

**Instituto Superior de Engenharia do Porto**  
Departamento de Engenharia Electrotécnica  
Rua Dr. António Bernardino de Almeida, 431, 4249-015 Porto

# **Sistema de Gestão de Clínica**

Mestrado em Engenharia Eletrotécnica e de Computadores

**Alexandre Amorim [1161497]**

Supervisor: Prof. Maria Benedita Campos Neves Malheiro

Ano Letivo: 2020-2021



---

# Resumo

---

O presente relatório tem como finalidade documentar o desenvolvimento de uma aplicação que tem o objetivo de automatizar os serviços de uma clínica de saúde. Por sua vez, estes serviços dividem-se no gestão de consultas, de disponibilidades, utilizadores, entre outros.

As tecnologias que foram empregues para a realização deste trabalho foram o *SQL Server Management Studio* (SSMS) para administração da Base de Dados (BD), framework *ASP.NET Core*, o *Object-Relational Mapping* (ORM) *Dapper*, a Interface de Programação de Aplicações (API) de gestão de eventos Calendário Google e o *Wkhtmltopdf* para emitir os Formato de Documento Portátil (PDF) de faturaçāo.

**Palavras-Chave:** Clinica de Saúde, Calendário Google, Gmail, SSMS, Web Apps.



---

# Índice

---

<b>Índice</b>	<b>i</b>
<b>Índice de Figuras</b>	<b>iii</b>
<b>Índice de Tabelas</b>	<b>v</b>
<b>Acrónimos e Abreviaturas</b>	<b>vii</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Enquadramento e Objetivos . . . . .	1
1.2 Motivação e Metodologia . . . . .	2
1.3 Estrutura . . . . .	3
<b>2 Ambiente de Desenvolvimento</b>	<b>5</b>
2.1 Servidor de Base de Dados . . . . .	6
2.2 Servidor de Aplicações . . . . .	6
2.3 ASP.NET Core . . . . .	6
2.4 Dapper . . . . .	7
2.5 API Calendário Google . . . . .	8
2.6 Wkhtmltopdf . . . . .	8
2.7 Sumário . . . . .	8
<b>3 Aplicação Web</b>	<b>9</b>
3.1 Descrição . . . . .	9
3.2 Proposta de solução . . . . .	10
3.2.1 Casos de Uso . . . . .	11
3.2.1.1 Paciente . . . . .	11
3.2.1.2 Médico . . . . .	12
3.2.1.3 Administrador . . . . .	13
3.2.2 Modelo da Base de Dados . . . . .	14

3.3	Funcionalidades . . . . .	15
3.3.1	Autenticação . . . . .	15
3.3.2	Papel de Paciente . . . . .	16
3.3.3	Papel de Médico . . . . .	20
3.3.4	Papel de Administrador . . . . .	21
3.3.5	Funcionalidades Adicionais . . . . .	23
3.4	Sumário . . . . .	27
<b>4</b>	<b>Avaliação da Aplicação em Ambiente Produtivo</b>	<b>29</b>
4.1	Testes Funcionais . . . . .	30
4.2	Testes Não Funcionais . . . . .	32
4.3	Sumário . . . . .	34
<b>5</b>	<b>Conclusão</b>	<b>35</b>
	<b>Referências Bibliográficas</b>	<b>37</b>
<b>A</b>	<b>Instalação e Configuração do Ambiente de Desenvolvimento</b>	<b>39</b>
A.1	Servidor de Base de Dados . . . . .	39
A.2	Dapper . . . . .	43
A.3	Google Calendar API . . . . .	44
A.4	Wkhtmltopdf . . . . .	45
A.5	ASP.NET MVC Core . . . . .	46
<b>B</b>	<b>Complementos dos Testes Não-Funcionais</b>	<b>49</b>
B.1	Testes de Carga e de Tempo de Resposta . . . . .	49
B.2	Testes de Usabilidade . . . . .	50

---

# Índice de Figuras

---

2.1	Tecnologias envolvidas na aplicação . . . . .	5
2.2	Arquitectura de desenvolvimento Web ASP.NET MVC Core . . . . .	7
2.3	Processo de mapeamento entre BD e linguagens de programação . . . . .	7
2.4	Funcionalidades da API Calendário Google . . . . .	8
3.1	Logótipo da RINTECare Porto . . . . .	9
3.2	Caso de Uso de Paciente . . . . .	11
3.3	Caso de Uso de Médico . . . . .	12
3.4	Caso de Uso de Administrador . . . . .	13
3.5	Diagrama da BD da aplicação da RINTECare Porto . . . . .	14
3.6	Processo de registo . . . . .	15
3.7	Autenticação de utilizador . . . . .	16
3.8	Email de redefinição de palavra-passe . . . . .	16
3.9	Vista de paciente . . . . .	17
3.10	Perfil do Paciente . . . . .	17
3.11	Processo de marcação de consulta . . . . .	18
3.12	Notificação de agendamento de consulta . . . . .	19
3.13	Histórico de consultas . . . . .	20
3.14	Procura de pacientes . . . . .	20
3.15	Gestão de disponibilidade . . . . .	21
3.16	Painel de administrador . . . . .	22
3.17	Processo de consulta realizada . . . . .	22
3.18	Gestão de Operações . . . . .	23
3.19	Página Casa . . . . .	24
3.20	Esquema da página Sobre Nós . . . . .	25
3.21	Livro de reclamações . . . . .	25
3.22	<i>Blog Health+</i> . . . . .	26
3.23	Resumo da página de contactos . . . . .	26
4.1	Testes funcionais de registo . . . . .	30

4.2	Testes funcionais de autenticação . . . . .	31
4.3	Alteração do estado da consulta . . . . .	31
4.4	Proposta de <i>chatbot</i> para a RINTECare Porto . . . . .	34
A.1	Visão holística de instalação e configuração dos <i>softwares</i> em estudo .	39
A.2	<i>Download</i> e configuração do servidor Linguagem de Consulta Estruturada (SQL) . . . . .	40
A.3	Finalização da configuração do servidor SQL . . . . .	40
A.4	<i>Download</i> e configuração do SSMS . . . . .	41
A.5	Instalação e inicialização do SSMS . . . . .	41
A.6	Inicialização das <i>Stored Procedure</i> (SP) . . . . .	42
A.7	Verificação do nome do sistema . . . . .	42
A.8	Seleção do item instalador <i>Manage NuGet Packages</i> ... . . . . .	43
A.9	Processo de instalação do Dapper . . . . .	43
A.10	Configuração de cliente . . . . .	44
A.11	Instalação das configurações de cliente . . . . .	44
A.12	Processo de Instalação da API Google Calendar . . . . .	45
A.13	Processo de Instalação do Wkhtmltopdf . . . . .	45
A.14	Criação de um novo projeto . . . . .	46
A.15	Configuração do novo projeto e selecionamento do tipo de ASP.NET Core Web App . . . . .	46
A.16	Verificação do projeto . . . . .	47
B.1	Rotas e tempo de trânsito do domínio Livro de Reclamações . . . . .	49
B.2	Rotas e tempo de trânsito do domínio Gmail . . . . .	50
B.3	Rotas e tempo de trânsito do domínio da API Calendário Google . . . . .	50
B.4	Processo de autenticar . . . . .	50
B.5	Processo de alterar palavra-passe . . . . .	51
B.6	Processo de cancelar consulta . . . . .	51
B.7	Processo de consultar perfil . . . . .	52
B.8	Processo de contactar assistentes . . . . .	52
B.9	Processo de editar consulta . . . . .	53
B.10	Processo de marcar uma consulta . . . . .	54
B.11	Processo de registrar . . . . .	54
B.12	Processo de consultar detalhes da consulta . . . . .	55

---

# **Índice de Tabelas**

---

3.1	Estados da consulta . . . . .	20
4.1	Testes Não-Funcionais das Páginas Exteriores . . . . .	32
4.2	Testes Não-Funcionais Utilizadores Externos . . . . .	33
4.3	Teste de Usabilidade . . . . .	33



---

# Acrónimos e Abreviaturas

---

**API** Interface de Programação de Aplicações

**BD** Base de Dados

**CRUD** Atualizar, Criar, Eliminar e Ler

**CSS** *Cascading Style Sheets*

**GUI** Interface Gráfica de Utilizador

**HTML** Linguagem de Marcação de Hipertexto

**HTTPS** Protocolo de Transferência de Hipertexto Seguro

**IIS** *Internet Information Services*

**IP** *Protocolo de Internet*

**LOD** *Linked Open Data*

**MSSQL** *Microsoft SQL Server*

**MVC** Modelo-Vista-Controlador

**ORM** *Object-Relational Mapping*

**PDF** Formato de Documento Portátil

**SP** *Stored Procedure*

**SQL** Linguagem de Consulta Estruturada

**SSMS** *SQL Server Management Studio*

**RINTE** Redes Inteligentes e Serviços

**UC** Unidade Curricular

**UI** Interface de Utilizador

**UML** Linguagem de Modelagem Unificada

**URI** Identificador de Recurso Uniforme

**URL** Localizador Uniforme de Recursos

# **Capítulo 1**

---

## **Introdução**

---

### **1.1 Enquadramento e Objetivos**

O trabalho desenvolvido neste relatório corresponde ao segundo momento de avaliação da Unidade Curricular (UC) de Redes Inteligentes e Serviços (RINTE), que pertence ao plano curricular do mestrado em Engenharia Eletrotécnica e Computadores - Telecomunicações.

A aplicação a conceber destina-se a automatizar os processos de uma unidade de serviços de saúde e permitir o acesso à informação consoante os perfis de utilizadores desenvolvidos. Caracterizada por um funcionamento *user-friendly* e eficaz, a aplicação foi proposta de modo a solucionar e permitir aos utilizadores realizar as seguintes ações:

- Autenticar-se através da página de autenticação, bem como registar-se;
- Cancelar agendamentos de consultas;
- Consultar uma panóplia de informações, tais como: disponibilidades, informações pessoais, marcação de consultas, entre outros;
- Criar novos administradores, disponibilidades, especialidades, médicos, pacientes utilizadores, entre outros;
- Editar informações relativas a cada perfil de utilizador, disponibilidades, consultas, entre outros;
- Emitir *mails* de boas-vindas, de redefinição de palavra-passe, marcação de consulta e da sua respetiva faturaçāo;

- Facilitar a autenticação dos utilizador por intermédio de uma funcionalidade *Remember me?*;
- Gerir administradores, especialidades, médicos, pacientes utilizadores, entre outros;
- Obter informações sobre os contactos da clínica, bem como da caracterização da mesma;
- Solicitar uma consulta de especialidade e o seu respetivo horário;
- Visualizar todas as informações relativas as consultas na API Calendário Google, bem como a gestão das consultas.

Relativamente as tecnologias selecionadas, recorreu-se ao *Microsoft SQL Server* (MSSQL) como motor de BD, ao SSMS como ferramenta de administração de BD com Interface Gráfica de Utilizador (GUI), o *framework* .NET para o desenvolvimento da aplicação, ao *package* ORM Dapper, de modo a mapear os modelos de domínio orientado a objetos para uma base de dados relacional persistente, a API Calendário Google para apresentar e gerir os eventos de calendário relativo as consultas e ao Wkhtmltopdf para a geração de PDF de faturação.

## 1.2 Motivação e Metodologia

Os primeiros sistemas, que utilizavam serviços *Web*, eram designados de *Web sites*, e eram limitados a uma coleção de documentos com conteúdo estático e interconectados via *hyperlinks* [1] (Jazayeri, 2007). Todavia, nos tempos atuais as *Web Apps* proporcionam uma plataforma que pode integrar diferentes tecnologias, tais como as API da Google<sup>1</sup>.

Deste modo, possibilitam funcionalidades adicionais, tais como: e-mails, calendários, serviços de faturação, entre outros [2] (Dissanayake e Dias, 2017). Permitindo que as empresas se tornem mais simples e atingam vários clientes e utilizadores em simultâneo [3] (Singer, 2020). Tendo em conta a sua importância descrita anteriormente e relevância para a UC de RINTE, elaborou-se a aplicação documentada no presente relatório.

A aplicação *Web* desenvolvida segue uma abordagem tradicional de um sistema de engenharia, dispondo de uma estrutura robusta para o desenvolvimento de um projeto desta natureza. Podem ser identificadas diversas etapas dentro desta metodologia, tais como as que se encontram de acordo com a seguinte ordem:

- Análise das metodologias e das tecnologias atuais relevantes para o projeto em causa;

---

<sup>1</sup><https://developers.google.com/apis-explorer>

- Desenvolvimento de um manual de instalação e configuração das tecnologias em causa;
- Implementação, testes, validação e avaliação das soluções desenvolvidas;
- Documentação da solução proposta.

De salientar, que existem constantemente atualizações e novas tecnologias para o desenvolvimento de projetos desta natureza, assim espera-se que em propostas futuras seja possível a incorporação de novas metodologias.

### 1.3 Estrutura

Relativamente a estrutura deste relatório, esta encontra-se essencialmente dividida em cinco capítulos. O Capítulo 1 contém uma breve contextualização do âmbito e dos objetivos do trabalho. Ao longo do Capítulo 2, são descritas as tecnologias aplicadas para o desenvolvimento da *Web App* em estudo, contemplando o servidor de aplicações, da base de dados, o ambiente de desenvolvimento, assim como o *framework*, o ORM, o gestor de calendário e o gerador de PDF de faturaçāo. Posteriormente, no Capítulo 3 encontra-se presente a descrição e caracterização da aplicação desenvolvida. Mais especificamente dos casos de utilizador, do diagrama da BD, da sua implementação e das suas funcionalidades. No Capítulo 4 é descrita a avaliação da aplicação em ambiente de produção, contemplando testes, validação e avaliação da mesma. Ulteriormente, no Capítulo 5 estão presentes as principais conclusões advinientes da elaboração do trabalho, assim como possíveis linhas de investigação a serem exploradas no futuro.



## Capítulo 2

---

# Ambiente de Desenvolvimento

---

Este capítulo introduz os conceitos e funcionalidades das tecnologias aplicadas no desenvolvimento da aplicação em estudo. Inicialmente, irá ser explicado o servidor de BD (Secção 2.1) e de aplicações (Secção 2.2). Seguido, de uma breve descrição do framework ASP.NET Core (Secção 2.3) e do ORM Dapper (Secção 2.4). Finalmente, irão ser caracterizadas as ferramentas do Calendário Google (Secção 2.5) e o do Wkhtmltopdf (2.6), como ilustrado na Figura 2.1.



Figura 2.1: Tecnologias envolvidas na aplicação

## 2.1 Servidor de Base de Dados

O servidor de base de dados optado para a aplicação em causa foi a versão 18.7.1 do SSMS. O SSMS é um ambiente integrado desenvolvido pela Microsoft<sup>2</sup>, que permite gerir, configurar, monitorizar, administrar, criar consultas e *scripts* de bases de dados [4]. Tendo sido principalmente aplicado na construção da base de dados da aplicação e na sua respetiva administração. Os procedimentos de instalação encontram-se descritos e ilustrados no Anexo A.1.

## 2.2 Servidor de Aplicações

Relativamente ao servidor de aplicações, a escolha predileta para servir páginas ou arquivos Linguagem de Marcação de Hipertexto (HTML) solicitados foi o *Internet Information Services* (IIS). Similarmente ao SSMS, o IIS foi desenvolvido pela Microsoft e, aceita solicitações de computadores clientes remotos e retorna a resposta apropriada [5]. A sua principal funcionalidade neste projeto consistiu em compartilhar e fornecer informações em redes locais (LAN), como *intranets* corporativas, e redes de longa distância (WAN), como a Internet.

## 2.3 ASP.NET Core

O *framework* a que se recorreu para desenvolver a aplicação é o ASP.NET Core (v5.0.1 - 2021) da família de soluções Microsoft [6], que segue o *standard* Modelo-Vista-Controlador (MVC). Caracterizado por ser de código aberto e ser o sucessor do ASP.NET, oferece suporte ao desenvolvimento simultâneo de diferentes módulos, detalhados de seguida:

- Controlador: Os controladores gerem as interações do utilizador, trabalham com o modelo e, por fim, selecionam uma vista para renderizar (daí a sua designação - controla como o aplicativo responde a uma determinada solicitação).
- Modelo: O modelo representa o estado do aplicativo e qualquer lógica de negócio ou operações que devem ser realizadas por ele.
- Encaminhamento: Permite mapeamento de Localizador Uniforme de Recursos (URL), criar aplicativos com URL comprehensíveis e pesquisáveis e definir rotas com suporte a restrições, padrões e valores opcionais.
- Vista: Responsável por toda a lógica da UI do aplicativo. Por exemplo, a visualização do Cliente incluirá todos os componentes da Interface de Utilizador (UI), como caixas de texto, menus suspensos etc. com os quais o usuário final interage.

---

<sup>2</sup><https://www.microsoft.com/pt-pt/>

Na Figura 2.2 adaptada de [7], encontra-se representado a arquitetura geral do framework. Inicialmente é realizado um pedido solicitado pelo utilizador através do motor de busca ao MVC (passo 1). De seguida, é analisado pelo componente de Encaminhamento, que irá encaminhar para o controlador apropriado (passo 2). Posteriormente o componente Modelo interage com a BD, de modo a obter as informações necessárias para o processo em execução (passo 3). Ulteriormente, essa mesma informação é escalada até ao componente de Vista (passo 4) que irá renderizar a saída no motor de busca (passo 5). A sua configuração encontra-se no Anexo A.5.

