos.

5. Utilização da Aplicação

Para a utilização da aplicação basta aceder ao link criado www.cheli4.pt sendo mostrada a página de *Login* caso ainda não se tenha autenticado, sendo pedido o *Username* e a *Password*, pois devido à utilização de *cookies* se já tiver realizado o *Login* será remetido automaticamente para a página inicial da aplicação. No caso de ainda não ser utilizador da aplicação pode realizar um *Registo* inserindo algumas informações pessoais.

Depois de efetuado o *Login* pode-se pesquisar uma receita na página inicial pelo seu nome ou na barra superior pode escolher seleccionar uma receita através de filtros/nome, ver os códigos acumulados pela realização de receitas ou procurar pelos nossos contactos para alguma questão.

Quando uma receita é seleccionada é mostrado as informações da receita como ingredientes, tempo de preparação e os comandos de interação com o assistente pessoal. Depois de iniciar a receita clicando no botão *Start Recipe* pode passar os passos para a frente e para trás utilizando respetivamente os botões *Next* e *Back*, ou pode fazê-lo utilizando o botão *Speak with cheli* em que o assistente fala com o utilizador dizendo este o comando que pretende, o assistente não se limita apenas a esses dois comandos (*Next* e *Back*), reagindo ao comando *Repeat* e o passo da receita é repetido, o comando *Expressions* onde são apresentadas no ecrã as expressões mais complicadas do passo atual como referido anteriormente, este responde ainda ao comando *Help* mostrando um *PopUp* com os comandos reconhecidos pelo assistente.

Terminada a receita é redirecionado para a página inicial novamente clicando no botão *Finish* ou na barra superior.

6. Validação do Projeto

Depois do tempo do projeto terminado foi feita uma validação junto do cliente, sendo apresentados os tópicos concluídos até então bem como os não concluídos para uma fase futura com a possibilidade de acrescentar ainda novas funcionalidades.

Apesar de nem todos os tópicos predefinidos terem sido concluídos o nosso cliente *SONAE* mostrou-se bastante satisfeito com o trabalho realizado pois foram implementadas as funcionalidades mais importantes para a aplicação, como o assistente pessoal e a possibilidade de realização das receitas.

Conclusão

No final da análise e levantamento de requisitos foi possível à equipa avaliar que com o tempo disponível para a realização do trabalho não seria possível implementar todas as funcionalidades possíveis, tendo então sido implementadas apenas as funcionalidades principais para o correto funcionamento da aplicação.

Depois de implementadas as funcionalidades principais foram deixadas para trabalho futuro as seguintes funcionalidades:

- Verificar bateria do dispositivo para avisar que poderá não ter bateria suficiente para realizar a receita em causa;
- Incluir receitas dos utilizadores para depois ser colocada ao dispor de todos os utilizadores;
- Fazer compras online através dos ingredientes em falta;
- Adicionar mais comandos de voz para tornar a App completamente hands off, com exceção do Login que tem de ser manual;
- Colocar um avatar para uma maior interação com o utilizador.
- Colocar mais idiomas, como espanhol, mandarim, entre outras, para uma maior adesão no exterior do país, colocando também em português para uma maior adesão nacional.

A equipa de trabalho acha que a especificação dos requisitos, do sistema de base de dados e o sistema de interfaces foram bem explícitos, sendo no futuro realizados novos com as novas funcionalidades a implementar.

Contudo, tal como foi referido anteriormente, a especificação e definição dos use cases, não foi seguida ao pormenor na fase de implementação, pois quando foram realizados o grupo não fez uma pesquisa do funcionamento da linguagem C# e framework ASP.NET, pelo que, certas funcionalidades não foram aplicadas, devido ao desconhecimento por parte da equipa.

No entanto, não foram totalmente descartadas as decisões tomadas na definição dos use cases, pelo contrário, foi bastante útil, essa estruturação inicial.

Do mesmo modo, os tempos definidos no diagrama de gantt não foram cumpridos, pelo que a distribuição dos tempos não foi a mais correta.

