



# **Gestão de Clínica de Enfermagem – Siltes Saúde**

Licenciatura em Engenharia Informatica

Ana Beatriz Nunes Ribeiro, nº 216096

Andreia Filipa Gomes Fernandes, nº 2160844

Leiria, julho de 2020





# **Gestão de Clínica de Enfermagem – Siltes Saúde**

Licenciatura em Engenharia Informatica

Ana Beatriz Nunes Ribeiro, nº 216096

Andreia Filipa Gomes Fernandes, nº 2160844

Trabalho de Projeto da unidade curricular de Projeto Informático realizada/o sob a orientação do(a) Professor(a) Doutor(a) Paulo Jorge Ferreira Batista Pinheiro Cordeiro.

Leiria, julho de 2020

# **Dedicatória**

Dedicamos este projeto a todos os que sempre nos apoiaram e sempre acreditaram em nós ao longo deste percurso.

# Agradecimentos

Gostaríamos de aproveitar esta oportunidade para agradecer a todas as pessoas que de alguma forma, direta ou indiretamente, contribuíram para a realização deste projeto e das quais tentaremos enunciar todas elas.

Queremos agradecer, em primeiro lugar ao professor Paulo Cordeiro, por ter aceite ser nosso orientador, mesmo sendo uma autoproposta, pelo tempo dispensado no apoio ao projeto e na resolução de problemas nas várias etapas de desenvolvimento.

Ao professor Rui Oliveira, pela ajuda na definição da Base de Dados.

À professora Marisa Maximiano, pela ajuda com a manipulação das imagens.

Ao Sr. José Chambel, administrador da clínica de enfermagem, que aceitou a parceria com a ESTG para realizarmos o projeto, e que sempre se mostrou disponível para esclarecimentos e reuniões mensais.

# Resumo

Este relatório descreve na íntegra o desenvolvimento da autoproposta do projeto Gestão de Clínica de Enfermagem – Siltes Saúde e visa explicar em detalhe todas as decisões tomadas ao longo do desenvolvimento do mesmo e ainda tudo o que foi implementado e porque foi feito dessa forma.

Visto que a clínica Siltes Saúde não tinha qualquer sistema informático para fazer a gestão da mesma e depois de uma breve conversa com a administração, foi neste contexto que nasceu este projeto. Contendo a intenção de solucionar muitos dos problemas que tanto a administração e os enfermeiros têm presentemente para realizar a gestão da clínica e dos seus utentes.

As funcionalidades disponibilizadas para este projeto, como utilizador do tipo Administrador incluem a recuperação de *password*, recuperação de nome de utilizador, gestão de fornecedores, encomendas e produtos. É também possível, registar enfermeiros na plataforma e editar os seus dados, e gerir as despesas da clínica.

As funcionalidades disponibilizadas para este projeto, como utilizador do tipo Normal (Enfermeiro, sem privilégios de administração) incluem registar utentes e alterar os seus dados pessoais na plataforma. Fazer o registo de uma consulta, registar várias doenças, alergias, cirurgias, tipos de aleitamento, dados de análises laboratoriais efetuadas pelo utente, dados de exames efetuados pelo utente, métodos contraceptivos, profissões, tipos de parto, tipos de queimadura e tipos de úlcera, guardar o histórico clínico de cada utente, marcar consultas, desmarcar consultas e alterar consultas. Registar os dados da avaliação objetiva, e ver os dados registados nas consultas anteriores, dados dos tratamentos. O enfermeiro pode prescrever uma medicação e imprimir essa mesma medicação, registar várias atitudes terapêuticas e os produtos que usa numa consulta.

A metodologia adotada pelo grupo para desenvolvimento foi uma metodologia ágil. Não se pode afirmar que seguimos à risca o *Scrum* (devido aos atores e eventos que caracterizam o *Scrum*), contudo baseámo-nos principalmente neste estilo de desenvolvimento de software porque o nosso projeto não é académico onde os requisitos são tipicamente todos impostos no arranque do projeto. Temos um cliente real com o qual temos reuniões mensais e reuniões

semanais com o nosso orientador, o que previsivelmente nos iria fazer oscilar os requisitos ao longo do tempo.

Na primeira fase de desenvolvimento definimos cinco *milestones* onde pretendíamos fazer uma análise da evolução e do cumprimento dos requisitos junto com o orientador e o cliente. Estimou-se um período de 1 semana para cada um dos *sprints* e a cada duas destas fases pretendíamos fazer uma *release* para o cliente poder aprovar ou corrigir alguns dos requisitos impostos à cabeça.

Antes, durante e depois da fase de desenvolvimento fomos tendo reuniões com a administração da clínica.

O corpo de desenvolvimento do projeto divide-se principalmente em duas áreas chave. A primeira é uma aplicação desktop que funciona como interface gráfica. A segunda é a base de dados onde são armazenados todos os dados.

Depois do projeto a funcionar, executámos vários testes de usabilidade para garantirmos que a aplicação estava segura e de acordo com o que o cliente havia requisitado.

Por fim, efetuámos o *deployment* do nosso projeto.

Sucintamente, este projeto foi desenvolvido recorrendo maioritariamente a tecnologia Microsoft e SQL, nomeadamente C# e SQL.

**Palavras-chave:** C#, SQL, *Scrum*, Aplicação *Desktop*, Gestão Clínica, Enfermagem

# Abstract

This report describes the complete development of the clinical nursing clinic management project - Saúde Siltes and intends to explain in detail all decisions such as those taken during its development, everything that was implemented and why it was done in this way.

Since the Siltes Saúde clinic had no management system to handle it and after a brief conversation with the administration, this project was born with the intention to solve many problems that both management and nurses currently have to manage in the clinic and with its patients.