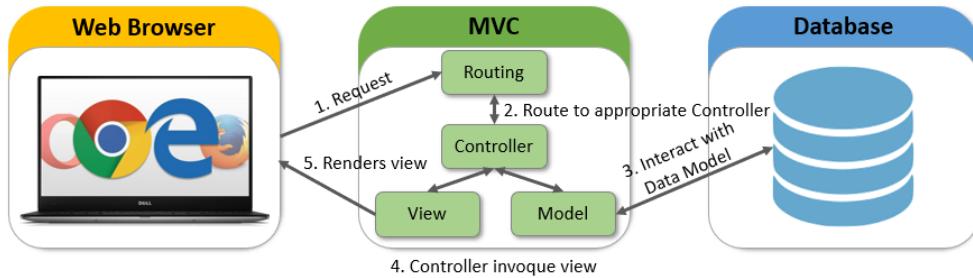


Figura 2.2: Arquitectura de desenvolvimento Web ASP.NET MVC Core

## 2.4 Dapper

O Dapper (v2.0.78 - 2021) consiste num Mapeador Relacional de Objetos (ORM), ou seja, é responsável pelo mapeamento entre BD e as linguagens de programação [8]. Sendo esse o seu propósito para este projeto, permitindo uma maior flexibilidade e agilidade de desenvolvimento da mesma. Adicionalmente, de modo a tornar ainda mais célere e eficiente este processo, recorreu-se ao CodVerter<sup>3</sup> para gerar as entidades de classes em C#. Sendo na Figura 3.6a ilustrado um exemplo desenvolvido para a entidade do paciente, como representado na Figura 3.6b. De salientar, que no Anexo A.2 é possível verificar a sua instalação e configuração.

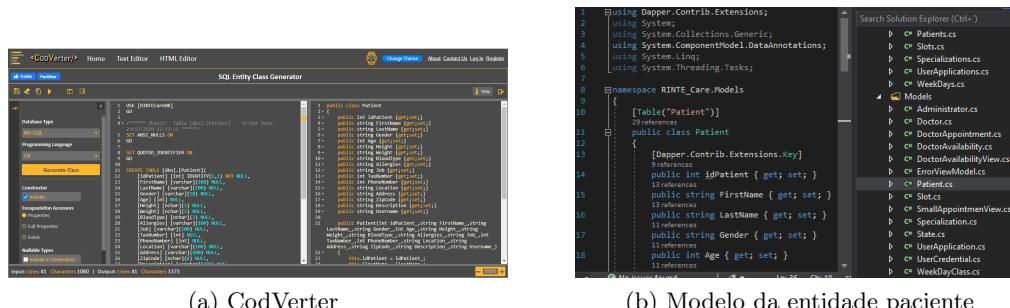


Figura 2.3: Processo de mapeamento entre BD e linguagens de programação

<sup>3</sup><https://codverter.com/src/sqltoclass>

## 2.5 API Calendário Google

A API Calendário Google (v1.49.0.2170 - 2021) permite que as *Web Apps* apresentem e atualizem eventos de calendário na forma de *feeds* de dados [9]. Possibilitando desse modo criar novos eventos, editar ou excluir eventos existentes e consultar eventos, como ilustrado na Figura 2.4. Sendo essas as principais funcionalidades na *Web App* em estudo. No Anexo A.3 encontram-se A sua instalação e configuração.

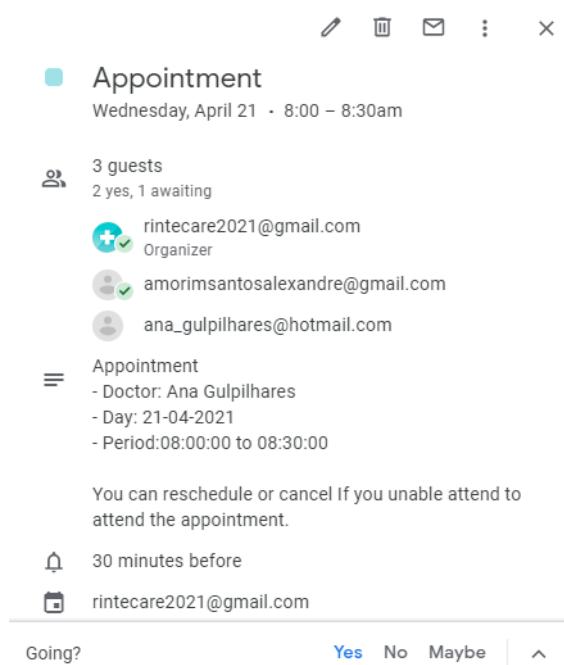


Figura 2.4: Funcionalidades da API Calendário Google

## 2.6 Wkhtmltopdf

Como a sua designação menciona, o Wkhtmltopdf (v3.0.2 - 2021) é uma ferramenta de código aberto desenvolvida em C++, que permite ao utilizador converter HTML (página Web) em um documento PDF [10]. Tendo sido empregue para criar o PDF concernente a faturação das consultas (instalação no Anexo A.4).

## 2.7 Sumário

Neste capítulo foram estabelecidos as bases necessárias para compreender os restantes capítulos. Estes conhecimentos incluem a definição/princípios de funcionamento do servidor de BD e de aplicações, do ASP.NET MVC Core, do Dapper, da API Calendário Google e do Wkhtmltopdf. Sendo as suas respetivas instalações e configurações abordadas no capítulo dos anexos A.

## **Capítulo 3**

---

# **Aplicação Web**

---

*Neste capítulo é demonstrado o processo de desenvolvimento da aplicação. Na Secção 3.1 encontra-se uma breve descrição da aplicação e da entidade. Seguidamente, na Secção 3.2 é descrita a sua respetiva proposta de solução. Finalmente, na Secção 3.3 são apresentadas as diversas funcionalidades desenvolvidas.*

### **3.1 Descrição**

A aplicação desenvolvida tem como missão automatizar os processos de uma clínica de saúde, designada de RINTECare Porto. Os seus principais processos consistem na marcação de consultas, na sua devida faturação, na consulta de informações provenientes da BD, na gestão de utilizadores e do envio de *emails* de boas-vindas, redefinição de palavra-passe. Sendo os três perfis de utilizadores enumerados de seguida: (i) administrador, (ii) médico e (iii) paciente. Ulteriormente, na Figura 3.1 é ilustrado o logótipo da entidade alvo.



Figura 3.1: Logótipo da RINTECare Porto

### 3.2 Proposta de solução

Posteriormente ao levantamento de requisitos e dos objetivos definidos para o projeto, é necessário realizar a modelação do sistema a desenvolver. Como referido por Bill Gates, um dos maiores problemas no desenvolvimento de aplicações Web é a falta de planeamento e arquitetura, que resulta em ineficiências significativas, menor qualidade de código e elevadas derrapagens:

"The common thread for everything I do is this idea of a Web-services architecture. What does that mean? It means taking components of software and systems and having them be self-describing, so that you can aim them, ask them what their capabilities are, and communicate with them using a standard protocol." - Bill Gates

Deste modo, optou-se pelas melhores práticas de desenvolvimento, sendo a linguagem implementada para modelar o projeto em causa a Linguagem de Modelagem Unificada (UML). Esta linguagem foi desenvolvida pela Rational Software em 1994, sendo este um dos principais *standards* atuais para construir, especificar, documentar e visualizar sistemas de informação e os seus componentes intrínsecos:

- Construir: Os modelos UML podem ser conectados diretamente a uma variedade de linguagens de programação através do mapeamento de um modelo de UML para uma linguagem de programação como JAVA ou C#. Permitindo desse modo realizar mecanismos como engenharia reversa.
- Documentar: As entregas de um projeto além da codificação são alguns artefatos, que são essenciais para controlar, medir e comunicar sobre um sistema durante o seu desenvolvimento requisitos, arquitetura, desejo, código-fonte, planos de projeto, testes, libertadores de protótipos, etc ...
- Especificar: Permite construir modelos precisos e inequívocos, abordando as especificações de todo o projeto, que deve ser feito no desenvolvimento e implantação de um sistema de software.
- Visualizar: Através da UML, é possível visualizar o sistema existente e como o sistema estará após a implementação. A menos que pensemos, não podemos implementar. UML ajuda a visualizar como os componentes do sistema se comunicam e interagem com cada de outros.

A UML define diversos diagramas para modelar sistemas informáticos, sendo que para o desenvolvimento da aplicação Web em causa foram implementados os seguintes diagramas: casos de uso e diagrama de base de dados (não contemplado na UML).

### 3.2.1 Casos de Uso

Na UML, o diagrama de caso de uso resume os detalhes dos utilizadores do sistema (também conhecidos como atores) e as suas interações com o sistema. Permitindo deste modo apresentar uma visão holística gráfica das funcionalidades fornecidas por um sistema em termos de atores, casos de uso e quaisquer dependências entre os mesmos. Sendo que na aplicação em estudo se distinguem três tipos de utilizadores: administrador, médico e paciente.

#### 3.2.1.1 Paciente

O primeiro perfil é o paciente, que é um utilizador externo ao sistema, sendo lhe impostas restrições de acesso. Além disso, carece de mais restrições de segurança, dado serem o principal causador de problemas nas aplicações e a "janela" de onde são realizados a maioria dos ataques de segurança informático nesta vertente. Assim, são atribuídos necessariamente menos funcionalidades a este perfil.

Relativamente as suas funcionalidades, seguem o padrão do mercado neste contexto. Na Figura 3.2 é possível observar que lhes é possível alterar e redefinir as suas palavras-passe, em que a redefinição da mesma é enviada por *email*. Ademais, é também lhes proporcionado a permissão de gerir as suas consultas (criar e cancelar) tanto pela aplicação como pela API Calendário Google e consultar o seu próprio perfil.

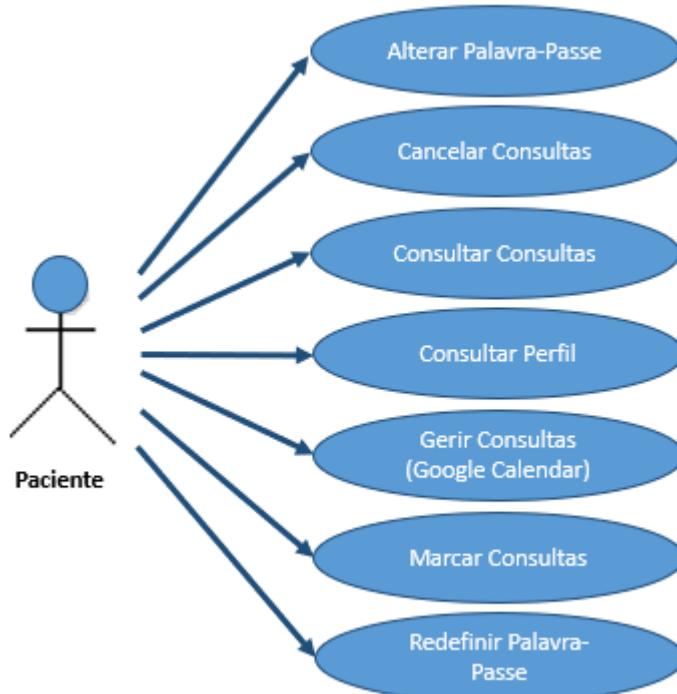


Figura 3.2: Caso de Uso de Paciente

### 3.2.1.2 Médico

O segundo perfil com maior quantidade de funcionalidades é o médico. Assim como o paciente, é um utilizador externo (dado que pode trabalhar em mais que um instituto de saúde) e não tem funções administrativas, sendo que logicamente lhe são impostas restrições de modo a preservar a segurança e privacidade da entidade em questão.

Como no contexto real, o perfil de médico tem como foco a sua própria gestão (disponibilidades, alteração e redefinição de palavra-passe, entre outros) e de consultar informação essencial para as suas consultas (paciente, horário, entre outros), como ilustrado na Figura 3.3.

Adicionalmente, poderá encontrar toda a informação relativa a agendamentos através do seu próprio Calendário Google, o que lhe permite realizar operações Atualizar, Criar, Eliminar e Ler (CRUD), tal como criar, editar e cancelar uma consulta.

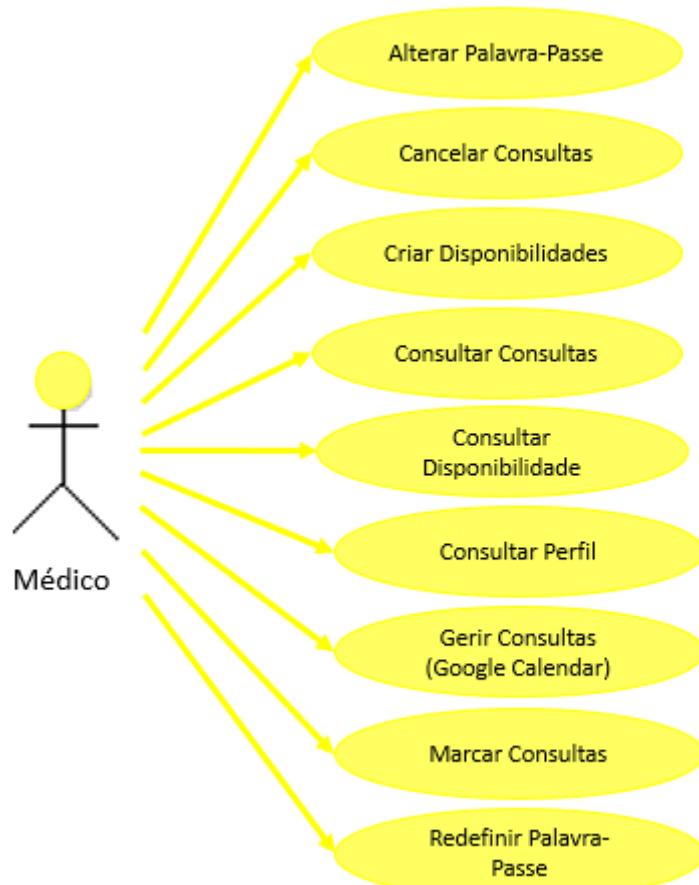


Figura 3.3: Caso de Uso de Médico

### 3.2.1.3 Administrador

Por fim, o perfil de administrador reúne todas as funcionalidades da aplicação. Isto é, para além de ter acesso as funcionalidades dos médicos e pacientes para qualquer utilizador, também tem acesso a funcionalidades adicionais, tais como: consultar e gerir administradores, consultas, consultas por intermédio da API Calendário Google, médicos, especialidades, pacientes e utilizadores, como ilustrado na Figura 3.4.

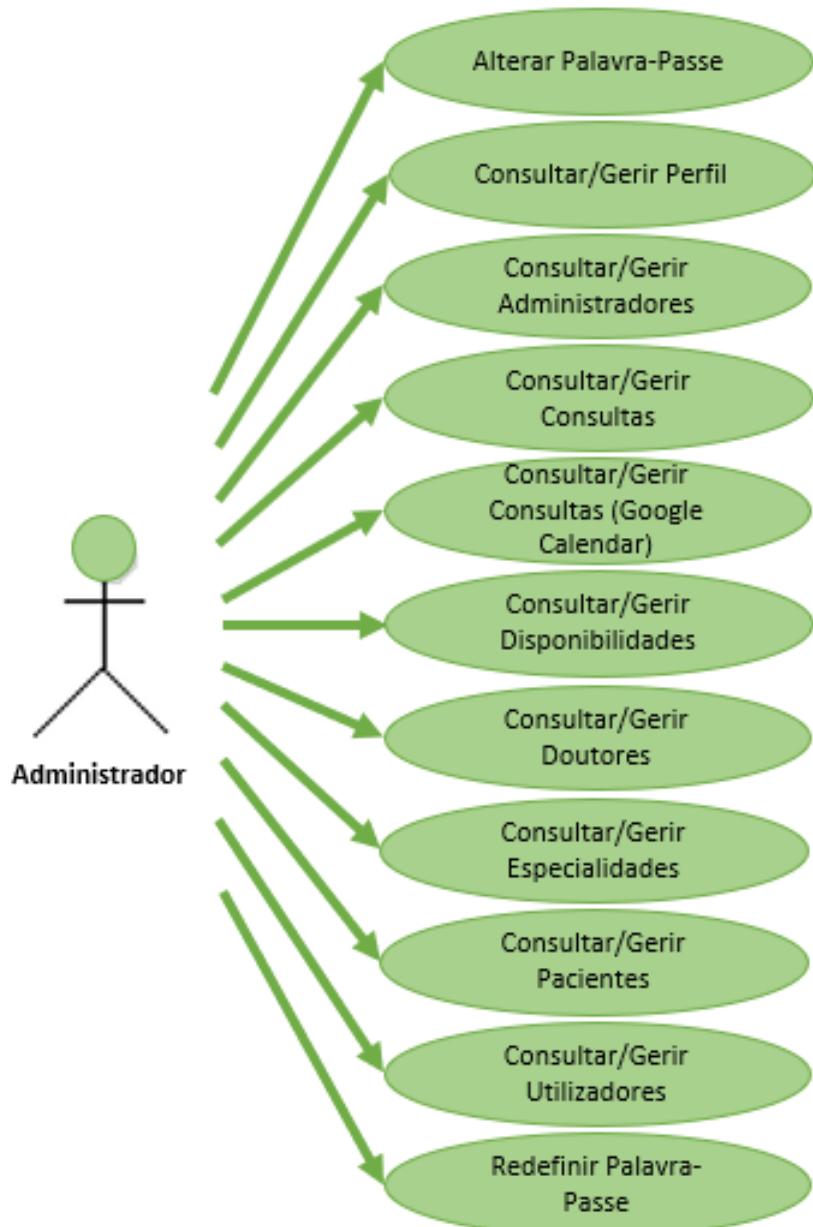


Figura 3.4: Caso de Uso de Administrador

### 3.2.2 Modelo da Base de Dados

Outro componente extremamente importante na arquitetura de uma aplicação é a BD. Uma base de dados consiste numa coleção de dados inter-relacionados armazenados com redundância mínima para servir múltiplos utilizadores rapidamente e eficientemente. A sua modelação é realizada de modo a obter consistência interna de dados, estabilidade máxima e ter o mínimo de redundância. Garantido deste modo a minimização da capacidade necessária para o armazenamento de dados e a sua optimização para atualizações futuras.