O resultado final foi validado pelo cliente, apesar de não terem sido cumpridos alguns dos requisitos e objetivos inicialmente propostos e discutidos com o mesmo, os representantes da SONAE afirmam que os componentes e funcionalidades principais estão presentes no resultado final, pelo que estão satisfeitos.

Por fim, o importante a reter deste projeto é a pré-avaliação das nossas capacidades/conhecimentos das ferramentas que nos fornecem e a importância de uma correta e completa estruturação de um projeto antes de qualquer implementação, para que não haja

surpresas mais adiante e seja tarde demais, pois neste projeto pode-se dar a volta, mas na indústria isso poderá não ser possível.

Lista de Siglas e Acrónimos

App Aplicação

Anexos

I Anexos - Diagramas de subsistemas

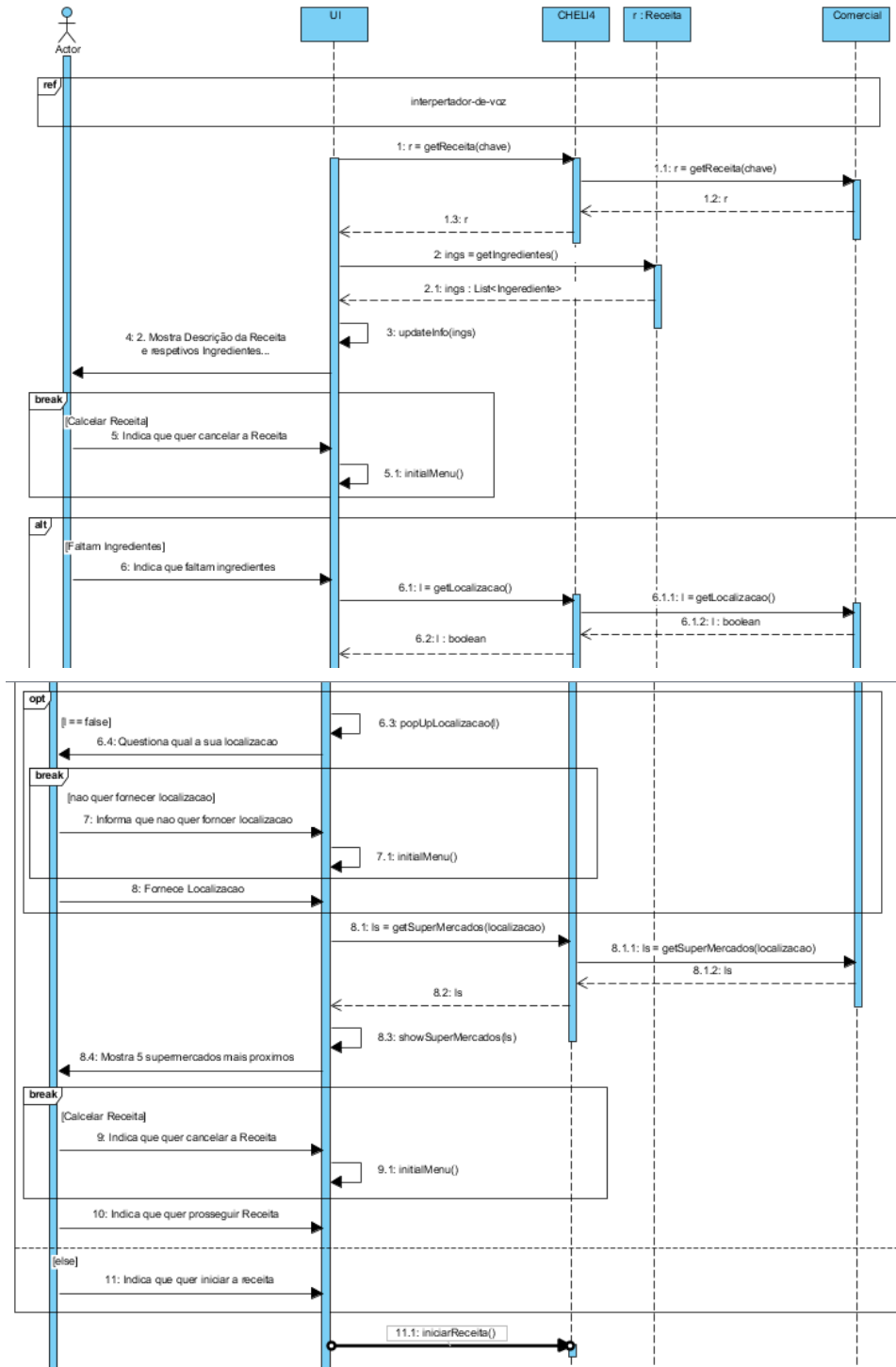


Figura 18: Diagrama de Subsistemas do use case Selecionar Receita