The services available for this project as Administrator type user are, password recovery, username recovery, supplier management, orders and products, it is also possible to register nurses on the platform and edit data, view statistical data, manage clinic expenses.

The functionalities available for this project, as a Nurse user are, record and change the platform data, register a consultation, register various diseases, allergies, surgeries, save the clinical history of each user, consultation, clear appointments and change queries.

The methodology adopted by the development group was Agile methodology. You cannot declare the following items in Scrum (due to the actors and events that characterize Scrum), but it is mainly based on the style of software development because our project is not academic where the requirements are typically set on the beginning of the project. We have a real client with whom we have weekly gatherings and also meetings with our advisor, that would probably make us oscillate the requirements during development of the project.

In the first phase of development, we defined five milestones where was intended to make an analysis of the evolution and fulfil of the requirements with the advisor and the client. Was estimated a period of 1 week for each of the sprints and at each set of two sprints was intended to make a release to the customer who could approve or correct some of the requirements settled.

Before, during and after the development phase, we had meetings with the clinic administration. It is possible to find information describing each one of them in the annex of this document.



The project's development body is mainly divided into two key areas. The first is a desktop application that works as a graphical interface. The second is the database where all data is stored.

After the development of the software, we ran several tests to ensure that the application was robust, safe and was complying to what was requested by client.

Finally, we deployed our project.

This project was successfully developed using mainly Microsoft and SQL technology, especially C # and SQL.

**Keywords:** C #, SQL, Scrum, Desktop Application, Clinical Management, Nursing

# Lista de Figuras

Figura 1 – <i>Mockup</i> do Registrar Utilizador .....	7
Figura 2 – Esquema de Parte da Base de Dados .....	13
Figura 3 – Arquitetura do Sistema .....	14
Figura 4 – <i>mockup</i> das despesas e receitas.....	61
Figura 5 – <i>mockup</i> do registo dos Utentes.....	61
Figura 6 - tabela AdministrarMedicacao .....	64
Figura 7 - tabela AgendamentoConsulta .....	65
Figura 8 - tabela Aleitamento.....	65
Figura 9 - tabela Alergia.....	66
Figura 10 – tabela AlergiaPaciente .....	67
Figura 11 - tabela Algaliacao .....	68
Figura 12- tabela analisesLaboratoriais.....	69
Figura 13 - tabela analisesLaboratoriaisPaciente .....	70
Figura 14 – tabela AspiracaoSecrecao .....	71
Figura 15 - tabela Atitude.....	72
Figura 16 - tabela AvaliacaoObjetivo .....	74
Figura 17 - tabela AvaliacaoObjetivoBebe .....	77
Figura 18 - tabela Cateterismo .....	78
Figura 19 - tabela Cirurgia .....	79
Figura 20 – CirurgiaPaciente.....	80
Figura 21 - tabela ColheidadeSanguePrecoce .....	81
Figura 22 - tabela ColheitaUrina .....	82
Figura 23 - tabela ColocacaoDIU.....	83
Figura 24 - tabela Colpocitologia.....	85
Figura 25 - tabela Consulta .....	87
Figura 26 - tabela ConsultaProdutoStock.....	88
Figura 27 - tabela Crioterapia.....	89

Figura 28 - tabela Desbridamento .....	90
Figura 29 - tabela Despesa .....	91
Figura 30 - tabela Doenca .....	92
Figura 31 - tabela DoencaPaciente.....	93
Figura 32 - tabela DopplerFetal .....	94
Figura 33 - tabela DrenagemLocas .....	95
Figura 34 - tabela Encomenda.....	96
Figura 35 - tabela Enfermeiro .....	97
Figura 36 - tabela ENG .....	98
Figura 37 - tabela Espirometria .....	100
Figura 38 - tabela Exame .....	101
Figura 39 - tabela Flebografia .....	102
Figura 40 - tabela Fornecedor .....	104
Figura 41 - tabela ImplanteContracetivo .....	105
Figura 42 - tabela Inalacoes .....	106
Figura 43 - tabela LavagemAuricular .....	107
Figura 44 - tabela LavagemOcular.....	108
Figura 45 - tabela LavagemVesical.....	109
Figura 46 - tabela LinhaEncomenda .....	110
Figura 47 - tabela LocalizacaoDor .....	111
Figura 48 - tabela LocalizacaoDorConsulta .....	112
Figura 49 - tabela LocalizacaoDorDopplerArterialVenoso.....	113
Figura 50 - tabela LocalizacaoDorDopplerFetal .....	114
Figura 51 - tabela LocalizacaoDorEspisiotomia .....	115
Figura 52 – tabela LocalizacaoDorTratamento .....	116
Figura 53 - tabela LocalizacaoDorVacinaçao .....	117
Figura 54 - tabela Medicacao .....	119
Figura 55 - tabela MetodoContracetivo.....	120
Figura 56 - tabela MonitorizacaoECG .....	121
Figura 57 - tabela Paciente .....	124

Figura 58 - tabela Parto .....	125
Figura 59 - tabela Pressoterapia .....	126
Figura 60 - tabela ProdutoStock.....	127
Figura 61 - tabela Profissao.....	127
Figura 62 - tabela Suturas.....	128
Figura 63 - tabela TesteAcuidadeVisual .....	129
Figura 64 - tabela TesteCombur .....	131
Figura 65 - tabela tipoDespesa .....	132
Figura 66 - tabela tipoExame .....	132
Figura 67 - tabela tipoQueimadura.....	133
Figura 68 - tabela tipoUlcera.....	133
Figura 69 - tabela Tratamento .....	134
Figura 70 - tabela TratamentoExcisoos .....	135
Figura 71 - tabela TratamentoMaosPes.....	136
Figura 72 - tabela TratamentoPaciente.....	138
Figura 73 - tabela Tricotomia.....	139
Figura 74 - tabela Vacinacao.....	140
Figura 75 - tabela VariasAtitudes.....	141
Figura 76 - tabela ZaragatoaOrofaringe .....	142