A base de dados da aplicação da RINTECare Porto foi desenvolvida através do SSMS, sendo constituída por nove tabelas: (i) Administrador, (ii) ConsultaMédico, (iii) DiasSemana, (iv) DisponibilidadeMédico, (v) Médico, (vi) Especialização, (vii) Paciente, (viii) Slots e (ix) UtilizadorAplicação. Destas nove tabelas, três correspondem a entidades (administrador, médico e paciente), sendo as restantes tabelas de indexação. No diagrama da Figura 3.5 é possível visualizar as suas respetivas ligações, atributos e registos.

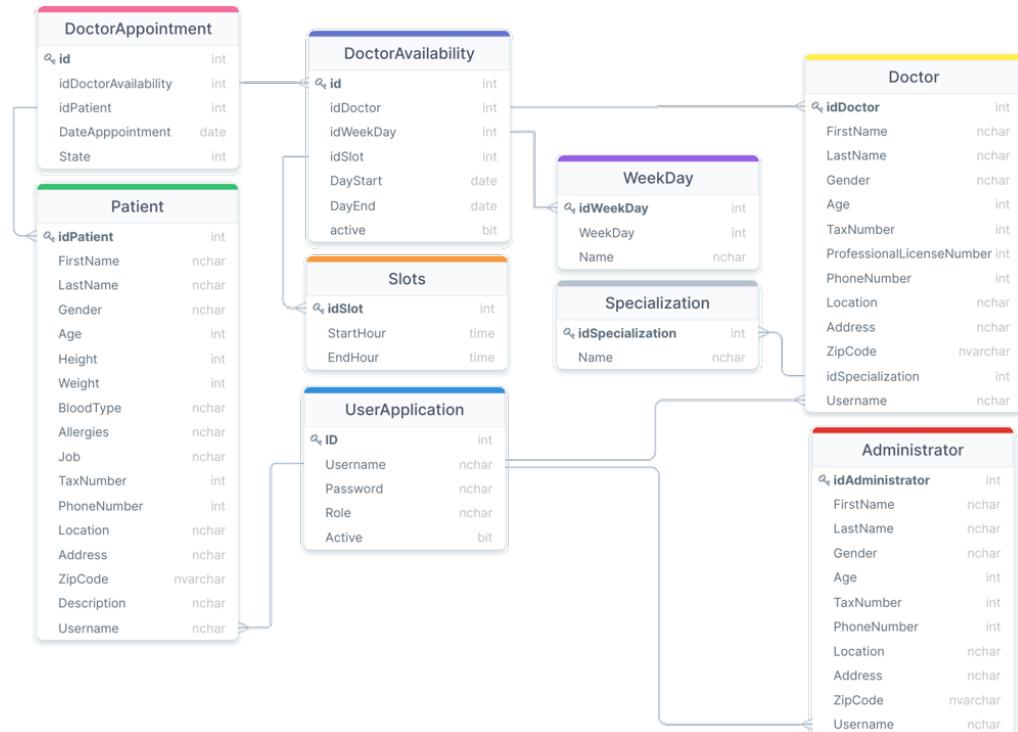


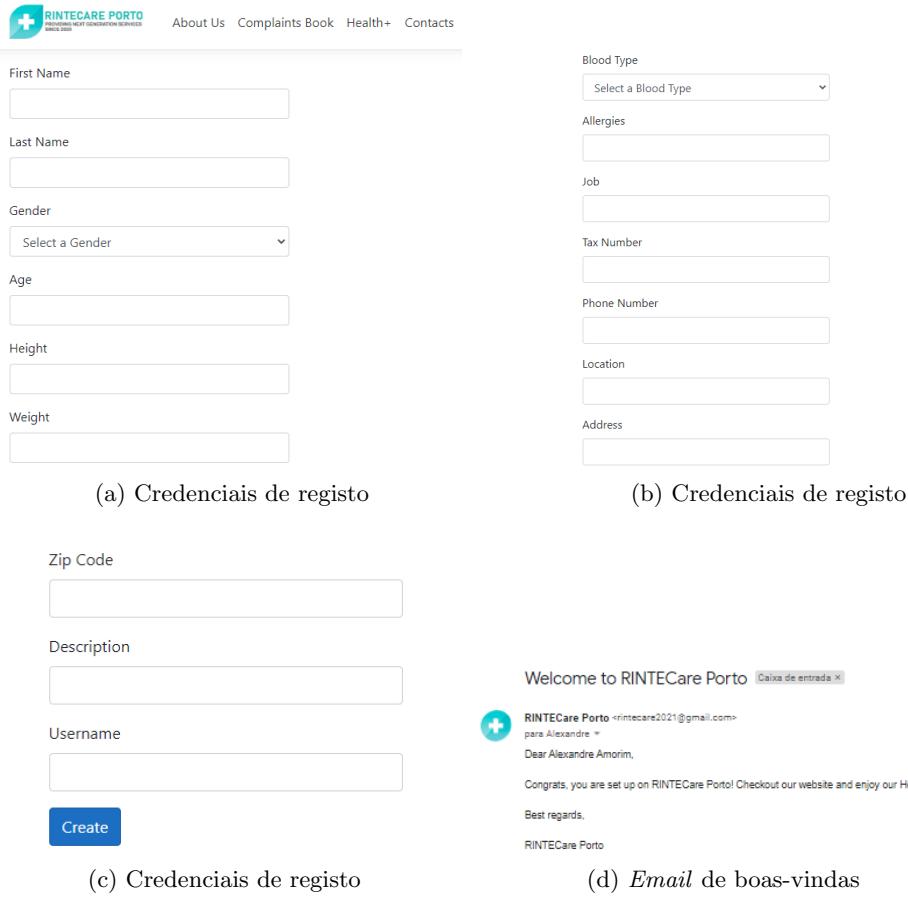
Figura 3.5: Diagrama da BD da aplicação da RINTECare Porto

## 3.3 Funcionalidades

### 3.3.1 Autenticação

Antes de o utilizador se poder autenticar, é necessário que primeiramente já se encontre na BD. Para isso é necessário que se registe, introduzindo os dados consoante o seu perfil (sendo perfil de paciente é o único que se pode registar, devido a boas regras de segurança informática).

Para o efeito basta que o utilizador opte pela opção *Register*, sendo lhe apresentado a interface da Figura 3.6a. Para o utilizador se registar, basta que preencha os campos das Figuras 3.6abc corretamente, caso contrário irá lançar alertas de erros como iremos abordar com mais ênfase no próximo Capítulo 4.



The figure consists of four screenshots illustrating the registration process:

- (a) Credenciais de registo:** A screenshot of the 'Register' form. It includes fields for First Name, Last Name, Gender (dropdown), Age, Height, Weight, Zip Code, Description, Username (dropdown), and a 'Create' button. The 'Zip Code' and 'Description' fields are highlighted in red, indicating they are required.
- (b) Credenciais de registo:** A screenshot of the 'Register' form showing the same fields as in (a), but with all fields filled out correctly. The 'Zip Code' and 'Description' fields are now grayed out, indicating they are no longer required.
- (c) Credenciais de registo:** A screenshot of the 'Register' form showing the same fields as in (a), but with the 'Create' button highlighted in blue.
- (d) Email de boas-vindas:** An email from RINTECare Porto to Alexandre Amorim. The subject is 'Welcome to RINTECare Porto'. The email body says: 'RINTECare Porto <rintecare2021@gmail.com> para Alexandre Amorim, Dear Alexandre Amorim, Congrats, you are set up on RINTECare Porto! Checkout our website and enjoy our Health+ blog. Best regards, RINTECare Porto'.

Figura 3.6: Processo de registo

Tendo o utilizador se registrado, é possível autenticar-se através da interface ilustrada na Figura 3.7, assim como lhe é possível redefinir a palavra-passe. Obtendo-se em no último caso um *email* como representado na Figura 3.8 podendo o utilizador alterar esta palavra passe na aplicação, assim que o desejar.

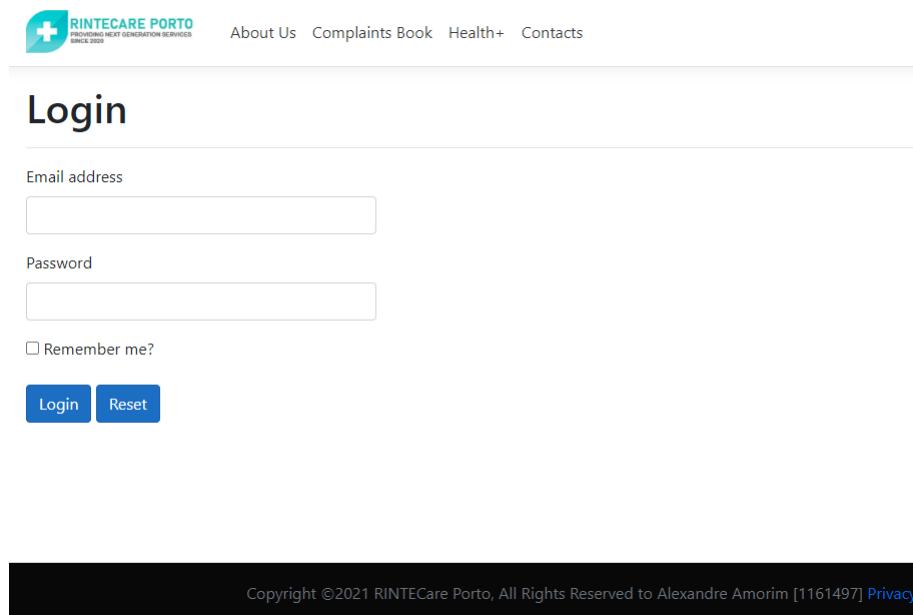


Figura 3.7: Autenticação de utilizador

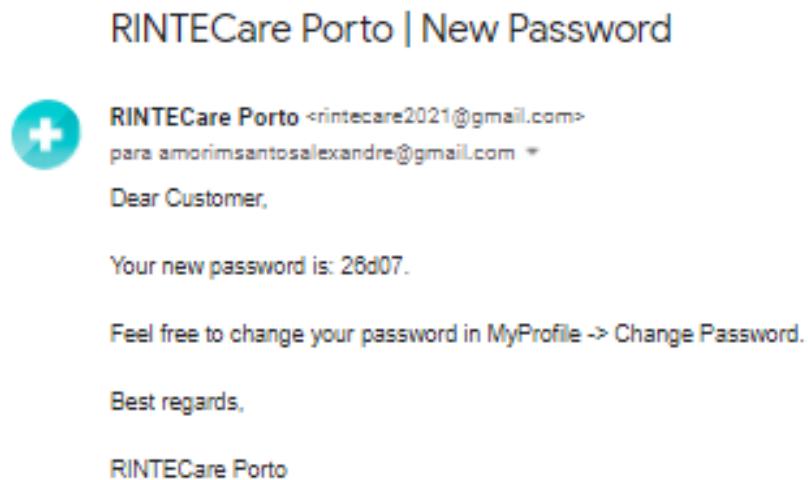


Figura 3.8: Email de redefinição de palavra-passe

### 3.3.2 Papel de Paciente

Na Figura 3.9 é possível verificar o menu principal com que o paciente se depara após se autenticar com sucesso. As quatro grandes funcionalidades que lhe são disponibilizadas pela aplicação são: alterar a sua palavra-passe, consultar o seu perfil e consultas, e por fim marcar uma consulta.

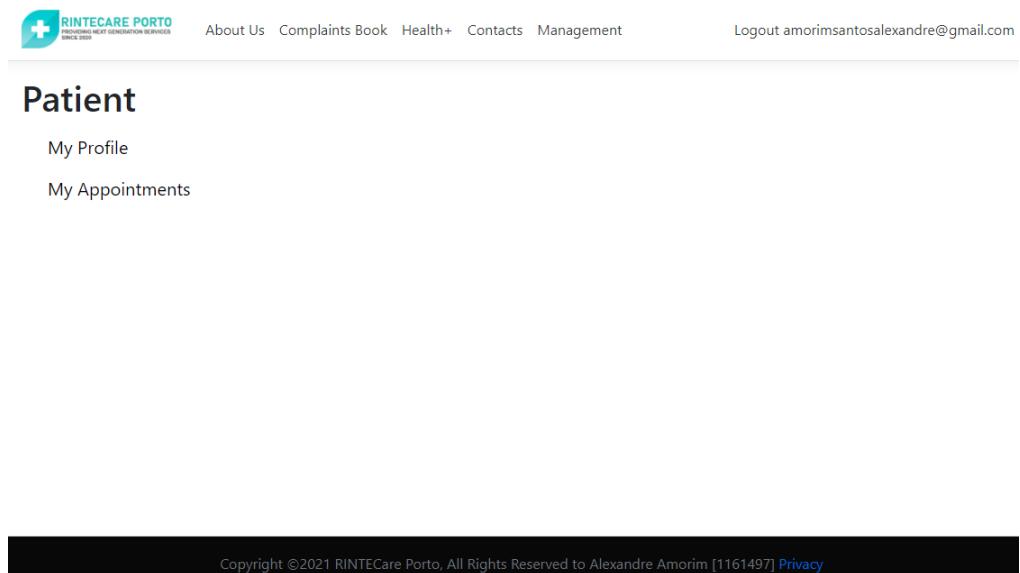


Figura 3.9: Vista de paciente

No sub-menu *My Profile* o utilizador pode gerir o seu perfil. Isto é, pode consultar/editar os seus dados (Figura 3.10a) e alterar a sua palavra-passe (Figura 3.10b).

First Name  
Afonso

Last Name  
Reis

Gender  
Male

Age  
20

Height  
180

Weight  
75

Blood Type  
A+

Password

Save

(a) Consulta/Edição de perfil

(b) Alteração de palavra-passe

Figura 3.10: Perfil do Paciente

Relativamente ao sub-menu *My Appointments*, é possível que o utilizador marque consultas, sendo que o processo consiste em selecionar a especialidade (Figura 3.11a), selecionar o médico (Figura 3.11b), selecionar o dia (Figura 3.11c), selecionar o período (Figura 3.11d) e por fim selecionar a data que pretende (Figura 3.11e).

Name	
Anesthesiologist	Select
Cardiologist	Select
Clinical Immunologists	Select
Dentist	Select
Dermatologist	Select
Endocrinologist	Select
Family Doctor	Select
Gastroenterologist	Select
General Surgeon	Select
Gynecologist	Select

(a) Escolha da especialidade

First Name	Last Name
Ana	Gulpilhares
André	Ferreira

(b) Escolha do médico

Name	
Sunday	Select
Monday	Select
Tuesday	Select
Wednesday	Select
Thursday	Select
Friday	Select
Saturday	Select

(c) Escolha do dia

Slot Information	Day Start	Day End	Select
08:00:00-09:30:00	06-01-2021	31-12-2021	Select
09:00:00-09:30:00	06-01-2021	13-01-2021	Select
10:00:00-10:30:00	06-01-2021	29-01-2021	Select
10:30:00-11:00:00	06-01-2021	31-01-2021	Select
11:30:00-12:00:00	06-01-2021	31-01-2021	Select

(d) Escolha do período

January 2021							February 2021								
Wk	Su	Mo	Tu	We	Th	Fr	Sa	Wk	Su	Mo	Tu	We	Th	Fr	Sa
52							1	2	4	1	2	3	4	5	6
53	3	4	5	6	7	8	9	5	7	8	9	10	11	12	13
1	10	11	12	13	14	15	16	6	14	15	16	17	18	19	20
2	17	18	19	20	21	22	23	7	21	22	23	24	25	26	27
3	24	25	26	27	28	29	30	8	28						
4	31							9							

Save

(e) Escolha da data

Figura 3.11: Processo de marcação de consulta

Sendo que posteriormente será enviado um *mail* de agendamento da consulta ao cliente e o seu respetivo agendamento na API Calendário Google, como ilustrado na Figura 3.12a e, 3.12b e 3.12c.