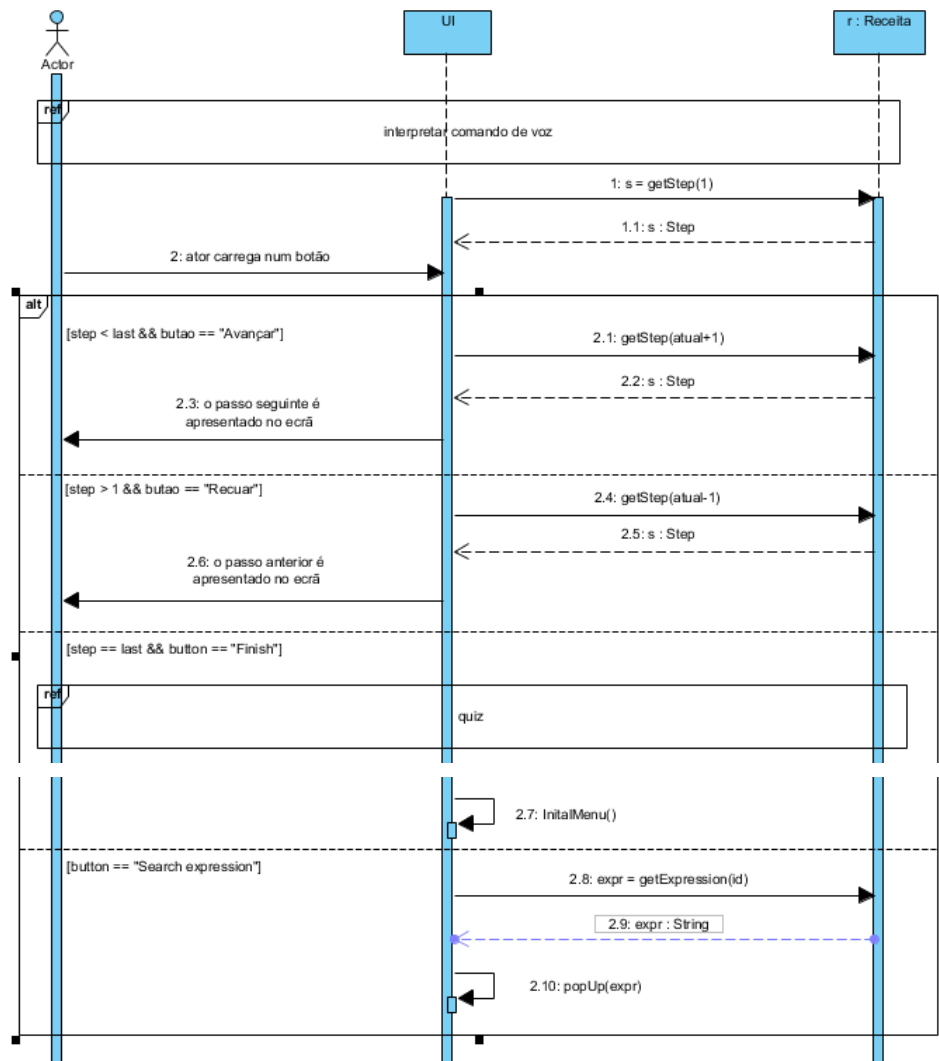


Figura 19: Diagrama de subsistemas do use case Realizar Receita

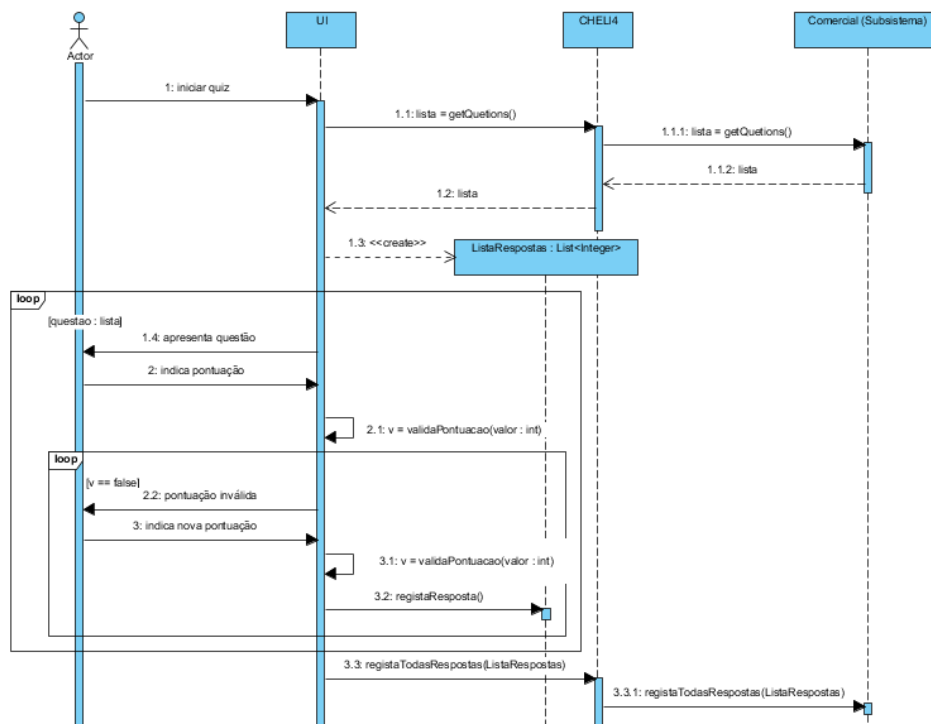


Figura 20: Diagrama de subsistemas do use case Quiz

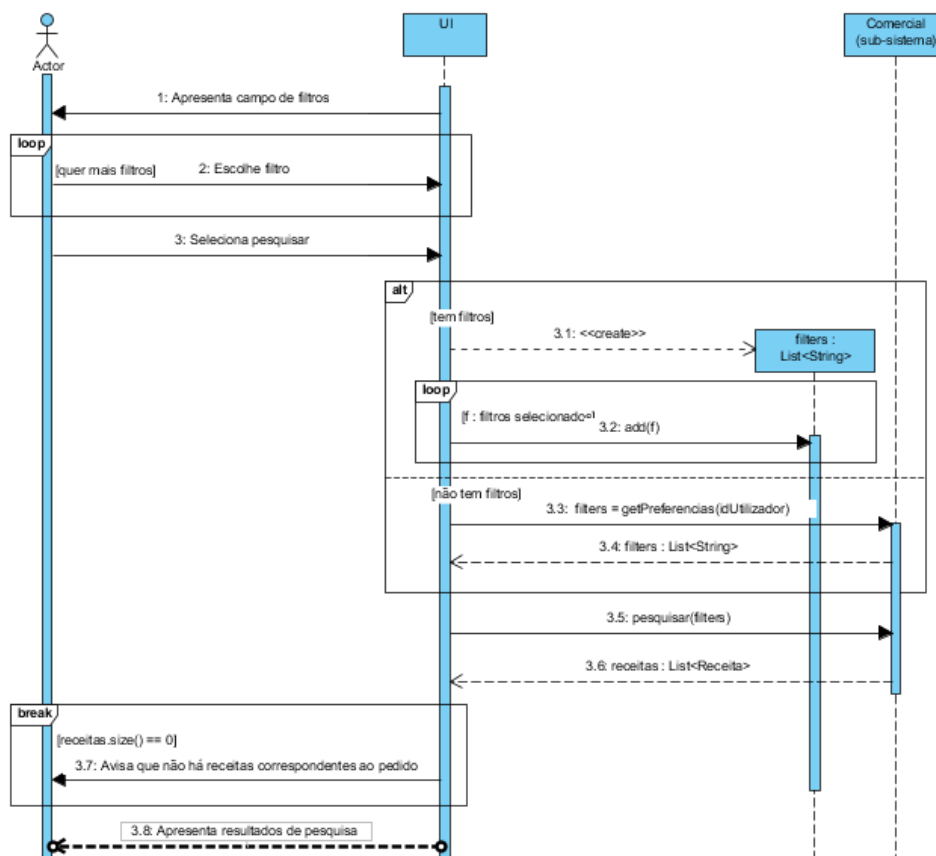
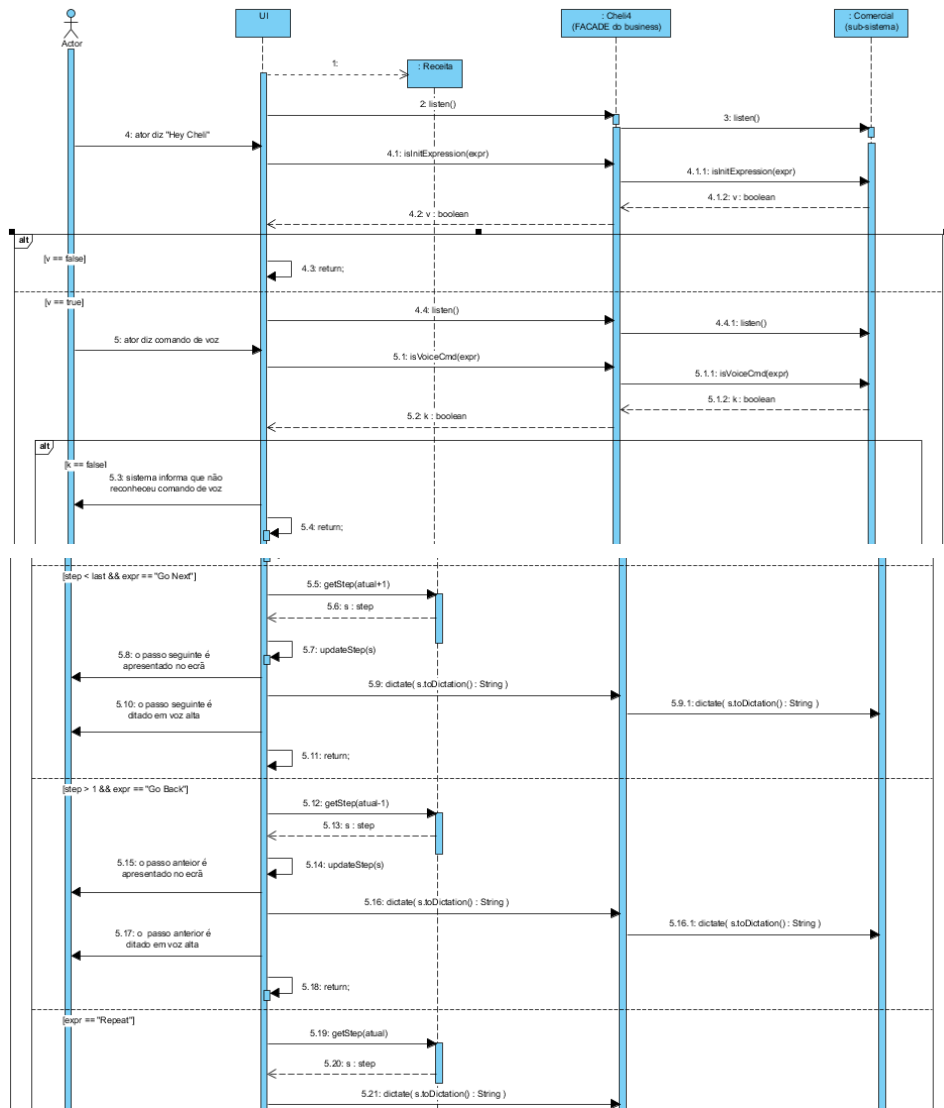