# Lista de tabelas

Tabela 1 – Requisitos Destacados .....	5
Tabela 2 – Requisitos Não Funcionais .....	6
Tabela 3 – requisitos de sistema.....	15
Tabela 4 – Cronograma do Projeto Informático.....	17
Tabela 5 – Requisitos Funcionais do Administrador.....	44
Tabela 6 - Requisitos Funcionais do Administrador e Utilizador Normal .....	60

## Lista de siglas e acrónimos

C#	C Sharp
DEI	Departamento de Engenharia Informática
ESTG	Escola Superior de Tecnologia e Gestão
GH	GitHub
IPL	Instituto Politécnico de Leiria
SGBD	Sistema de Gestão de Base de Dados
SNS	Sistema Nacional de Saúde
SQL	Structured Query Language
VS	Visual Studio

# Índice

<b>Dedicatória .....</b>	<b>ii</b>
<b>Agradecimentos.....</b>	<b>iii</b>
<b>Resumo.....</b>	<b>iv</b>
<b>Abstract .....</b>	<b>vi</b>
<b>Lista de Figuras .....</b>	<b>viii</b>
<b>Lista de tabelas.....</b>	<b>xi</b>
<b>Lista de siglas e acrónimos.....</b>	<b>xii</b>
<b>1. Introdução.....</b>	<b>1</b>
<b>1.1. Motivação .....</b>	<b>1</b>
<b>1.2. Objetivos .....</b>	<b>2</b>
<b>1.3. Estrutura do Relatório.....</b>	<b>3</b>
<b>2. Requisitos .....</b>	<b>5</b>
<b>2.1. Requisitos Funcionais .....</b>	<b>5</b>
<b>2.2. Requisitos Não Funcionais.....</b>	<b>6</b>
<b>3. Mockups .....</b>	<b>7</b>
<b>4. Metodologias .....</b>	<b>9</b>
<b>5. Tecnologias Usadas .....</b>	<b>11</b>
<b>6. Base de Dados .....</b>	<b>13</b>
<b>7. Arquitetura do Sistema.....</b>	<b>14</b>
<b>8. Requisitos Mínimos de Sistema.....</b>	<b>15</b>
<b>9. Implementação do Projeto.....</b>	<b>17</b>
<b>9.1. Planeamento (17/02/2020 a 24/02/2020).....</b>	<b>17</b>

9.2.	Implementação - (25/02/2020 a 02/03/2020) .....	18
9.3.	Implementação - (03/03/2020 a 09/03/2020) .....	18
9.4.	Implementação - (10/03/2020 a 16/03/2020) .....	18
9.5.	Implementação - (17/03/2020 a 24/03/2020) .....	19
9.6.	Implementação - (25/03/2020 a 31/03/2020) .....	19
9.7.	Implementação - (01/04/2020 a 07/04/2020) .....	20
9.8.	Implementação - (08/04/2020 a 14/04/2020) .....	22
9.9.	Implementação - (15/04/2020 a 21/04/2020) .....	23
9.10.	Implementação - (22/04/2020 a 28/04/2020) .....	24
9.11.	Implementação - (29/04/2020 a 05/05/2020) .....	24
9.12.	Implementação - (06/05/2020 a 12/05/2020) .....	25
9.13.	Implementação - (13/05/2020 a 19/05/2020) .....	25
9.14.	Implementação - (20/05/2020 a 26/05/2020) .....	27
9.15.	Implementação - (02/05/2020 a 02/06/2020) .....	27
9.16.	Implementação - (03/06/2020 a 09/06/2020) .....	27
9.17.	Implementação - (10/06/2020 a 16/06/2020) .....	28
9.18.	Implementação - (17/06/2020 a 23/06/2020) .....	29
9.19.	Implementação - (24/06/2020 a 30/06/2020) .....	30
9.20.	Implementação - (01/07/2020 a 10/07/2020) .....	30
10.	Erros de desenvolvimento ao longo do projeto.....	31
11.	Testes .....	35
11.1.	Testes de Usabilidade .....	35



<b>11.2.</b>	<b>Testes de Aceitação.....</b>	<b>35</b>
<b>11.3.</b>	<b>Testes de Sistema.....</b>	<b>36</b>
<b>12.</b>	<b>Conclusão .....</b>	<b>37</b>
<b>13.</b>	<b>Trabalho Futuro.....</b>	<b>38</b>



# 1. Introdução

Devido à inexistência de um sistema de gestão da clínica, onde tudo era registado em papel, surgiu a ideia de desenvolver uma solução mais acessível, cómoda e fiável de efetuar a gestão da mesma.

O projeto Gestão de Clínica de Enfermagem – Siltes Saúde enquadra-se na unidade curricular de Projeto Informático do curso de Engenharia Informática do Departamento de Engenharia Informática (DEI) da Escola Superior de Tecnologia e Gestão (ESTG) do Instituto Politécnico de Leiria (IPL).

Dadas as possibilidades de conseguirmos colmatar as lacunas que o administrador e enfermeiros sentem, decidimos implementar este projeto. Aliado a isto, está também todo o conhecimento potencial que esta implementação nos poderá trazer, o que será sem dúvida uma vantagem para o nosso futuro profissional.

Este projeto começou a ser desenvolvido em fevereiro do ano de 2020 e dado que se pretende colocar esta aplicação em produção, não se pode dizer que tenha uma data de término porque o produto estará em constante evolução. No entanto a última *release* no âmbito desta unidade curricular será entregue a 13 de julho de 2020.