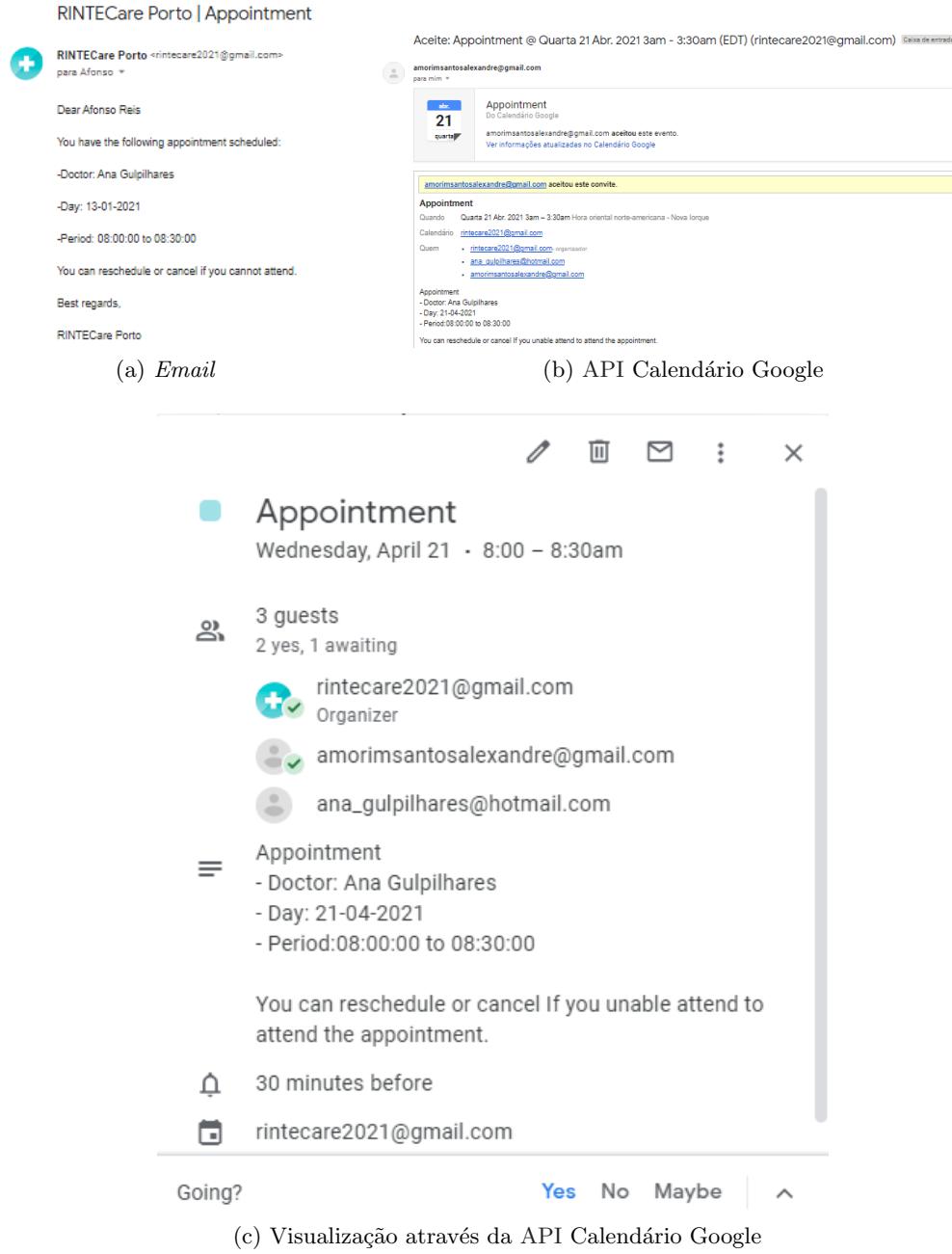


Figura 3.12: Notificação de agendamento de consulta

Por fim, o paciente pode consultar o seu histórico de consultas como presente na Figura 3.13. Adicionalmente verificam-se que existe cinco estados para a consulta: criada, cancelada pelo administrador, cancelado pelo médico, cancelada pelo paciente e realizada. A Tabela 3.1 resume os mesmos estados, podendo também se extrapolar que o estado criado pode evoluir para os restantes estados, mas os demais são imutáveis.



About Us Complaints Book Health+ Contacts Management Logout amorimsantosalexandre@gmail.com

Date	State	Patient	Physician	Week Day	Hours	
01/06/2021	Created	Afonso Reis	Ana Gulpilhares	Tuesday	08:00:00-08:30:00	<a href="#">Cancel</a>
03/03/2021	Cancelled by Client	Afonso Reis	Ana Gulpilhares	Tuesday	08:00:00-08:30:00	<a href="#">Cancel</a>
01/13/2021	Cancelled by Doctor	Afonso Reis	Ana Gulpilhares	Tuesday	08:00:00-08:30:00	<a href="#">Cancel</a>
03/24/2021	Cancelled by Administrator	Afonso Reis	Ana Gulpilhares	Tuesday	08:00:00-08:30:00	<a href="#">Cancel</a>
04/21/2021	Held	Afonso Reis	Ana Gulpilhares	Tuesday	08:00:00-08:30:00	<a href="#">Cancel</a>

Figura 3.13: Histórico de consultas

Tabela 3.1: Estados da consulta

Estado	Atores		Estado seguinte
	Administrador	Realizada	
Criada	Médico Paciente	Cancelada	
Realizada	Administrador Administrador	—	
Cancelada	Médico Paciente	—	

### 3.3.3 Papel de Médico

O perfil de médico acumula as funcionalidades do paciente, sendo de salientar que o processo de agendar uma consulta neste cenário ligeiramente diferente. A diferença reside em que o médico não precisa de introduzir a especialidade e o nome do médico (pois ao se autenticar o sistema já sabe a sua especialidade e nome). Contudo devido a poder ter múltiplos pacientes foi implementado na pesquisa de paciente um mecanismo de filtro do tipo "LIKE %texto%", como ilustrado na Figura 3.14.



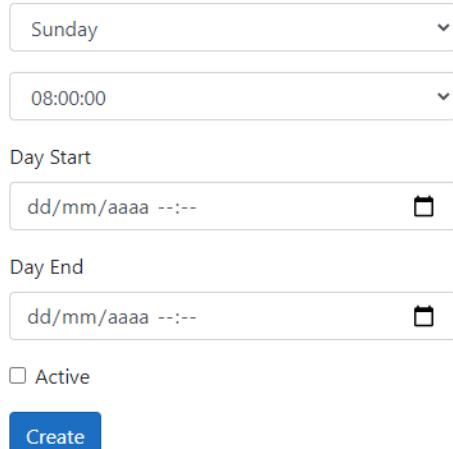
About Us Complaints Book Health+ Contacts Management Logout ana\_gulpilhares@hotmail.com

Lookup name here

First Name	Last Name	Gender	Age	Height	Weight	Blood Type	Allergies	Job	Tax Number	Description	
Afonso	Reis	Male	20	180	75	A+	None	Student	260900123	Abdominal Pains	<a href="#">Select</a>
Alice	Soares	Female	43	164	54	AB+	Peanuts	Nurse	260900124	Foot Pains	<a href="#">Select</a>
Ana	Cardoso	Female	12	156	50	A+	Peanuts	Student	260900125	Abdominal Pains	<a href="#">Select</a>
André	Medeiros	Male	36	178	77	A-	Peanuts	Architect	260900126	Abdominal Pains	<a href="#">Select</a>
Andreia	Costa	Female	39	170	73	O+	Peanuts	Secretary	260900127	Abdominal Pains	<a href="#">Select</a>
Ângela	Mourinho	Female	24	167	70	O-	Peanuts	Accountant	260900128	Abdominal Pains	<a href="#">Select</a>

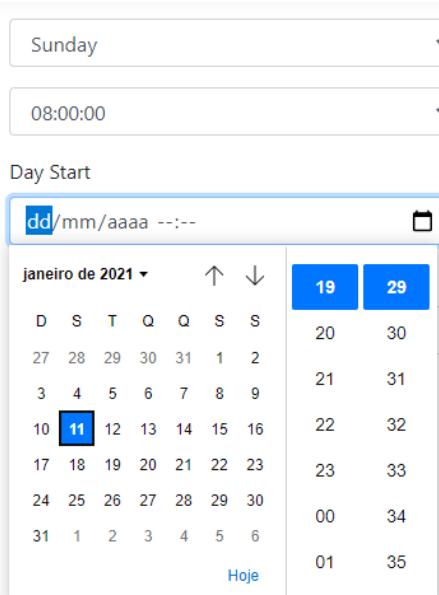
Figura 3.14: Procura de pacientes

Adicionalmente, o médico pode criar (Figura 3.15a e Figura 3.15b) e consultar (Figura 3.15c) a sua disponibilidade.



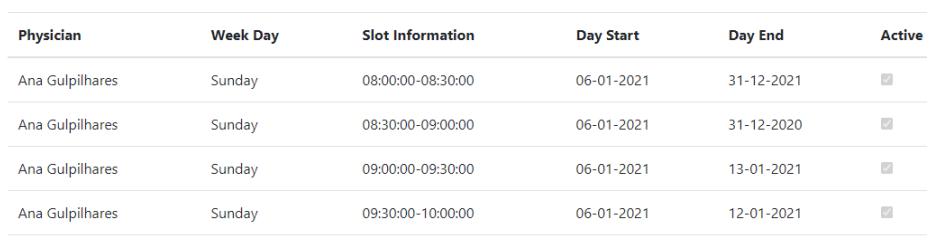
(a) Criação de disponibilidade

O formulário para criação de disponibilidade mostra campos para Week Day (semana dia), Slot Information (informação de slot), Day Start (dia de início) e Day End (dia de término). Um checkbox para 'Active' (ativo) está desmarcado. Um botão 'Create' (criar) está no final.



(b) Seleção do intervalo de dias

O calendário mostra o mês de Janeiro de 2021. O dia 11 é selecionado. Outros dias como 19, 29, 20, 30, 21, 31, 22, 32, 23, 33, 24, 34, 31, 01 e 35 também são visíveis.



(c) Consulta de disponibilidade

A tabela mostra a disponibilidade de um médico (Ana Gulpilhares) ao longo de uma semana. As colunas são: Physician (Médico), Week Day (semana dia), Slot Information (informação de slot), Day Start (dia de início) e Day End (dia de término). A coluna 'Active' (ativo) contém ícones que podem ser clicados.

Physician	Week Day	Slot Information	Day Start	Day End	Active
Ana Gulpilhares	Sunday	08:00:00-08:30:00	06-01-2021	31-12-2021	<input checked="" type="checkbox"/>
Ana Gulpilhares	Sunday	08:30:00-09:00:00	06-01-2021	31-12-2020	<input checked="" type="checkbox"/>
Ana Gulpilhares	Sunday	09:00:00-09:30:00	06-01-2021	13-01-2021	<input checked="" type="checkbox"/>
Ana Gulpilhares	Sunday	09:30:00-10:00:00	06-01-2021	12-01-2021	<input checked="" type="checkbox"/>
Ana Gulpilhares	Sunday	10:00:00-10:30:00	06-01-2021	29-01-2021	<input checked="" type="checkbox"/>

Figura 3.15: Gestão de disponibilidade

### 3.3.4 Papel de Administrador

Na Figura 3.16 encontra-se retratado o painel do administrador onde é possível verificar que reúne todas as funcionalidades anteriormente mencionadas, bem como gera os administradores, as especializações, os médicos, os pacientes e os utilizadores.

Figura 3.16: Painel de administrador

Assim como no perfil anterior, o administrador tem muitas similaridades na marcação de consultas, contudo neste caso acresce a especialidade e o nome do médico. Ademais, é o único perfil que pode alterar o estado de uma consulta para realizada (Figura 3.17a), sendo despotelada uma mensagem de faturação para o paciente, como ilustrado na Figura 3.17b e Figura 3.17c, respetivamente.

Week Day	Hours	Date	Patient	Physician	State		
Tuesday	08:00:00-08:30:00	12/15/2020	Ana Cardoso	Ana Gulpilhares	Cancelled by Administrator	Held	Cancel
Tuesday	08:00:00-08:30:00	12/30/2020	Afonso Reis	Ana Gulpilhares	Held	Held	Cancel
Tuesday	08:00:00-08:30:00	01/13/2021	Afonso Reis	Ana Gulpilhares	Held	Held	Cancel
Tuesday	08:00:00-08:30:00	01/20/2021	Afonso Reis	Ana Gulpilhares	Cancelled by Client	Held	Cancel

(a) Opção de consulta realizada

(b) Email de faturação

(c) Fatura da consulta

Figura 3.17: Processo de consulta realizada

Por fim, a gestão de operações realizada pelo administrador é baseada em CRUD, sendo que deste modo pode eliminar, consultar, editar e criar administradores (Figura 3.18a), consultas (Figura 3.18b), especialidades (Figura 3.18c), médicos (Figura 3.18d), pacientes (Figura 3.18e) e utilizadores (Figura 3.18f).

The figure consists of six screenshots labeled (a) through (f), each showing a different administrative interface of the Intecare Porto application.

- (a) Administrador:** Shows a table of administrators with columns: First Name, Last Name, Gender, Age, Phone Number, Location, Address, and Zip Code. Data includes Alexandre Felgueiras, Rui Figueiredo, and Vitor Feraz.
- (b) Consulta:** Shows a table of consultations with columns: Week Day, Hours, Date, Patient, Physician, and State. Data includes entries for Ana Gulpilhares and Afonso Reis.
- (c) Especialidade:** Shows a table of medical specialties with columns: Name, First Name, Last Name, Gender, Age, Phone Number, Location, Address, and Zip Code. Data includes Anesthetist, Cardiologist, Clinical Immunologists, Dentist, Dermatologist, and Endocrinologist.
- (d) Médico:** Shows a table of doctors with columns: First Name, Last Name, Gender, Age, Phone Number, Location, Address, and Zip Code. Data includes Ana Gulpilhares, André Ferreira, Américo Regal, Beatriz Cunha, and Bernardo Santos.
- (e) Paciente:** Shows a table of patients with columns: First Name, Last Name, Gender, Age, Phone Number, Location, Address, and Zip Code. Data includes Afonso Reis, Alice Soares, Ana Cardoso, André Medeiros, and Andreia Costa.
- (f) Utilizador:** Shows a table of users with columns: Username, Password, Role, and Active. Data includes admin accounts for alex\_felgueiras@gmail.com, rui\_figueiredo@gmail.com, vitor\_pascoal@gmail.com, ana\_gulpilhares@hotmail.com, and andre\_ferreira@live.com.

Figura 3.18: Gestão de Operações

### 3.3.5 Funcionalidades Adicionais

Em complemento as funcionalidades principais da aplicação, foram adicionadas funcionalidades que permitem ao utilizador averiguar uma série de informações tais como: comentários de utilizadores, notícias da clínica, contactos, horários, regulamentos de privacidade, *blog* sobre artigos de saúde, a sua localização, entre outros.

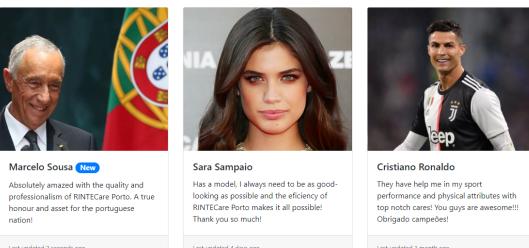
No momento em que o utilizador inicia a aplicação, depara-se com a página Casa, onde são descritos a missão e valores da entidade (Figura 3.20a), as suas especialidades (Figura 3.20b), os comentários mais relevantes a sujeito dos serviços da clínica (Figura 3.20c), a sua presença nas últimas notícias (Figura 3.20d) e uma vertente presente em todas as páginas da aplicação, o regulamento de privacidade (Figura 3.20e).



(a) Missão e Valores



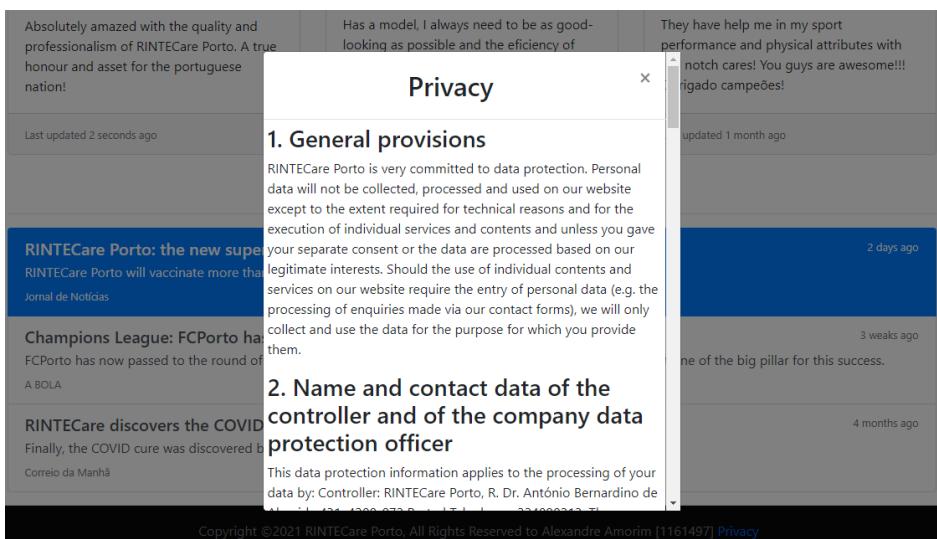
(b) Especialidades



(c) Comentários



(d) Notícias



(e) Regulamento de privacidade

Figura 3.19: Página Casa

O próximo separador da janela de navegação é o Sobre Nós, elemento presente na grande maioria das aplicações desta natureza. Neste elemento é possível encontrar informação geral da entidade (Figura 3.20a), as certificações da mesma (Figura 3.20b) e os depoimentos mais notáveis (Figura 3.20c).



Figura 3.20: Esquema da página Sobre Nós

Outra prática muito característica deste tipo de aplicações na vertente da saúde é a inclusão de um livro de reclamações. Deste modo, seguiu-se as normas gerais e contemplou-se a mesma através do *link*<sup>4</sup>, como ilustrado na Figura 3.21.

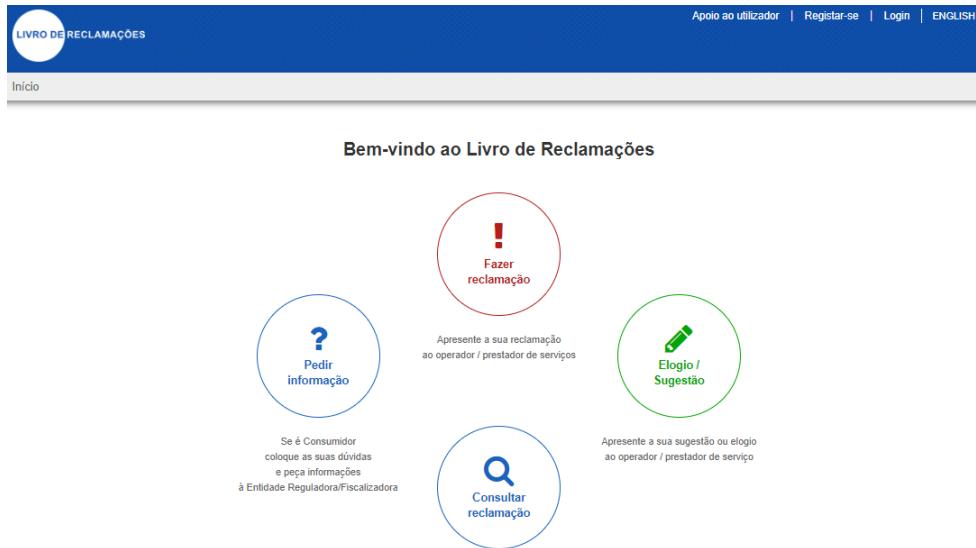


Figura 3.21: Livro de reclamações

<sup>4</sup><https://www.livroreclamacoes.pt/inicio>

Outra vertente extremamente importante neste âmbito é a inovação. Assim, realizou-se mais uma funcionalidade que permite aos médicos possam partilharem artigos científicos através do *blog Health+*, podendo os utilizadores acederem aos mesmos por ordem cronológica e saber as suas datas de concepção, como ilustrado nas Figuras 3.22a, 3.22b e 3.22c.