Figura 21: Diagrama de subsistemas do use case Pesquisar Receita



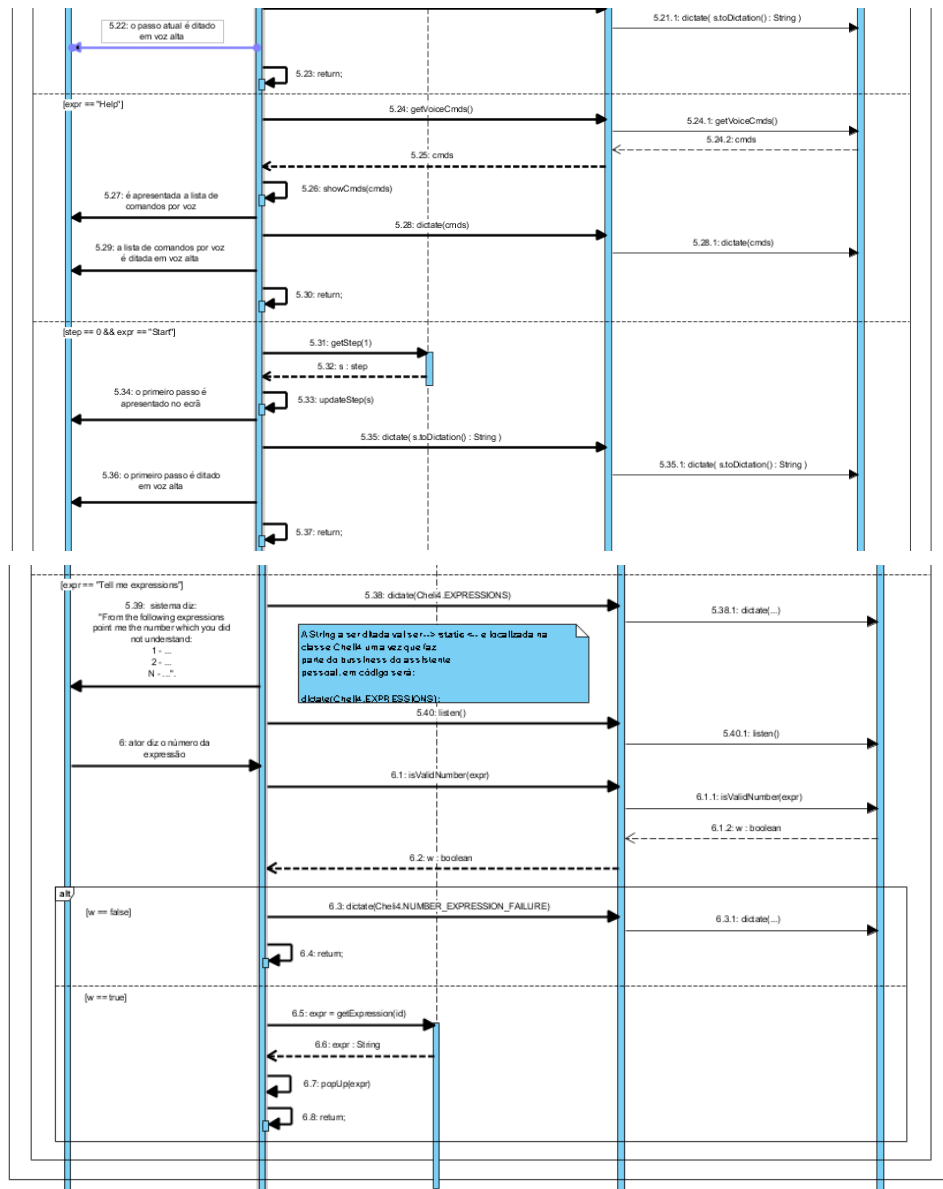


Figura 22: Diagrama de subsistemas do use case Interpretar Comando de Voz