Este projeto é constituído por uma aplicação *desktop*, que é a interface gráfica e uma base de dados.

## 1.1. Motivação

Observando o rústico método de registo de pacientes e os seus dados, que é tudo feito manualmente, a administração e enfermeiros da clínica têm manifestado algumas dificuldades na organização dos seus utentes. A principal motivação é otimizar o tempo da consulta, reduzindo o tempo de procura e inserção de dados dos utentes.

Por fim, existe o ponto de vista reputacional, porque sendo a clínica Siltes Saúde uma clínica inovadora, não lhe fica bem registar tudo em papel.

## 1.2.Objetivos

Os objetivos gerais deste projeto são otimizar o tempo de consulta, reduzir o tempo de procura dos dados das consultas dos utentes e reduzir o tempo de registos dos dados durante a consulta.

A aplicação desenvolvida tem dois perfis de utilizador, o utilizador do tipo Administrador e o utilizador do tipo normal.

Podem-se dividir as tarefas de um administrador em três grandes grupos:

Em primeiro lugar, permite que o administrador possa realizar a gestão de enfermeiros.

Em segundo lugar, o administrador poderá gerir produtos, *stocks* e encomendas através da aplicação.

Por fim, a aplicação permite que o administrador possa gerir as despesas da clínica.

Esta aplicação potenciará as seguintes vantagens para o enfermeiro, que é o utilizador do tipo normal:

Em primeiro lugar, esta aplicação permite que o enfermeiro possa gerir os seus utentes de uma forma mais eficaz. Esta funcionalidade vai permitir que o enfermeiro otimize o tempo da consulta, evitando assim de registar toda a informação manualmente e procurar os utentes em dossiês, como era feito até agora.

Em segundo lugar, o enfermeiro tem capacidade para marcar, adiar e desmarcar consultas dos seus utentes.

Em último lugar, permite registar todo o histórico clínico de cada utente.

Além destes objetivos, este projeto também tem objetivos a nível pessoal. Para nós estudantes que desenvolvemos esta aplicação, o principal objetivo passa por poder desenvolver um projeto que possamos apresentar a uma entidade empregadora, como sendo uma referência no nosso curriculum vitae.

### **1.3. Estrutura do Relatório**

O presente relatório é composto por 13 capítulos, tendo a seguinte estrutura:

- Capítulo 1: Introdução
- Capítulo 2: Requisitos
- Capítulo 3: *Mockups*
- Capítulo 4: Metodologias
- Capítulo 5: Tecnologias Usadas
- Capítulo 6: Base de Dados
- Capítulo 7: Arquitetura de Sistema
- Capítulo 8: Requisitos Mínimos de Sistema
- Capítulo 9: Implementação de Projeto
- Capítulo 10: Erros de Desenvolvimento ao longo do Projeto
- Capítulo 11: Testes
- Capítulo 12: Conclusão
- Capítulo 13: Trabalho Futuro



## 2. Requisitos

Os requisitos necessários para este projeto foram obtidos por duas fases: primeiro, foram obtidos na primeira reunião realizada com o administrador da clínica. A segunda parte dos requisitos foram obtidos a meio da implementação do projeto com o administrador da clínica e com dois enfermeiros. Os requisitos foram ajustados de acordo com o que seria possível fazer. A lista final resultou em 191 requisitos que podem ser consultados no [Anexo A](#), para cada tipo de utilizador.

De seguida na Tabela 1 são destacados apenas alguns dos requisitos funcionais.

### 2.1.Requisitos Funcionais

Designação	Caso de Uso	Descrição	Prioridade
US1	Registar Enfermeiro	A aplicação deverá permitir registar enfermeiros.	Alta
US2	Criar Listagem de Enfermeiros Registados	A aplicação deverá permitir listar todos os enfermeiros registados.	Alta
US18	Registar Utente	A aplicação deverá permitir que um enfermeiro registe todos os dados de um utente.	Alta
US20	Criar Listagem dos Utentes	A aplicação deverá permitir listar todos os utentes daquele enfermeiro.	Alta
US24	Marcar Consulta	A aplicação deverá permitir efetuar a marcação de consultas, numa data e numa determinada hora.	Alta
US52	Registar Análises Laboratoriais efetuadas pelo utente	A aplicação deverá permitir registar análises laboratoriais efetuadas pelo utente, a data em que foram feitas, resultados e observações.	Média
US81	Imprimir a medicação prescrita	A aplicação deverá permitir imprimir a medicação prescrita pelo enfermeiro.	Média

**Tabela 1 – Requisitos Destacados**

## 2.2. Requisitos Não Funcionais

Requisito R1: O sistema tem de ser fácil e intuitivo de usar.

Requisito R2: No sistema tem de fácil de navegar entre as diferentes janelas.

Requisito R3: O sistema tem de ser fácil de usar sem explicações.

Requisito R4: O *design* da aplicação tem de ser atrativo.

Requisito R5: O sistema tem de correr no Sistema Operativo *Windows*.

Requisito R6: O sistema tem de ser para *desktop*.

Requisito R7: O sistema tem de estar sempre disponível.

Requisito R8: O sistema tem de ser seguro.

Na Tabela 2 são apresentados os requisitos não funcionais do projeto.

Designação	Requisito
<b>R1</b>	Usabilidade fácil do sistema.
<b>R2</b>	Facilidade de navegação.
<b>R3</b>	Facilidade de uso (aprendizagem do programa e navegação).
<b>R4</b>	<i>Design</i> da aplicação.
<b>R5</b>	Correr em <i>Windows</i> .
<b>R6</b>	Aplicação <i>Desktop</i> .
<b>R7</b>	Disponibilidade do sistema.
<b>R8</b>	Confiabilidade.