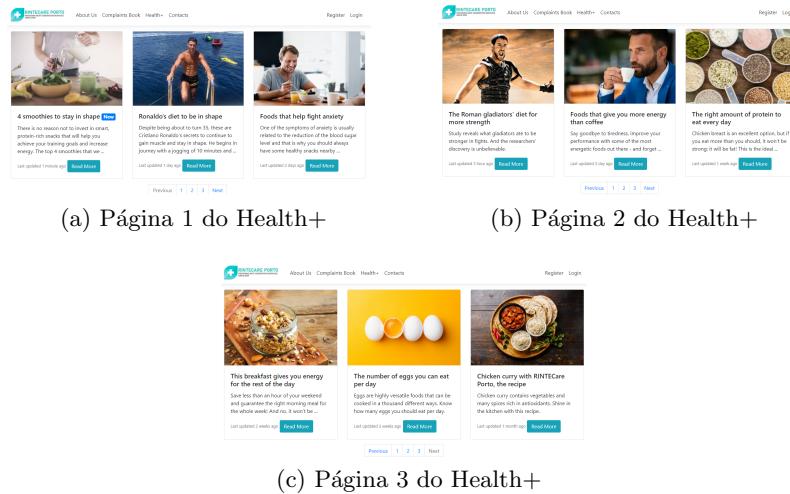


Figura 3.22: *Blog Health+*

Finalmente, a última funcionalidade extra de grande valor para os pacientes são os contactos. Os contactos mais relevantes apurados pelo *benchmarking* do mercado e desenvolvidos na aplicação foram o *email/telefone* (Figura 3.23a), as redes sociais (Figura 3.23b) e a localização no Google Maps (Figura 3.23a), de modo a facilitar a comunicação do utilizador com a entidade RINTECare Porto.



Figura 3.23: Resumo da página de contactos

## 3.4 Sumário

Neste capítulo, foram descritas a modelação da aplicação através de casos de uso e do diagrama da BD e, de toda a componente de desenvolvimento da aplicação. Em termos de panorama geral, conclui-se que a versão atual da prova de conceito da aplicação Web da RINTECare Porto satisfaça todos os requisitos desejados, até transcendendo os objetivos propostos inicialmente. Obtendo-se deste modo um projeto muito próximo do praticado no mercado profissional.

No entanto, prevê-se várias melhorias em termos de usabilidade e processo de extração de dados. Em termos de usabilidade, prevê-se que a escolha do dia, da hora e da data serem realizáveis através de um único calendário. Na mesma vertente, seria interessante o desenvolvimento de um *chatbot* que fizesse sugestões de marcações de consultas segundo as análises de dados dos pacientes, de modo a automatizar este processo. Outra possível melhoria seria a elaboração de *dashboards* para os respetivos utilizadores, de modo a consultarem as informações através de gráficos e obterem uma célebre percepção sobre os dados do seu interesse.

Relativamente ao processo de extração de dados, apesar de se terem emitidos e gerados PDF de faturação, o projeto inicialmente não estava projetado para esse fim, sendo que seria uma boa prática desenvolver as suas respetivas tabelas na BD para controlar esta operação. Outra possível medida de melhorar o processo de extração de dados dos utilizadores seria que o utilizador não tivesse de preencher todos os campos necessários para os médicos saberem o seu perfil de paciente, mas sim através de inserir unicamente o seu número de cartão de cidadão. Sendo possível obter com essa informação os seus restantes dados de acordo com a BD do Sistema Nacional de Saúde (parceria de entidades). Por fim, uma última melhoria passaria por a implementação de *Linked Open Data* (LOD) no *blog Health+*, que permitiria obter ainda mais fontes de informações, de uma forma mais simples e eficaz.



## **Capítulo 4**

---

# **Avaliação da Aplicação em Ambiente Produtivo**

---

*Neste capítulo são apresentados os resultados obtidos nas diferentes avaliações e testes a qual a aplicação foi submetida. Na Secção 4.1 são descritos os resultados obtidos nos testes funcionais, enquanto que na Secção 4.2 são demonstrados para os cenários dos testes não funcionais, contemplando testes de carga, de tempo de resposta e de usabilidade.*

No desenvolvimento de aplicações Web, é necessário realizar avaliações e testes dos mesmos, de modo a antecipar e corrigir falhas e, possíveis *bugs* que apareceriam ao utilizador final [11, 12], como referido por James Bach:

"Testing is an infinite process of comparing the invisible to the ambiguous in order to avoid the unthinkable happening to the anonymous." - James Bach

Estes problemas podem surgir de diferentes aspectos tais como o cliente lidar com instabilidades ou dificuldades em aceder a páginas defeituosas. Para mitigá-los, recorreu-se a diferentes tipos de testes para identificar e organizar os erros. Os testes foram realizados tendo em foco os utilizadores externos do sistema (médicos e pacientes), sendo estes testes enumerados de seguida:

- Testes funcionais - implementados para determinar se a aplicação cumpre as funções definidas no levantamento de requisitos. Permitindo assim encontrar falhas ou mesmo possíveis aprimoramentos no que já se encontra funcional.

- Teste não funcionais - permitem avaliar o desempenho da aplicação e subsequentemente detetar as vertentes que carecem de otimizações ou até mesmo da sua viabilidade na aplicação.

## 4.1 Testes Funcionais

A aplicação foi concebida de modo a que só seja apresentado ao utilizador as funcionalidades estritamente necessárias, dando-se assim preferência a componentes como *dropdowns*, calendários e botões, ao invés de caixas de texto (principalmente fonte de problemas neste contexto). Reduzindo desse modo a complexidade de decisão para o utilizador e tornando por sua vez a aplicação mais *user friendly*.

Os primeiros testes funcionais com que o utilizador se depara são os de componente de registo. Nas Figuras B.12a, B.12b, B.12c e B.12d é possível observar que o não preenchimento dos campos levanta os seus respetivos avisos, bem como o de não cumprimento das regras impostas para cada campo.

The figure consists of four screenshots labeled (a) through (d), each showing a registration form with validation errors:

- (a) Etapa 1 do registo:** Shows fields for First Name, Last Name, Gender, and Age. The First Name and Last Name fields have red error messages: "First Name needed" and "Last Name needed" respectively. The Age field has a red message: "The value '' is invalid."
- (b) Etapa 2 do registo:** Shows fields for Height, Weight, and Blood Type. The Height and Weight fields have red error messages: "Insert your valid height in centimeters (p.e. 180)" and "Insert your valid weight in kilograms (p.e. 75)" respectively. The Blood Type field has a red message: "Select a Blood Type".
- (c) Etapa 3 do registo:** Shows fields for Allergies, Job, Tax Number, Phone Number, and Location. The Allergies, Job, and Location fields have red error messages: "Insert your allergies. In case you don't have introduce None", "Insert your job title", and "Insert your location" respectively. The Tax Number field has a red message: "The value '' is invalid.". The Phone Number field has a red message: "The value '' is invalid."
- (d) Etapa 4 do registo:** Shows fields for Address, Zip Code, Description, Username, and a large "Create" button. The Address, Zip Code, and Description fields have red error messages: "Insert your job address", "Insert your zip code", and "Insert your description" respectively. The Username field has a red message: "Insert your email".

Figura 4.1: Testes funcionais de registo

Por conseguinte, os testes funcionais com que o utilizador se depara são relativos a componente de autenticação. Na eventualidade do utilizador não introduzir o seu email/palavra-passe ou os dados se encontrarem incorretos, avisos como os ilustrados na Figura B.12a irão ser apresentados. Ademais, no cenário em que o utilizador não se lembra da sua palavra-passe e quiser alterá-la, terá que introduzir o seu email correto. Caso contrário, será lhe apresentado o aviso presente Figura B.12b.

## Login

Email address

The email address is required

Password

The password is required

Remember me?

[Login](#) [Reset](#)

User or password invalid

(a) Credenciais de autenticação

Username

[Create](#)

username:blank@gmail.com does not exist in our system. Please, check if you inserted your correct username or contact our support team.

[Back to Login](#)

(b) Alteração de palavra-passe

Figura 4.2: Testes funcionais de autenticação

Finalmente, o último teste funcional consiste na restrição da alteração do estado da consulta. Mais especificamente, só é permitido aos utilizadores criar, anular e dado por concluído as consultas. Não sendo possível anular ou dar por concluída uma consulta sem que esteja primeiramente criada ou se uma das operações já foi efetuada por outro utilizador, como ilustrado na Figura 4.3

Tuesday	08:00:00-08:30:00	01/06/2021	Afonso Reis	Ana Gulpilhares	Created	<a href="#">Held</a>	<a href="#">Cancel</a>
Tuesday	08:00:00-08:30:00	03/03/2021	Afonso Reis	Ana Gulpilhares	Cancelled by Client	<a href="#">Held</a>	<a href="#">Cancel</a>
Tuesday	08:00:00-08:30:00	01/13/2021	Afonso Reis	Ana Gulpilhares	Cancelled by Doctor	<a href="#">Held</a>	<a href="#">Cancel</a>
Tuesday	08:00:00-08:30:00	03/24/2021	Afonso Reis	Ana Gulpilhares	Cancelled by Administrator	<a href="#">Held</a>	<a href="#">Cancel</a>
Tuesday	08:00:00-08:30:00	04/21/2021	Afonso Reis	Ana Gulpilhares	Held	<a href="#">Held</a>	<a href="#">Cancel</a>

Figura 4.3: Alteração do estado da consulta

## 4.2 Testes Não Funcionais

Por fim, foram realizados testes não funcionais de modo a avaliar o desempenho da aplicação, em termos de carga, tempo de resposta e de usabilidade. Em primeira análise, incide-se nas páginas externas aos perfis de utilizadores. Sendo que como esperável, as páginas que necessitam de maior carga de dados e que consomem maior tempo de resposta são as que incorporam uma panóplia de imagens/mapas embbebidos, tais como a Casa, Contactos e Sobre Nós. De referir, que apesar da página Livro de Reclamações não ser a com maior carga de dados é a que tem a segunda maior latência devido a ser uma página externa à aplicação desenvolvida. Isto é, irá ser encaminhada e submetido a diversos testes de segurança até alcançar o domínio [www.livroreclamacoes.pt](http://www.livroreclamacoes.pt), como ilustrado na Figura B.1 presente nos Anexos. Outro aspeto necessário referir é que a página de Proteção de Dados é a com menor carga e subsequentemente menor tempo de resposta, dado incorporar apenas caracteres.

Tabela 4.1: Testes Não-Funcionais das Páginas Exteriores

Nome da Página	Carga de Dados (B)	Tempo de Resposta (ms)
Autenticação	942 k	119
Casa	2,7 M	138,17
Contactos	2,4 M	1160
Health+	1,0 M	126
Livro de Reclamações	1,6 M	172,92
Proteção de Dados	820	32
Redefinição de Palavra-Passe	942 k	125
Registrar	949 k	134
Sobre Nós	4,0 M	183

Relativamente, aos testes não funcionais das páginas alusivas aos perfis de utilizador pode se concluir que a página com maior carga é a da Escolha da Data, devido a possuir dois calendários (*datepickers*) (componentes que envolvem uma maior geração de carga de dados) e que a página com maior exigências de tempo de resposta é da Marcação de Consultas, pois envolve a comunicação com o Gmail e com a API Calendário Google (sendo possível visualizar as suas rotas na Figura B.2 e Figura B.3, respetivamente), como descrito na Tabela 4.2. De salientar que as páginas Consultar Pacientes e Visualização de Consultas se encontram respetivamente regidas pelas seguintes equações:

$$c_{paciente} = 996 + 1,17 \times n \text{ kB} \quad (4.1)$$

$$c_{consulta} = 941 + 1,13 \times n \text{ kB} \quad (4.2)$$

sendo  $n$  o número de pacientes/consultas em causa.

Tabela 4.2: Testes Não-Funcionais Utilizadores Externos

Nome da Página	Carga de Dados (B)	Tempo de Resposta (ms)
Alteração de Palavra-Passe*	942 k	135
Consultar Pacientes	$996 + 1,17n$ k	100
Escolha da Data*	1,6 M	343
Escolha do Dia da Semana*	944 k	127
Escolha da Especialidade	947 k	130
Escolha do Horário	947 k	131
Escolha do Médico	941 k	122
Escolha do Período*	947 k	188
Index*	732 k	98
Marcação de Consultas*	1.2 M	400
Visualização de Consultas*	$941 + 1,13n$ k	120
Visualização de Perfil*	945 k	139
Sair*	2,7 M	238

\* Páginas presentes tanto para os médicos como para os pacientes

O último teste destina-se a avaliação da usabilidade da aplicação. Assim, realizou-se um *benchmarking* do estado atual da aplicação com três das melhores soluções do mercado nacional, Hospital CUF<sup>5</sup>, Hospital da Luz<sup>6</sup> e Hospital Lusíadas<sup>7</sup>, como sintetizado na Tabela 4.3. Na Secção B.2 encontra-se a metodologia concernente ao Hospital da Luz, sendo replicada para as restantes soluções. As principais operações para os pacientes são o processo de cancelar consultas, contactar assistentes, marcar consultas e registar. Destes processos, a presente solução só não lidera no registo. Contudo, se fosse comercializada, passaria-se de 18 *cliques* para 9, concluindo-se que tem um comportamento exímio, fruto de diversas tentativas de melhoria continua.

Tabela 4.3: Teste de Usabilidade

Estado	Número de <i>cliques</i>	<i>Benchmarking</i>		
		Hospital da Luz	Hospital Lusíadas	Hospital CUF
Autenticar	4	4	4	4
Cancelar consulta	3	4	5	3
Contactar assistentes	2	3	1	2
Consultar perfil	2	2	3	1
Editar consulta	6	9	9	6
Marcar consulta	8	12	17	8
Redefinir palavra-passe	3	4	4	4
Registrar	18	15	10	15
Visualizar consulta	3	2	2	2

<sup>5</sup><https://www.cuf.pt/>

<sup>6</sup><https://www.hospitaldaluz.pt/arrabida/pt>

<sup>7</sup><https://www.lusiadas.pt/hospitais-clinicas/hospital-lusiadas-porto>

### 4.3 Sumário

O presente capítulo permitiu avaliar a aplicação em ambiente de produção. Num panorama holístico, os testes realizados permitiram concluir que a aplicação se encontra “*ready-to-market*”, dado a sua boa prestação nos testes de carga, de tempo de resposta e de usabilidade. No entanto, a aplicação poderia ser otimizada a respeito dos testes de carga de dados e de tempo de resposta, através da conversão do formato das imagens de *Joint Photographic Experts Group* (JPEG) para Identificador de Recurso Uniforme Identificador de Recurso Uniforme (URI) e/ou para *Cascading Style Sheets* (CSS) *sprites*. Isto pois, a aplicação desenvolvida tem como protocolo o Protocolo de Transferência de Hipertexto Seguro (HTTPS), assim é exigido que todos os elementos das suas páginas sejam transferidas por conexões seguras, o que pode levar a elevada latência devido a quantidade imagens que necessitam de serem transferidas. Sendo que os URI e os CSS *sprites* permitem reduzir pedidos HTTPS e assim tornando mais célere a disponibilização da página para o visitante.

Por outro, também se perspetiva que a introdução de um *chatbot* permitiria reduzir o número de *cliques* necessários para os utilizadores realizarem determinadas tarefas. No momento em que o utilizador navegassem na aplicação, bastaria selecionar a funcionalidade que necessita, sendo respondido de imediato pelo *chatbot* sem ter que se autenticar (na verdade, dependendo do pedido do utilizador, o *chatbot* poderia lhe pedir o seu número de utente), reduzindo desse modo a quantidade de *cliques*, como ilustrado na Figura 4.4 adaptada do site do Hospital da Prelada<sup>8</sup>.

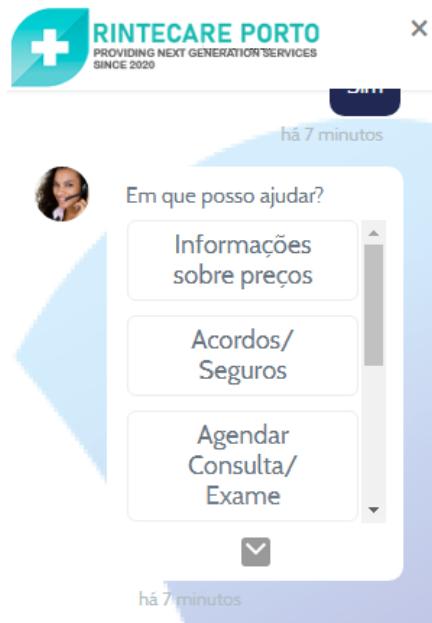


Figura 4.4: Proposta de *chatbot* para a RINTECare Porto

<sup>8</sup><https://portaldasaude.scmp.pt>

## Capítulo 5

---

# Conclusão

---

O projeto atual aborda uma infinidade de tópicos e abordagens, nesta secção, examinamos cada um destes. Serão apresentadas as conclusões, a revisão dos objetivos propostos para este projeto, enumeração das tecnologias empregues e sugestão de ideias de trabalho futuro.