**Tabela 2 – Requisitos Não Funcionais**



### 3. *Mockups*

Foram desenvolvidos os *mockups* em papel, com o objetivo de tomarmos algumas decisões a nível de interface gráfica. Um dos principais motivos de realizarmos a prototipagem da aplicação foi para que o cliente pudesse ter uma perceção de como a aplicação iria ficar, e ser mais fácil ajustar detalhes estéticos e garantir que a aplicação “funcionava” como era pretendido. Um dos objetivos desta tarefa também foi para podermos validar os requisitos e garantir não faltava nada.

Temos alguns *mockups* que foram realizados durante o planeamento do projeto que estão no [Anexo B](#).

Este é um exemplo de um *mockup* que fizemos que pertence ao requisito de registar um utilizador no sistema. Neste *mockup* pode-se ver os diferentes campos que achamos necessários para este requisito. Esses campos são, nome do utilizador, a função que desempenha na clínica, data de nascimento, contacto telefónico, *password*, campo para confirmar a password, e um campo para definir o *username*. Existe a opção para registar o utilizador e outra para cancelar em que fecha a janela não registando os dados inseridos.

Hand-drawn mockup of a "Registar Utilizador" window. The window has a title bar "Siltes Saúde - Registar Utilizador". Inside, there are input fields for "nome completo:", "função:", "Data Nascimento:", "contacto:", "password", "confirmar password", and "username". At the bottom, there are two buttons: "✓ Guardar" and "✗ Cancelar".

**Figura 1 – Mockup do Registar Utilizador**



## 4. Metodologias

Foram estudadas as diferentes metodologias existentes e que foram abordadas em algumas unidades curriculares do nosso curso. Estudamos a metodologia em cascata que consiste numa metodologia mais tradicional. Esta metodologia é uma abordagem linear que conta com uma sequência de acontecimentos. Esses acontecimentos são, reunir e documentar os requisitos, desenhar o sistema, desenvolver o código e testes unitários, realizar testes de sistema, executar testes de aceitação do utilizador, corrigir problemas e entregar o produto totalmente acabado. <sup>[1][2][3][4][5][6]</sup>

Nesta metodologia são revistos todos os requisitos pedidos pelo cliente e aprovados pelo mesmo. O projeto não avança se existirem requisitos pendentes, pois esta metodologia não aceita bem novos requisitos no meio do desenvolvimento. Uma das grandes desvantagens da metodologia em cascata é que existe uma grande possibilidade de o cliente ficar insatisfeito com o resultado final, pois como o sistema é baseado em requisitos documentados, o cliente não tem a perspetiva de como vai ficar o resultado final do projeto. E no fim do sistema estar implementado, é muito difícil realizar alterações. Os outros tipos de metodologias existentes são as metodologias ágeis. <sup>[1][2][3][4][5][6]</sup>

Este tipo de metodologia é uma abordagem iterativa do processo de desenvolvimento. O tempo de desenvolvimento é dividido em “*sprints*”, em que cada “*sprint*” tem uma duração de tempo definida e tem um conjunto de *User Stories* definidos também. À medida que o projeto avança todo o trabalho é revisto pela equipa de desenvolvimento e pelo cliente. Utilizando uma metodologia ágil o cliente tem a oportunidade de poder solicitar alterações ao longo do projeto. Escolhemos as metodologias que mais se enquadravam no nosso projeto que são o *Scrum* e o *Extreme Programming*. Decidimos que iríamos seguir 2 metodologias ágeis diferentes que são o *Scrum* e o *Extreme Programming*. <sup>[1][2][3][4][5][6]</sup>

Sendo uma autoproposta e um projeto real, não fazia qualquer sentido não escolher metodologias ágeis, porque é um tipo de projeto muito propício a alterações. Estas 2 metodologias são muito semelhantes em termos de valores de desenvolvimento de

projetos, pois são as mais adequadas quando existem alterações no decorrer do processo de desenvolvimento. <sup>[1][2][3][4][5][6]</sup>

## 5. Tecnologias Usadas

Por fim, decidimos quais as tecnologias a usar para a realização deste projeto.

A tecnologia escolhida foi a linguagem C# (*C Sharp*), porque foi exigência do cliente a aplicação ser para *desktop*, e das tecnologias que aprendemos no curso, C# pareceu-nos a que melhor se aplicava, por ter também parte gráfica. Algumas tecnologias não permitiam o desenvolvimento da parte gráfica.

O Microsoft Visual Studio foi usado para desenvolvimento do projeto.

O Microsoft SQL Server Management Studio 18, foi usado criação da base de dados local.

O GitHub foi usado para partilhar o projeto (<https://github.com/beatriznRibeiro96/GestaoClinicaEnfermagemProjetoInformatico>).





## 7. Arquitetura do Sistema

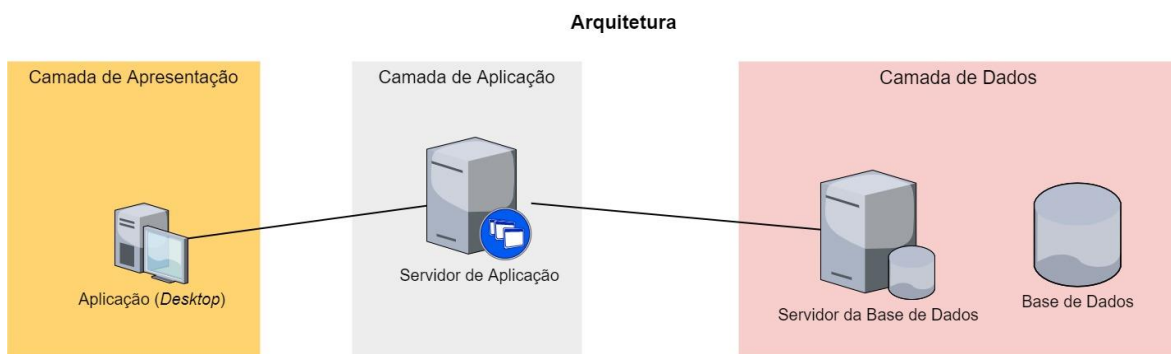
Este capítulo aborda a arquitetura do sistema. Sendo uma aplicação desktop para um computador local da clínica, não requer ligação à Internet. O *software* desenvolvido é alojado numa máquina local, o software apenas precisa de ligação à base de dados local. [13]