Inicialmente, foi identificado os problemas e objetivos inerentes ao desenvolvimento de uma aplicação no contexto de uma clínica de saúde. Para esse contexto foi necessário realizar um *benchmarking* das soluções existentes no mercado, tendo-se concluído que para uma prova de conceito seria necessário elaborar três perfis de utilizadores (administrador, médico e paciente), possibilitar a gestão de processos (administrador, consultas, disponibilidades, médicos, pacientes e utilizadores) e o envio notificações para os pacientes *via email*. Sendo que como se pode verificar ao longo do relatório, esses objetivos foram transcendidos com sucesso.

Posteriormente, foi necessário realizar um estudo das tecnologias atuais para a vertente de desenvolvimento de aplicações em causa. Tendo-se verificado que as tecnologias mais adequadas neste contexto serem as BD SQL, dado ser necessário um maior nível qualidade de serviço em termos de segurança de informação e não ser predominante elevados tráfegos de dados; o framework ASP.NET Core de modo a tornar o desenvolvimento mais célere e beneficiar das melhores práticas de desenvolvimento de *Web Apps* do mercado; o ORM Dapper que permite uma maior flexibilidade e agilidade de mapeamento entre a BD e as linguagens de programação; a API Calendário Google devido a permitir efetuar uma gestão completa de eventos e da sua forte presença na vida da maioria dos utilizadores e por fim do Wkhtmltopdf para gerar os PDF de faturação em código aberto.

No entanto, prevê-se várias melhorias em termos de usabilidade e processo de extração de dados. Em termos de usabilidade, prevê-se que a escolha do dia, da hora e da data serem realizáveis através de um único calendário. Na mesma vertente, seria

interessante o desenvolvimento de um *chatbot* que fizesse sugestões de marcações de consultas segundo as análises de dados dos pacientes, de modo a automatizar este processo. Outra possível melhoria seria a elaboração de *dashboards* para os respetivos utilizadores, de modo a consultarem as informações através de gráficos e obterem uma célere percepção sobre os dados do seu interesse.

Relativamente ao processo de extração de dados, apesar de se terem emitidos e gerados PDF de faturação, o projeto inicialmente não estava projetado para esse fim, sendo que seria uma boa prática desenvolver as suas respetivas tabelas na BD para controlar esta operação. Outra possível medida de melhorar o processo de extração de dados dos utilizadores seria que o utilizador não tivesse de preencher todos os campos necessários para os médicos saberem o seu perfil de paciente, mas sim através de inserir unicamente o seu número de cartão de cidadão. Sendo que com essa informação possível obter os seus restantes dados de acordo com a BD do Sistema Nacional de Saúde (parceria de entidades). Por fim, uma última melhoria passaria por a implementação de LOD no *blog Health+*, que permitiria obter ainda mais fontes de informações, de uma forma mais simples e eficaz.

Por fim, foram realizadas avaliações e testes da aplicação, de modo a avaliar a sua performance em termos de carga de dados, tempo de resposta e usabilidade. Em traços gerais, a aplicação responde como previsto, tendo sido comparada a três das melhores soluções do mercado nacional e liderado em muitos aspectos. Adicionalmente, é necessário salientar que é inovadora no aspeto em que integra a API Calendário Google, o que facilita a vida dos seus utilizadores em termos de gestão de tempo. Contudo, e apesar das diversas melhorias continuas realizadas ao longo deste projeto, acredita-se que a introdução de um *chatbot* e da implementação de URI e/ou CSS *sprites* poderia tornar a solução mais eficiente e eficaz, otimizando os três critérios de avaliação. Em termos conclusivos, apesar do projeto ser uma prova de conceito, reúne diversos aspectos que a põe no patamar de solução *ready-to-market* e deste modo preenchendo todos os requisitos solicitados com sucesso.

---

# Referências Bibliográficas

---

- [1] M. Jazayeri, "Some trends in web application development," in *Future of Software Engineering (FOSE '07)*, pp. 199–213, 2007. [citado na p. 2]
- [2] N. Dissanayake and G. Dias, "Web-based applications: Extending the general perspective of the service of web," in *Proc. 10th International Research Conference of KDU (KDU-IRC 2017) on Changing Dynamics in the Global Environment: Challenges and Opportunities*, 2017. [citado na p. 2]
- [3] J. Singer, R. Gifford, M. Cotten, and D. Robertson, "Cov-glue: a web application for tracking sars-cov-2 genomic variation," *Preprints*, 2020. [citado na p. 2]
- [4] Microsoft, "What is sql server management studio (ssms)?," 2019. <https://docs.microsoft.com/en-us/sql/ssms/sql-server-management-studio-ssms?redirectedfrom=MSDN&view=sql-server-ver15> [Acedido a 23 de Dezembro 2020]. [citado na p. 6]
- [5] M. Rouse, "Internet information services (iis)," 2019. <https://searchwindowsserver.techtarget.com/definition/IIS> [Acedido a 23 de Dezembro 2020]. [citado na p. 6]
- [6] Microsoft, "Overview of asp.net core mvc," 2020. <https://docs.microsoft.com/en-us/aspnet/core/mvc/overview?view=aspnetcore-5.0> [Acedido a 23 de Dezembro 2020]. [citado na p. 6]
- [7] A. Feller, "Mvc framework," 2015. <https://www.slideshare.net/AshtonFeller/mvc-framework> [Acedido a 23 de Dezembro 2020]. [citado na p. 7]
- [8] Dapper, "What's dapper?," 2020. <https://dapper-tutorial.net/dapper> [Acedido a 23 de Dezembro 2020]. [citado na p. 7]
- [9] Google, "What can you do with calendar?," 2021. <https://support.google.com/a/users/answer/9302892?hl=en> [Acedido a 12 de Janeiro 2021]. [citado na p. 8]

- [10] J. Truelson, "Wkhtmltopdf," 2021. <https://wkhtmltopdf.org/> [Acedido a 11 de Janeiro 2021]. [citado na p. 8]
- [11] K. Ali and X. Xiaoling, "A reliable and an efficient web testing system," *International Journal of Software Engineering & Applications*, vol. Vol 10, p. 1 to 16, 01 2019. [citado na p. 29]
- [12] D. Lakshmi and S. Mallika, "A review on web application testing and its current research directions," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 7, p. 2132, 08 2017. [citado na p. 29]

## Anexo A

---

# Instalação e Configuração do Ambiente de Desenvolvimento

---

### A.1 Servidor de Base de Dados

Nesta secção, irão ser descritos os procedimentos de instalação e de configuração do servidor de base de dados. Deste modo, propõe-se duas abordagens para este fim: uma visão holística como representado no fluxograma da Figura A.1, e de seguida uma abordagem mais detalhada.

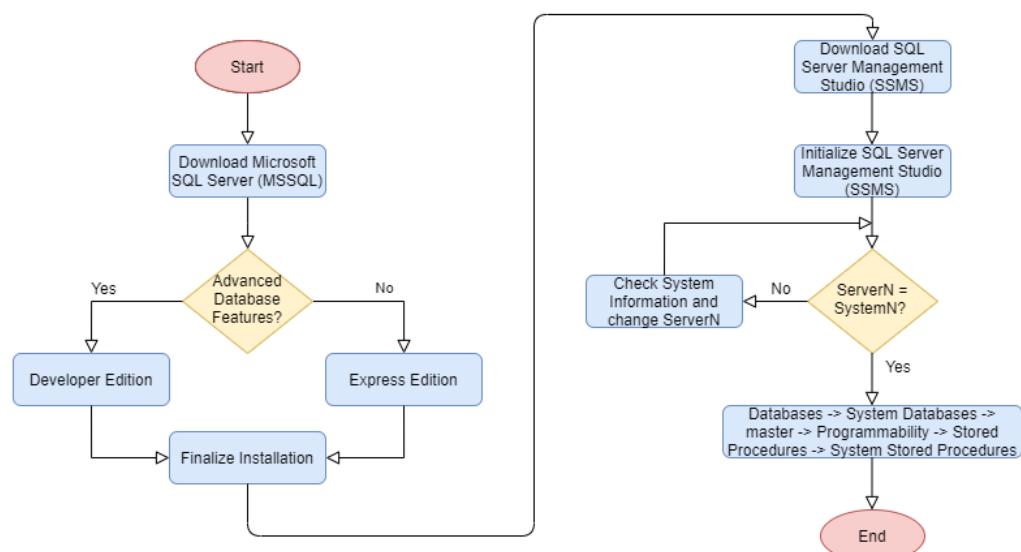


Figura A.1: Visão holística de instalação e configuração dos *softwares* em estudo

## ANEXO A. INSTALAÇÃO E CONFIGURAÇÃO DO AMBIENTE DE DESENVOLVIMENTO

### Servidor Microsoft SQL

Inicialmente, o utilizador deverá navegar no *link*<sup>9</sup>, optar pelo *download* da edição do servidor SQL 2019 *Developer* (passo 1) e quando finalizado executar o ficheiro SQL2019-SSEI-Dev.ex (passo 2), como ilustrado na Figura A.2(a). Relativamente à escolha do tipo de instalação, a Figura A.2(b) evidencia a sugestão de optar pela edição básica, de modo a tornar o processo de configuração/instalação mais célere.

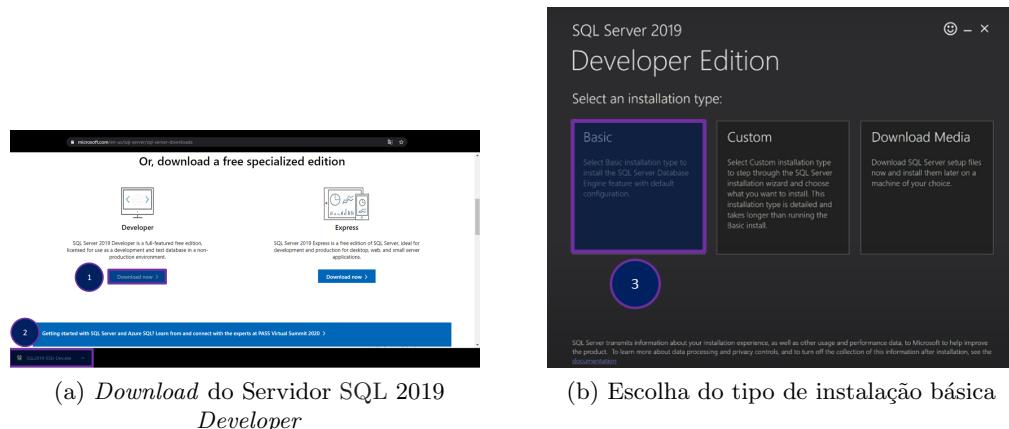


Figura A.2: *Download* e configuração do servidor SQL

À posteriori, deverá realizar a aceitação dos termos de licença (passo 4 - Figura A.3(a)) e especificar o local de instalação (passo 5 - Figura A.3(b)). Sendo que se sugere analisar o *link*<sup>10</sup> para obtenção de mais informações concernente as restantes opções não selecionadas.

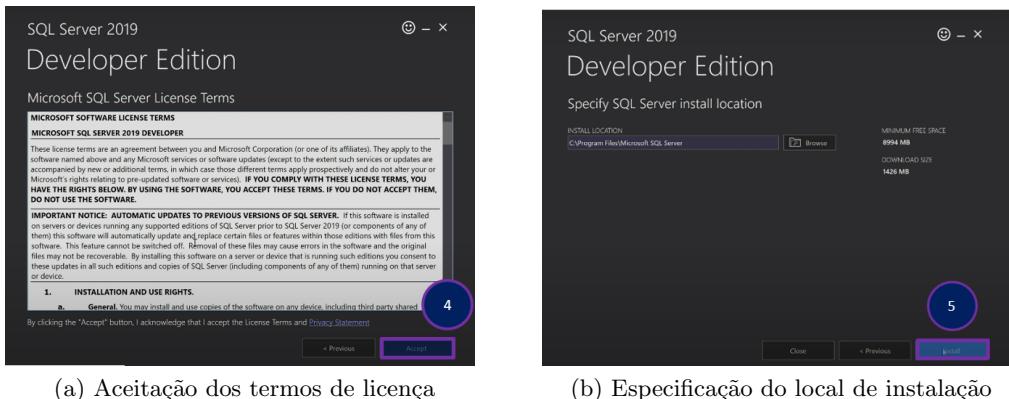


Figura A.3: Finalização da configuração do servidor SQL

<sup>9</sup> <https://www.microsoft.com/en-us/sql-server/sql-server-downloads>

<sup>10</sup> <https://docs.microsoft.com/en-us/sql/sql-server/editions-and-components>

### SQL Server Management Studio

Ulteriormente, é necessário realizar o mesmo processo realizado para o SSMS. Ao finalizar as etapas anteriores, o utilizador será confrontado com a interface presente na Figura A.4(a), onde deverá premir o botão *Install SSMS* (passo 6). Posteriormente, será reencaminhado para o site<sup>11</sup> ilustrado na Figura A.4(b), no qual efetuará o *download* do SSMS (passo 7) e executará o ficheiro obtido (passo 8).

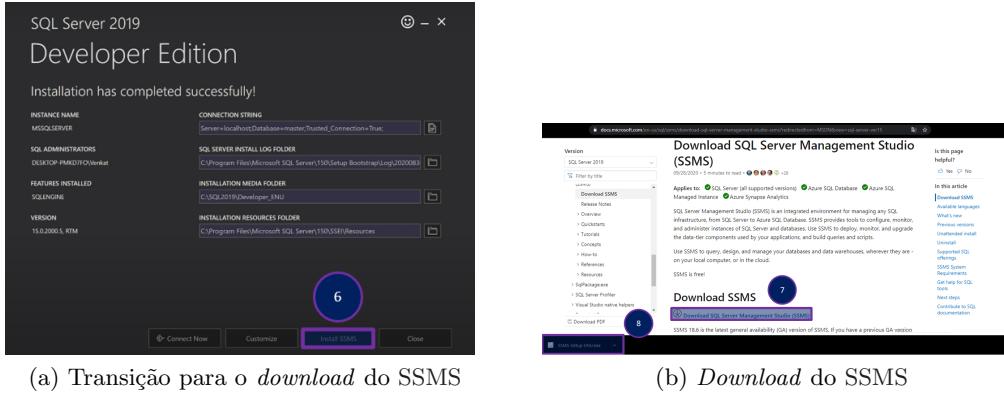


Figura A.4: *Download* e configuração do SSMS

Seguidamente, o utilizador deverá especificar o local de instalação pretendido SSMS (passo 8 - Figura A.5(a)) e realizar o caminho de inicialização do SSMS (passo 9), como pode ser visualizado na Figura A.5(b).

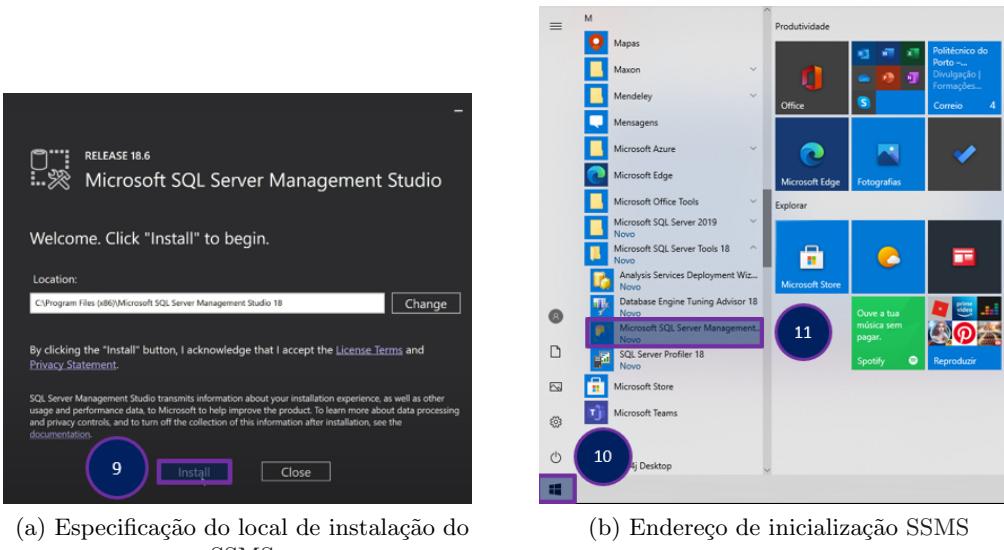


Figura A.5: Instalação e inicialização do SSMS

<sup>11</sup><https://docs.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio>

**ANEXO A. INSTALAÇÃO E CONFIGURAÇÃO DO AMBIENTE DE  
DESENVOLVIMENTO**

Finalmente, resta apenas iniciar o SSMS (passo 10 - Figura A.14(a)) e efetuar o caminho ilustrado na Figura A.14(b): Databases -> System Databases -> master -> Programmability -> Stored Procedures -> System Stored Procedures. De salientar que é recomendável verificar se o nome do servidor é homólogo ao do sistema em operação. O utilizador pode ser identificado através dos passos explicitados na Figura A.8.

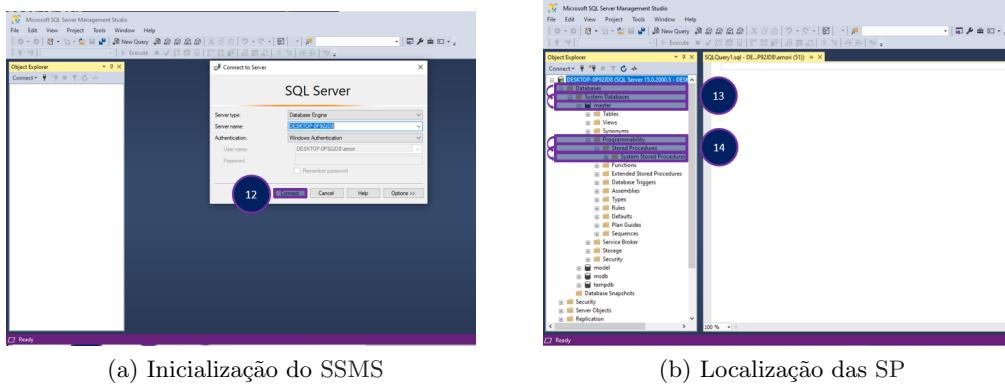


Figura A.6: Inicialização das SP

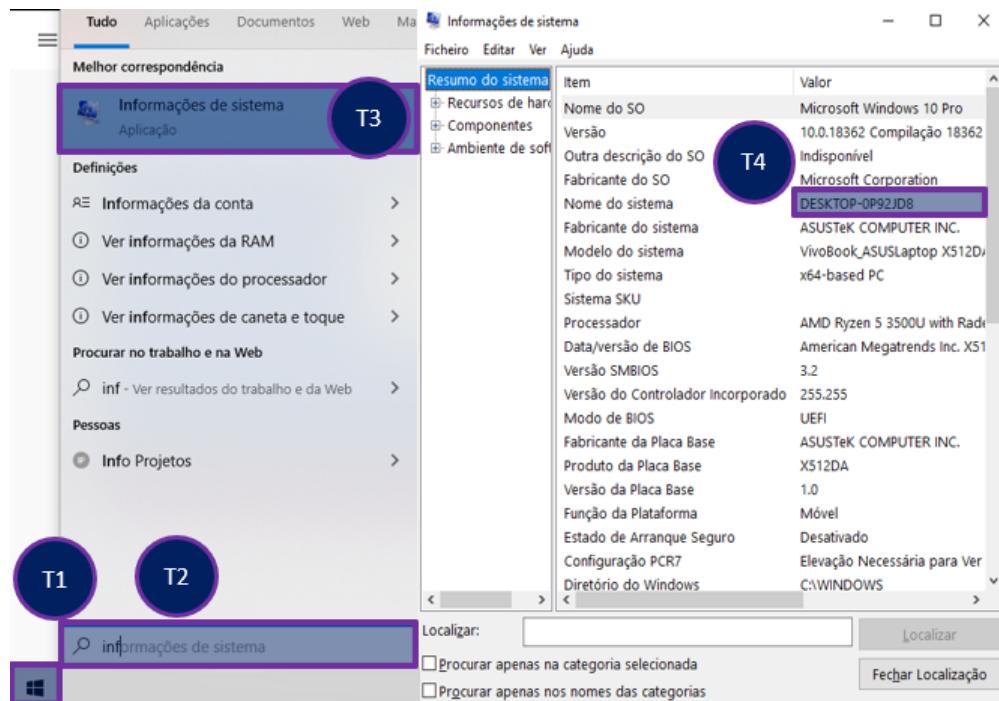


Figura A.7: Verificação do nome do sistema

## A.2 Dapper

Para o ambiente de trabalho optado (Visual Studio Code) para este projeto, o método de instalação do Dapper consiste em primeiramente selecionar a *tab* Projeto (passo 1) e de seguida optar pelo item *Manage NuGet Packages...* (passo2). Posteriormente, é necessário introduzir a palavra Dapper na barra de pesquisa (passo 3), selecionar (passo 4) e instalar (passo 5) o mesmo. Recomenda-se que o utilizador verifique se o Dapper foi bem instalado no seu projeto, através da verificação do mesmo nos pacotes do projeto (passo 6).

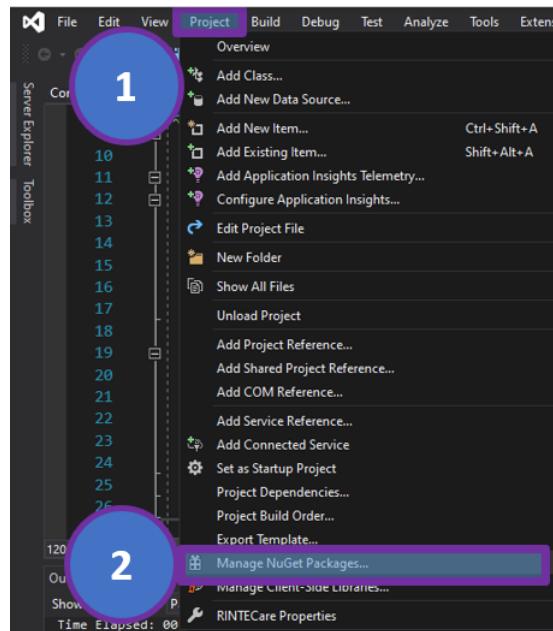


Figura A.8: Seleção do item instalador *Manage NuGet Packages...*

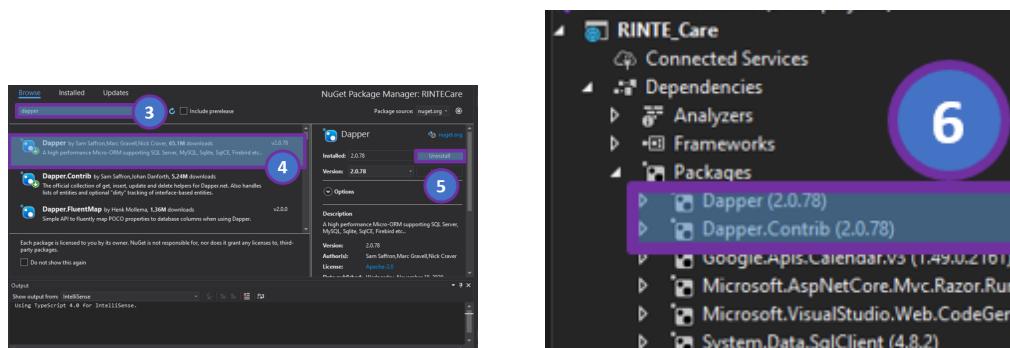


Figura A.9: Processo de instalação do Dapper

### A.3 Google Calendar API

Para começar, é necessário criar uma conta de desenvolvedor ou realizar a autenticação através do *link*<sup>12</sup>. Logo após, é necessário criar um novo projeto no *link*<sup>13</sup> (etapa 1), definir o nome do projeto (etapa 2) e seguir para o próximo passo (etapa 3), como ilustrado na Figura A.10. Posteriormente, no passo 4 recomenda-se optar pela opção de *Desktop app* e confirmar a criação do projeto (passo 5).

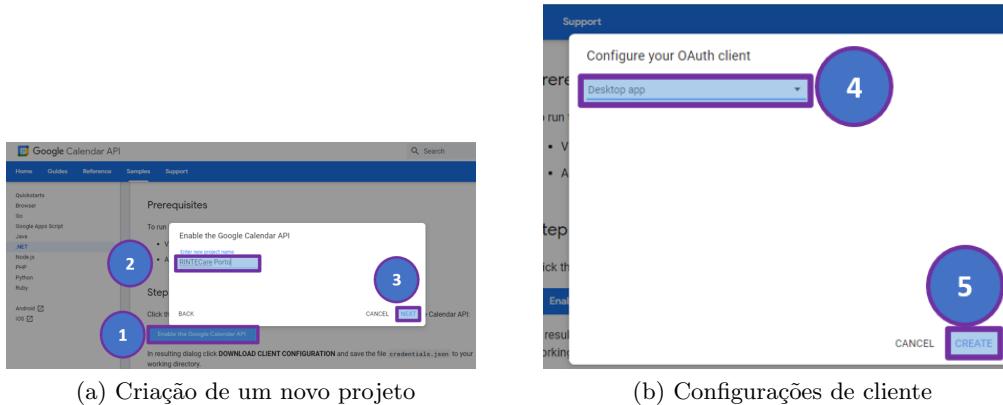


Figura A.10: Configuração de cliente

De seguida e como ilustrado na Figura A.11, é necessário descarregar as configurações de cliente (passo 6). Que de seguida necessitam de ser importadas no projeto que o utilizador deseja (passo 7).

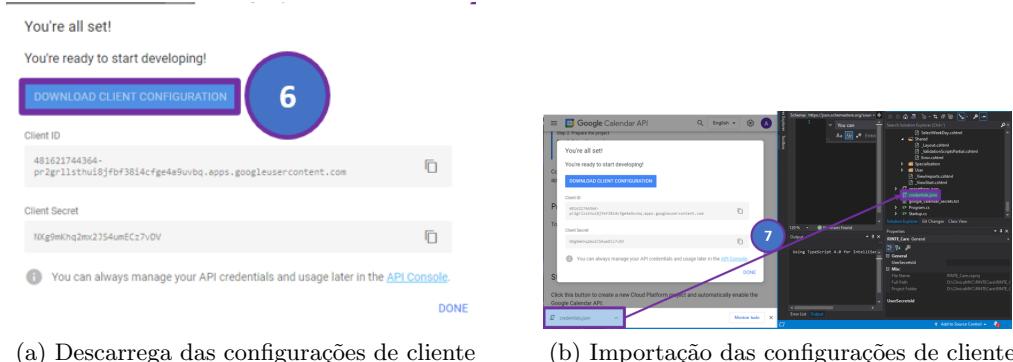


Figura A.11: Instalação das configurações de cliente

Por fim, é necessário instalar o *package* do Google Calendar API. Inicialmente, seleciona-se a opção de projeto e do *Manage NuGet Packages...* (etapa 8).

<sup>12</sup><https://developers.google.com/>

<sup>13</sup><https://developers.google.com/calendar/quickstart/dotnet>

O processo seguinte consiste em pesquisar e selecionar o Google Calendar API (etapa 9) e por fim instalar o mesmo (passo 10), como ilustrado na Figura B.12.

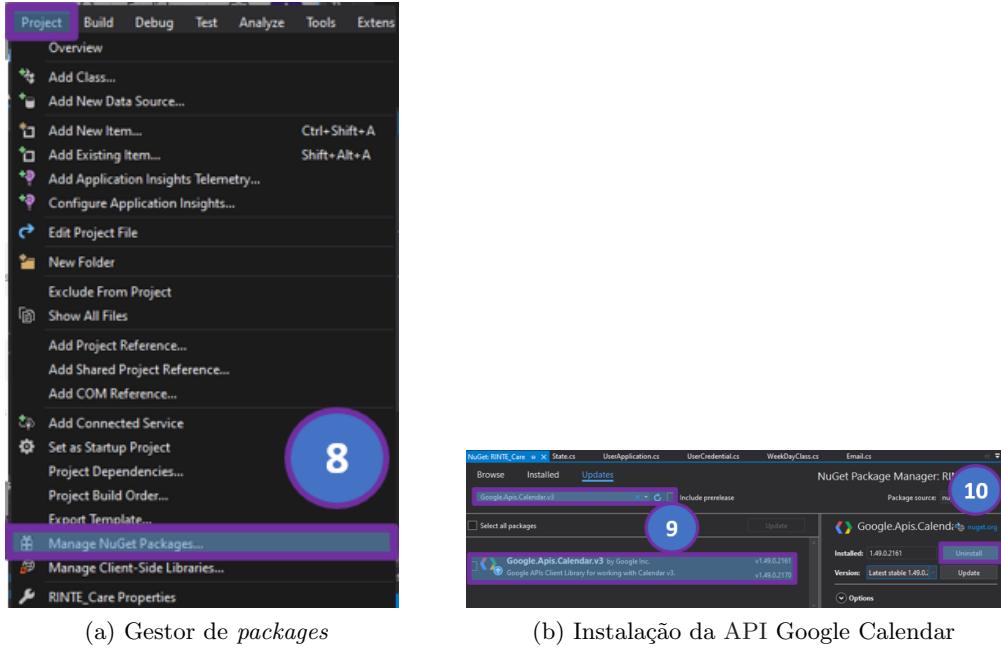


Figura A.12: Processo de Instalação da API Google Calendar

## A.4 Wkhtmltopdf

Assim como nas ferramentas anteriores, é necessário inicialmente selecionar a opção de projeto e do *Manage NuGet Packages...* (etapa 1). Sendo que de seguida é necessário pesquisar e selecionar o Wkhtmltopdf (etapa 2) e por fim instalar o mesmo (passo 3), como ilustrado na Figura B.12.



Figura A.13: Processo de Instalação do Wkhtmltopdf

## ANEXO A. INSTALAÇÃO E CONFIGURAÇÃO DO AMBIENTE DE DESENVOLVIMENTO

### A.5 ASP.NET MVC Core

De modo a criar um novo projeto em ASP.NET MVC Core, é necessário primeiramente selecionar as opções **Criar um novo projeto** (passo 1) e **Aplicação ASP.NET MVC Core** (passo 2), e prosseguir para a próxima página (passo 3), como ilustrado na Figura A.14.

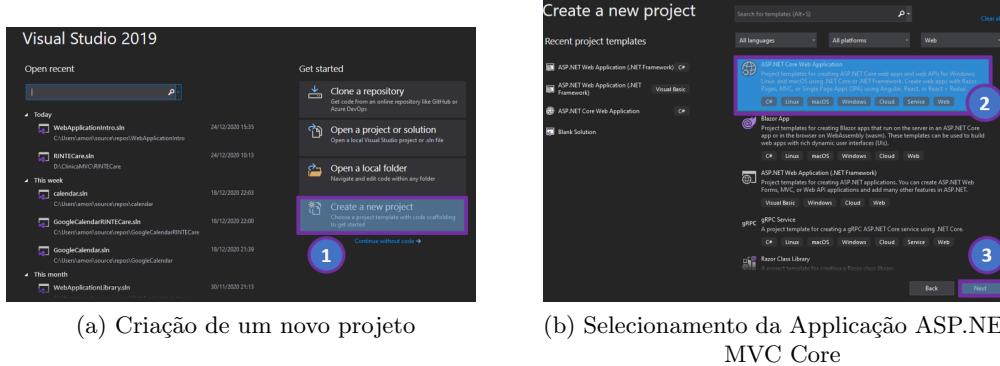


Figura A.14: Criação de um novo projeto

De seguida e como ilustrado na Figura A.15, é necessário configurar o nome do projeto, localização e afins (passo 4), e prosseguir (passo 5) para a escolha do tipo de aplicação web ASP.NET Core se deseja. Introduzindo os parâmetros **.NET Core** e **ASP.NET Core 5.0** (passo 6) e, optando pelo **ASP.NET Core Web App (Model-View-Controller)** (passo 7), bem como finalizar o processo de criação (passo 8).



Figura A.15: Configuração do novo projeto e selecionamento do tipo de ASP.NET Core Web App

De modo a averiguar o projeto, é necessário premir o botão **IIS Express** (passo 9), em que de seguida é possível observar o resultado final como na Figura A.16.

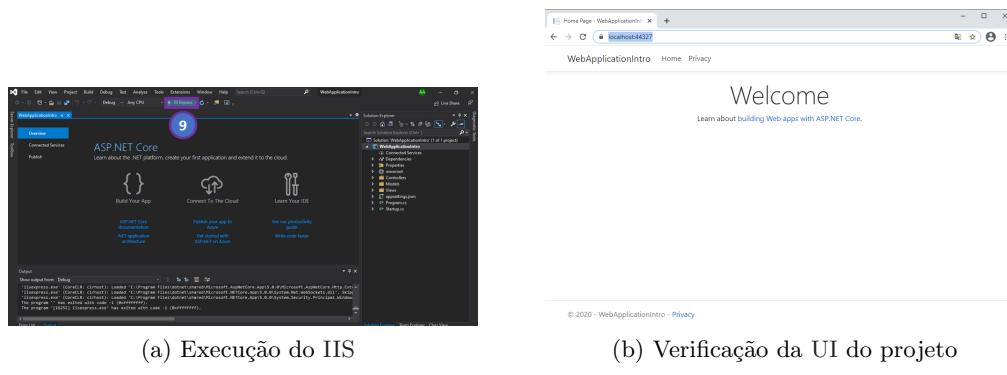


Figura A.16: Verificação do projeto



## Anexo B

---

# Complementos dos Testes Não-Funcionais

---

### B.1 Testes de Carga e de Tempo de Resposta

Através do comando de diagnóstico de rede de computador *tracert* é possível exibir possíveis rotas e medir atrasos de trânsito de pacotes numa rede *Protocolo de Internet* (IP). Sendo que deste modo foi possível obter as rotas e o tempo de trânsito do domínio da página do Livro de Reclamações. Observando-se que são necessários 7 *hops* e cerca de 200 ms para alcançar o mesmo.

```
C:\Users\amori>tracert www.livroreclamacoes.pt
Tracing route to www.livroreclamacoes.pt [193.17.0.168]
over a maximum of 30 hops:
  1    7 ms      6 ms      6 ms  192.168.1.1
  2   14 ms     16 ms     20 ms  10.27.191.254
  3   19 ms     17 ms     16 ms  10.137.211.113
  4    76 ms     24 ms     30 ms  10.255.48.78
  5    25 ms     21 ms     22 ms  186.55.30.213.rev.vodafone.pt [213.30.55.186]
  6    18 ms     20 ms     20 ms  202.25.30.213.rev.vodafone.pt [213.30.25.202]
  7    23 ms     22 ms     24 ms  148.69.178.83
  8    *         *         *       Request timed out.
```