A camada de Apresentação serve para fornecer a interação entre o utilizador e a aplicação. Nesta camada é onde se trata os dados inseridos pelos utilizadores, visualizar os dados inseridos, tratam-se validações que não seja preciso ir à base de dados (ex: validar números de telefone, NIF, etc). [13]

A camada de Aplicação representa o núcleo da aplicação em termos de processamento, pois apenas se preocupa com a implementação das regras de negócios associadas aos problemas e as entidades de negócio. (ex: implementar as validações para saber se o email é válido). [13]

A camada de Dados fornece serviços de persistência de dados. [13]

A Figura 3 ilustra a arquitetura do sistema para a Siltes Saúde.



**Figura 3 – Arquitetura do Sistema**



## 8. Requisitos Mínimos de Sistema

Para a aplicação desenvolvida são necessários os requisitos mínimos apresentados na tabela 3.

<b>Sistema Operacional</b>	Windows 7 e superiores
<b>Processador</b>	Processador dual core x86 ou x64 bits de 1,8 gigahertz (GHz)
<b>Armazenamento</b>	Mínimo de 800MB (apenas para aplicação desktop), mais espaço para armazenamento dos dados da base de dados localmente
<b>Memória</b>	2 GB de RAM
<b>Software</b>	Acrobat Reader ou similar
<b>Base de Dados</b>	Base de dados instalada localmente

**Tabela 3 – requisitos de sistema**



## 9. Implementação do Projeto

Neste capítulo são abordados os estudos levados a cabo antes da elaboração da aplicação propriamente dita, de modo a que satisfaça os requisitos da clínica Siltes Saúde.

O desenvolvimento do projeto englobou um vasto conjunto de etapas, as quais serão apresentadas como sendo fases de desenvolvimento do mesmo.

Em cada uma das fases indicaremos resumidamente cada uma das tarefas que permitiram a realização do projeto.

Antes da fase de planeamento existiu uma reunião com o administrador da clínica, para se perceber em que consistia o projeto. Estes tópicos servem para se ter uma ideia da organização e evolução do projeto em geral.

### 9.1. Planeamento (17/02/2020 a 24/02/2020)

Em primeiro lugar, tivemos uma reunião com a administração da clínica no dia 25 de fevereiro de 2020 para fazer o levantamento de requisitos e esclarecimento de alguns pormenores sobre o projeto. A especificação de requisitos é a tarefa mais importante na fase de análise de um projeto. Os requisitos funcionais são aqueles que descrevem as ações que o sistema deve ser capaz de realizar. Os requisitos não funcionais são aqueles que descrevem como deve ser feito, habitualmente referem-se à performance, características do sistema e restrições para os requisitos funcionais.

Cronograma do Projeto Informático				
Designação	Prioridade	Estado	Início	Fim
<b>Análise e Levantamento de Requisitos</b>	Alta	Feito	2020-02-17	2020-02-24
<b>Desenho de Protótipos e Layout</b>	Alta	Feito	2020-02-24	2020-03-08
<b>Implementação</b>	Alta	Feito	2020-02-25	2020-06-30
<b>Testes de Usabilidade</b>	Alta	Feito	2020-02-25	2020-06-30
<b>Relatório</b>	Alta	Feito	2020-03-01	2020-07-10

**Tabela 4 – Cronograma do Projeto Informático**

Após a criação de um cronograma, seguimos a ordem de trabalhos do mesmo, começando com a análise e levantamento de requisitos, resultando três tabelas, uma com os requisitos do administrador, outra com os requisitos do utilizador Normal (enfermeiro) (disponíveis no [Anexo A](#)) e administrador (disponíveis no Anexo A) e, finalmente, outra com os [requisitos não funcionais](#). Seguidamente procedemos à fase de implementação que explicaremos em detalhe na secção seguinte.

## **9.2. Implementação - (25/02/2020 a 02/03/2020)**

Ao longo destes dias, efetuamos as primeiras tentativas do modelo DER (Diagrama Entidade Relacionamento) e do modelo lógico da base dados. Como as primeiras tentativas de desenho do DER e modelo lógico de dados não foram realizadas com sucesso, no dia 2 de março de 2020 tivemos a primeira reunião com o professor Rui Oliveira da unidade curricular de Base de Dados. Esta reunião teve como principal objetivo obtermos um feedback das nossas tentativas que tínhamos efetuado até à data e podermos otimizar ao máximo a base de dados. Desta reunião resultou uma nova reestruturação do modelo DER e do modelo lógico da base de dados.

## **9.3. Implementação - (03/03/2020 a 09/03/2020)**

Durante estes dias, realizamos a correção do modelo DER e no modelo lógico de dados de acordo com o feedback que obtivemos do professor Rui Oliveira. No dia 9 de março de 2020 tivemos novamente reunião com o professor Rui Oliveira para confirmar as alterações que fizemos e registarmos novas melhorias possíveis.

## **9.4. Implementação - (10/03/2020 a 16/03/2020)**

Em primeiro lugar, efetuamos pequenas alterações e correções no modelo DER e no modelo lógico de dados.