Figura B.1: Rotas e tempo de trânsito do domínio Livro de Reclamações

Similarmente ao processo anterior, realizou-se a pesquisa e medição do trânsito de pacotes numa rede IP para o Gmail. Sendo necessários 10 *hops* e cerca de 283 ms para alcançar o mesmo.

```
C:\Users\amori> tracert mail.google.com
Tracing route to googlemail.l.google.com [142.250.184.5]
over a maximum of 30 hops:
 1  10 ms   13 ms   8 ms  192.168.1.1
 2  15 ms   13 ms   23 ms  10.27.191.254
 3  16 ms   14 ms   18 ms  10.137.211.97
 4  18 ms   23 ms   19 ms  10.255.48.78
 5  28 ms   20 ms   18 ms  72.14.202.214
 6  23 ms   22 ms   30 ms  74.125.245.98
 7  33 ms   31 ms   29 ms  142.250.239.127
 8  27 ms   32 ms   27 ms  108.170.253.225
 9  30 ms   40 ms   30 ms  142.250.214.41
10  36 ms   42 ms   27 ms  mad41s10-in-f5.1e100.net [142.250.184.5]

Trace complete.
```

Figura B.2: Rotas e tempo de trânsito do domínio Gmail

Finalmente, realizou-se este processo para a API Calendário Google. Tendo sido realizados 11 *hops* e cerca de 239 ms para alcançar o mesmo.

```
C:\Users\amori>tracert calendar.google.com
Tracing route to calendar.google.com [172.217.17.14]
over a maximum of 30 hops:
 1  7 ms   6 ms   6 ms  192.168.1.1
 2  15 ms   15 ms   16 ms  10.27.191.254
 3  16 ms   14 ms   16 ms  10.137.211.97
 4  18 ms   21 ms   19 ms  10.255.48.82
 5  20 ms   22 ms   20 ms  72.14.209.204
 6  18 ms   19 ms   17 ms  172.253.64.199
 7  19 ms   19 ms   21 ms  74.125.245.83
 8  30 ms   28 ms   36 ms  72.14.239.53
 9  28 ms   31 ms   30 ms  74.125.242.177
10  31 ms   31 ms   41 ms  74.125.253.199
11  29 ms   28 ms   29 ms  mad07s09-in-f14.1e100.net [172.217.17.14]

Trace complete.
```

Figura B.3: Rotas e tempo de trânsito do domínio da API Calendário Google

## B.2 Testes de Usabilidade

- Autenticar



(a) Etapa 1 do registo



(b) Etapa 2 do registo

Figura B.4: Processo de autenticar

- Alteração de palavra-passe

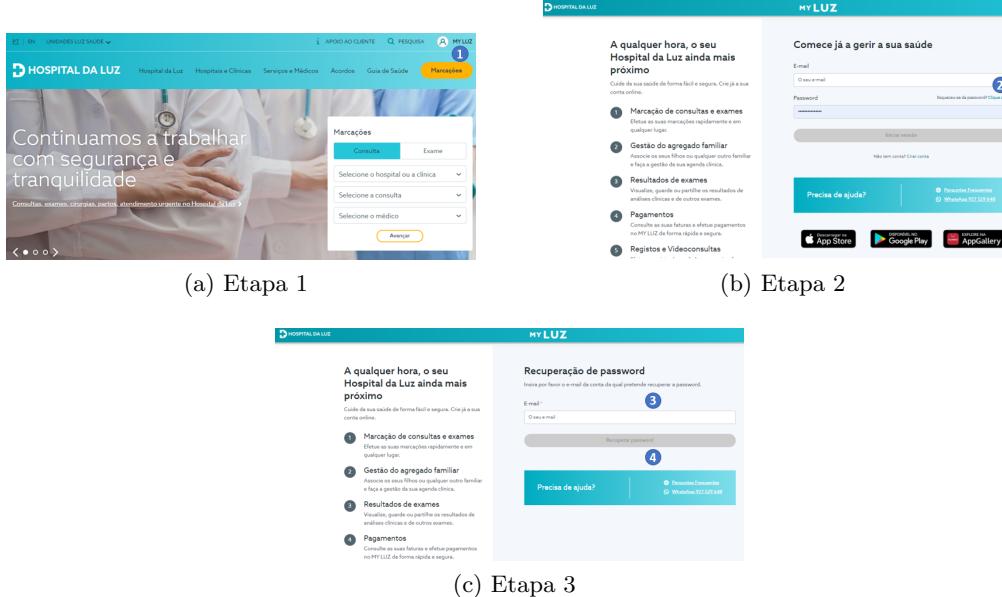


Figura B.5: Processo de alterar palavra-passe

- Cancelar consulta

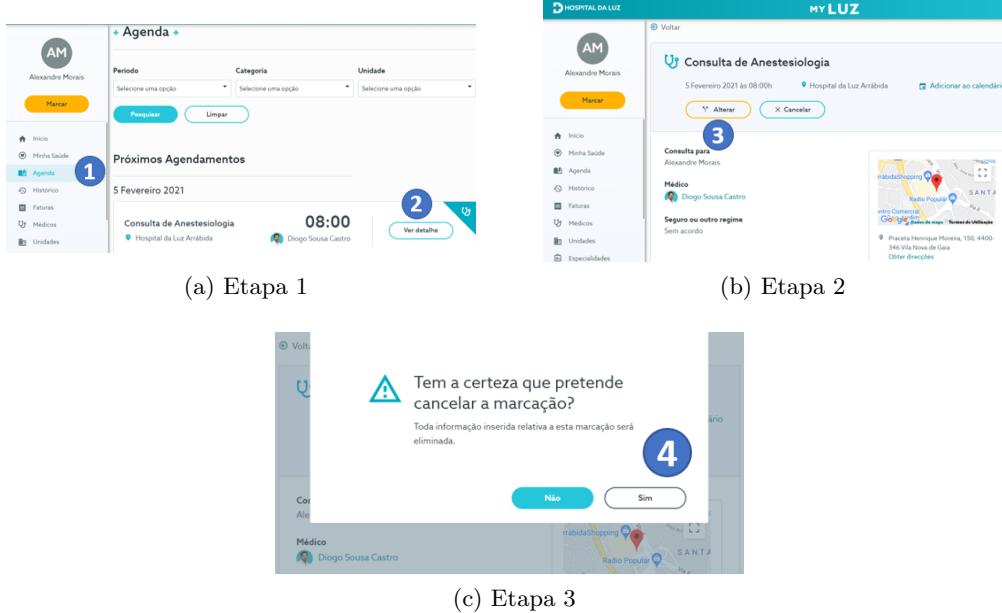


Figura B.6: Processo de cancelar consulta

- Consultar perfil

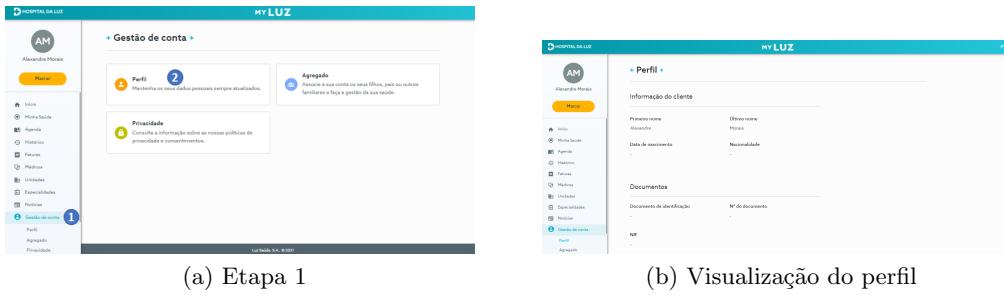


Figura B.7: Processo de consultar perfil

- Contactar assistentes

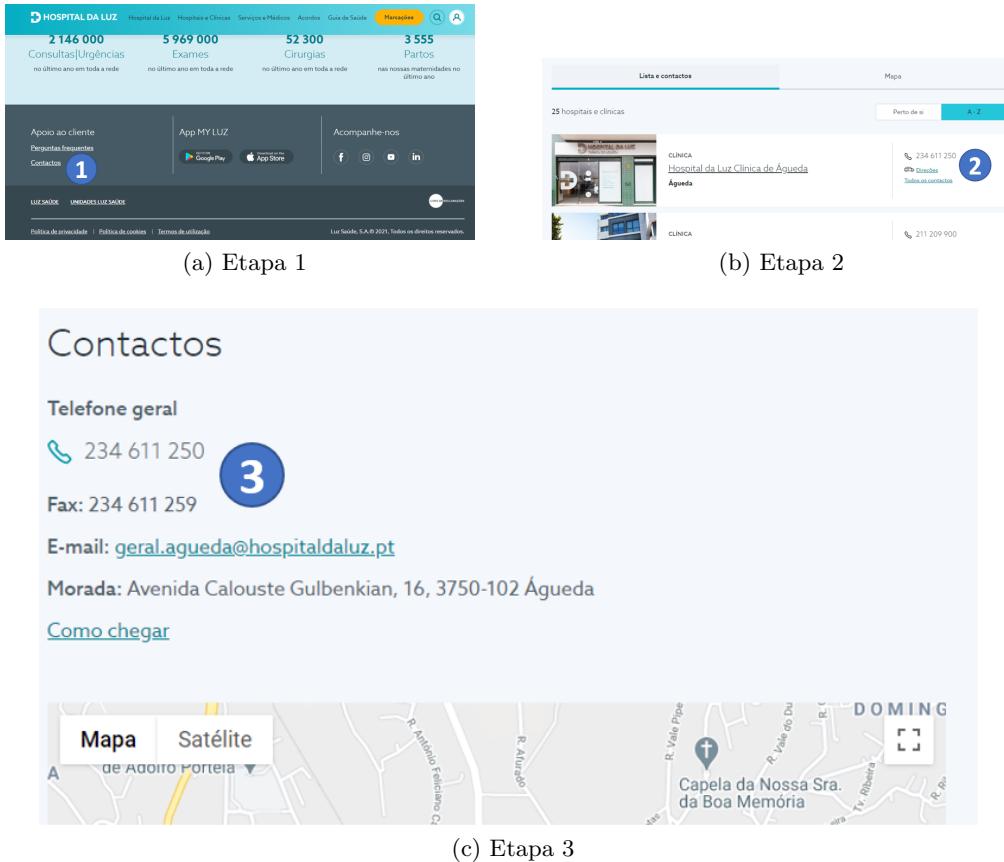


Figura B.8: Processo de contactar assistentes

- Editar consulta

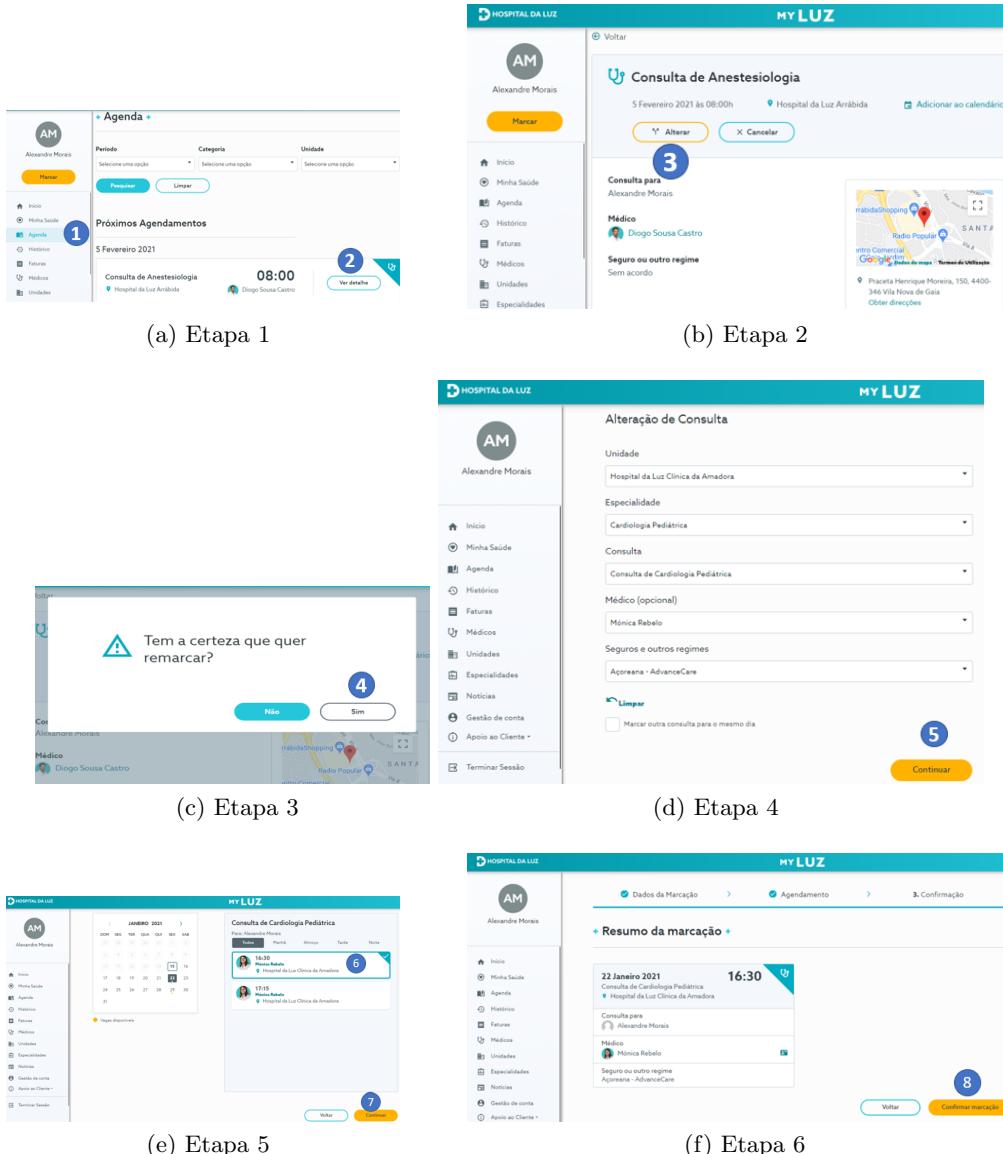


Figura B.9: Processo de editar consulta

### ● Marcação de consulta

(a) Etapa 1

(b) Etapa 2

(c) Etapa 3

Detailed description: The figure consists of three screenshots of a web application for booking a consultation at Hospital da Luz. 
 - Screenshot (a) shows the main dashboard with a sidebar for 'Minha Saúde' containing links for Agenda, Histórico, and Faturas. The main area is titled 'O que pretende marcar?' with dropdown menus for Unidade (Hospital da Luz Arábida), Especialidade (Anestesiologia), Consulta (Consulta de Anestesiologia), Médico (Diogo Souza Castro), Seguro e outros regimes (Sem acordo), and a 'Continuar' button. Numbered circles 1 through 7 point to these elements.
 - Screenshot (b) shows the 'Selecionar a data' step. It displays a calendar for February 2021 with a highlighted date (number 8). To the right, it lists 'Consulta de Anestesiologia' for 'Para: Alexandre Moraes' at '08:00' with 'Diogo Souza Castro'. Numbered circles 9 and 10 point to these details. Below the calendar is a 'Continuar' button (number 11).
 - Screenshot (c) shows the 'Resumo da marcação' step. It summarizes the booking: '5 Fevereiro 2021 08:00', 'Consulta de Anestesiologia', 'Hospital da Luz Arábida', 'Consultor para Alexandre Moraes', 'Médico Diogo Souza Castro', and 'Seguro ou outro regime Sem acordo'. It includes a 'Voltar' button and a prominent 'Confirmar marcação' button (number 12).

Figura B.10: Processo de marcar uma consulta

### ● Registo

(a) Etapa 1

(b) Etapa 2

(c) Etapa 3

Detailed description: The figure consists of three screenshots of a web application for user registration at Hospital da Luz. 
 - Screenshot (a) shows the homepage with a banner about safety and a 'Marcções' (bookings) sidebar. Numbered circles 1 through 4 point to the sidebar fields: 'Selecione o hospital ou a clínica' (Hospital da Luz), 'Selecione a consulta' (Consulta de exames), 'Selecione o médico' (Diogo Souza Castro), and a 'Avançar' button.
 - Screenshot (b) shows the 'Comece já a gerir a sua saúde' step. It asks for personal information: 'Nome' (Nome completo), 'Sexo', 'Nacionalidade', 'Documento de identificação', 'Número de documento', 'E-mail', and 'Password'. Numbered circles 5 through 12 point to these fields.
 - Screenshot (c) shows the 'Preencha os seus dados' step. It continues with 'Nome', 'Sexo', 'Nacionalidade', 'Documento de identificação', 'Número de documento', 'E-mail', and 'Password'. It also includes fields for 'Teléfono' (with 'IHM' checkbox), 'Unidade profissional' (with 'Especialidade' dropdown), and a 'Declaração' checkbox. Numbered circles 13 through 15 point to these additional fields.

Figura B.11: Processo de registrar

- Visualizar detalhes de consulta

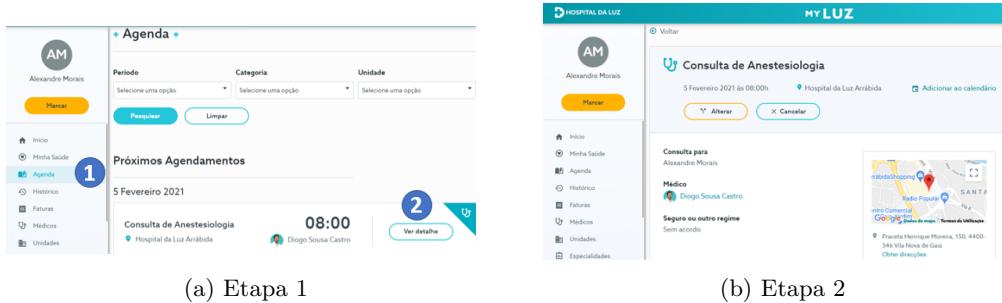


Figura B.12: Processo de consultar detalhes da consulta