De seguida último, tentámos implementar a base de dados em várias plataformas gratuitas *online*. Este processo foi efetuado sem sucesso, por vários motivos tais como: algumas plataformas não tinham todas as opções que necessitávamos, outras tinham limite de armazenamento, o que não convinha, pois é um programa que necessita de armazenar

muitas quantidades de dados, e ainda algumas teríamos que pagar pois não eram totalmente gratuitas ou teríamos que associar um cartão de crédito. Ao fim destas tentativas sem sucesso, decidimos implementar a base de dados localmente. Para efeitos de desenvolvimento obtivemos o efeito que pretendíamos, e futuramente pretendemos por a base de dados online, num servidor, para melhor comodidade dos utilizadores desta aplicação.

### **9.5. Implementação - (17/03/2020 a 24/03/2020)**

Em primeiro lugar, demos início à implementação do modelo físico da base dados. Criamos as tabelas necessárias de acordo com os requisitos pedidos até à data e os respetivos atributos.

De seguida, iniciámos a implementação física do desenvolvimento de *software*. Nesta primeira fase realizámos várias *User Stories*. Das várias *User Stories*, desenvolvemos as seguintes; Registo de Utilizador na aplicação e Registar Enfermeiros.

### **9.6. Implementação – (25/03/2020 a 31/03/2020)**

Ao longo destes dias foram desenvolvidas várias funcionalidades da aplicação, as quais vão ser descritas em pormenor.

Em primeiro lugar, foi desenvolvida a janela inicial da aplicação (*layout*). A janela inicialmente apenas tinha as opções de Registar Utente e Registar Enfermeiro.

De seguida, foi implementada a funcionalidade de registar enfermeiro. Para esta funcionalidade ficar concluída, foi necessário desenvolver todas as validações para os dados que fossem inseridos pelo administrador ao registar um enfermeiro na aplicação. Ainda nesta fase inicial, o registo da *password* do enfermeiro na base de dados não era encriptado e não tinham sido feitas as validações necessárias para este campo.

A janela do login da aplicação foi desenvolvida ao longo destes dias. Esta janela inclui um campo *username* e um campo *password*.

Depois do *login* estar implementado, percebemos que a funcionalidade registar enfermeiro não estava sincronizada com a base de dados, isto é, era necessário registar o

enfermeiro, terminar o programa, voltar a abrir o programa e só desta forma era possível iniciar sessão com o enfermeiro que se tinha acabado de registar.

Foi implementada a encriptação da *password* do enfermeiro. A encriptação foi feita usando o algoritmo de encriptação MD5. Este algoritmo é uma função de *hash*, que produz um valor de *hash* de 128 bits.

Foi iniciada a pesquisa sobre como fazer a recuperação da *password* do enfermeiro com um email real (*Gmail*).

Por fim, foi efetuada uma correção na função de encriptação da *password*.

### **9.7. Implementação – (01/04/2020 a 07/04/2020)**

Depois de uma reunião com o coordenador de projeto, concluímos que era necessário e fazia sentido ter dois tipos de utilizadores, um do tipo administrador e outro do tipo enfermeiro. Este utilizador administrador teria acesso a várias funcionalidades que só a ele diziam respeito. Apenas o administrador deve registar enfermeiros na aplicação, e realizar outras tarefas que ainda serão implementadas. Um enfermeiro ao ser criado por um administrador, tem uma *password default* (User1234\*).

Foi criada uma função enfermeiro com o objetivo de se poder aceder aos dados que estão na base de dados sem ter que se fazer constantemente *getters* e *setters*.

Foi iniciada a janela do menu de administrador e alteração da janela do enfermeiro para ser visível apenas para o administrador e só este consegue registar enfermeiros.

Implementou-se a janela de ver todos os enfermeiros registados, disponível apenas para o administrador (mas o administrador não vê todos os dados do enfermeiro).

Concluímos a janela de registar utente, em que esta janela está disponível para o administrador e para o enfermeiro. De seguida, foram criados os botões de fechar em cada janela. Estes botões foram implementados de forma que não termine o programa abruptamente e não perca o que estava a fazer.

Implementação da janela para redefinir o Primeiro Acesso. Esta janela consiste em que o enfermeiro ao iniciar sessão pela primeira vez possa definir a sua *password* e o seu *username*.

No fim desta funcionalidade concluída, criámos a janela *Send Code Password*. Esta janela tem a função de enviar um código para o email real do utilizador (*Gmail*) para voltar a definir o *username* e a *password* se forem esquecidos.

Foram feitas correções na funcionalidade de inserção dos dados dos utentes. Em todas as janelas já desenvolvidas até à data adicionámos a hora e data atual. Para que seja visível a hora e data na aplicação.

Seguidamente, foi feita a janela das definições do enfermeiro. Esta janela permite ao enfermeiro realizar duas opções. Em primeiro lugar, permite alterar todos os campos pessoais que pretender. Por fim, permite alterar a sua *password*. Para estas funcionalidades foi necessário criar mais duas janelas, uma para cada funcionalidade. A janela para realizar a alteração da *password* e a janela para alterar os dados pessoais.

Durante a implementação destas funcionalidades reparamos na existência de um erro na classe Enfermeiro e na base de dados pois o tipo de dados do campo contacto era *double* e estava a causar conflitos. O problema foi resolvido ao alterar o contacto na base de dados e na classe Enfermeiro para um inteiro.

Foi desenvolvida uma função geral (função *validarDados()*) para validação dos dados inseridos ao registar enfermeiro e o registar utente.

Depois, foi criada a janela para listar todos os utentes registados na base de dados. Ao fazer a listagem dos utentes registados na base de dados percebemos, que qualquer enfermeiro com sessão iniciada podia ver todos os utentes, e isto não é suposto acontecer, cada enfermeiro apenas deve ter acesso aos seus utentes. Assim corrigimos o erro e já ficou a funcionar como pretendido. Cada enfermeiro assim tem acesso à lista dos seus utentes. O erro foi corrigido na base de dados, na tabela do Paciente ao inserir como chave estrangeira o *IdEnfermeiro*, com ligação à tabela Enfermeiro.

Foram feitas correções nas janelas ***ForgotPassword*** e ***ForgotUsername***, estas correções tiveram como objetivo por a funcionar com o *Gmail*, devido a uns parâmetros do *Gmail*, não era possível colocar estas funcionalidades a funcionar. Corrigindo estes problemas, as janelas ***SendCodePassword*** e ***SendCodeUsername*** também ficaram a funcionar.

De seguida, foi criada a janela para iniciar as consultas (IniciarConsultaMarcada) bem como a alteração da tabela consulta. Esta janela tem como objetivo registar todos os dados da consulta. A janela de marcação de consultas foi criada de seguida que tem como objetivo registar a hora e data da consulta de um paciente, sendo estes dados todos validados.

Foi dado início à janela com imagem do corpo humano, esta janela tem como principal objetivo o enfermeiro registar o local da dor do paciente durante a consulta, com *textBox* a aparecer onde o enfermeiro clica. Foi criada uma janela extra para auxiliar na resolução deste problema para fazer aparecer as *textBox* em qualquer sítio.

Implementamos a funcionalidade para editar os dados de utentes já registados na aplicação e as respetivas validações.

### **9.8. Implementação – (08/04/2020 a 14/04/2020)**

Ao testar algumas funcionalidades da aplicação, detetamos um erro ao guardar a data e hora. Aparecia um erro ao tentar guardá-las, mas não sabíamos qual o motivo daquele erro. Ao fim de várias inserções de dados e análise dos mesmos inseridos na base de dados, percebeu-se que na base de dados a data é guardada em formato americano (mm/dd/YYYY). Por este motivo dava conflitos e nem sempre guardava a data, pois o pretendido era que a data fosse guardada no formato europeu (dd/mm/YYYY), como o mês e o dia eram trocados, causava problemas no registo e leitura das datas. De modo a corrigir este problema foi necessário registar e ler as datas em formato americano e só depois é feita a conversão para o formato europeu através da função (*ToString*).

Seguidamente foi desenvolvida a funcionalidade de visualizar apenas as consultas do próprio dia na janela principal depois de se fazer login. Para realizar esta tarefa, criamos uma *dataGridView* onde mostra o nome dos utentes, o contribuinte e a hora da consulta. A janela de IniciarConsultaMarcada foi associada a esta janela criada.

Foram implementadas as funcionalidades de editar uma marcação, isto é, alterar uma hora e/ou data de uma consulta já marcada, e eliminar uma marcação, caso seja necessário cancelar uma consulta previamente marcada. Estas funcionalidades foram implementadas na janela de marcações.



Implementamos uma janela para se visualizar as consultas por cliente. Foi também implementada a janela para registar o histórico clínico do utente, outra para efetuar a listagem do histórico clínico para cada utente e outra para registar o histórico das doenças de cada utente. Nesta fase estas janelas ainda não tinham validações dos dados inseridos. Chegou-se à conclusão de que a janela para visualizar as consultas por paciente era redundante e por esse motivo foi eliminada.

Ao fim de alguns testes à aplicação percebeu-se que existiam erros a serem corrigidos, e tal foi detetado porque a única tabela que permite eliminação de dados é a janela de marcações. Em todas as *dataGridView* existentes na aplicação, no final aparecia uma linha, a mais vazia, e na janela de Marcações ao selecionar essa linha a mais e clicar no botão de ‘Desmarcar Consulta’, na base de dados era eliminada a última linha com campos preenchidos. Este erro foi resolvido alterando a propriedade *allowUsersToAddRows* para *false* da *dataGridView*. Outro erro relacionado com as *dataGridView*, era o seguinte: quando a *dataGridView* estava vazia e eram adicionados dados posteriormente, ao ir à janela onde se pretendia visualizar os dados inseridos, dava erro de execução na *dataGridView*. Estes erros foram resolvidos com o *binding* à *dataGridView*.

Foi necessário criar as classes Alergia, Cirurgia, Doença e Exame, pois são objetos usados para aceder aos dados. De seguida foi necessário criar as janelas para inserir os dados das alergias, cirurgias, doenças e exames. Depois foram adicionadas as janelas para efetuar a listagem e edição dos dados inseridos nas janelas anteriormente mencionadas.

Foi preciso criar a funcionalidade para ver as cirurgias, alergias, exames e doenças já registadas na aplicação. E, a funcionalidade para registar todas estas opções.

Por último, foram desenvolvidas as janelas relacionadas com os utentes, tais como: registar e visualizar as cirurgias, doenças, alergias e exames no utente selecionado.

## 9.9. Implementação – (15/04/2020 a 21/04/2020)

Foram corrigidos erros na inserção de exames e cirurgias no lado do utente.

De modo a evitar *SQL Injection*, foram alterados todos os métodos de inserção e atualização dos dados de todas as funcionalidades até aqui implementadas.

Nas janelas para visualizar a avaliação objetiva do utente, visualizar as cirurgias do utente, visualizar as doenças do utente, visualizar os exames do utente e na área das marcações, foram acrescentadas as validações necessárias.

Foi criada a classe para o tratamento de encomendas. De seguida, foram criadas as janelas para registar encomendas, linha de encomenda, listar produtos, finalizar encomenda, produtos em *stock* (aqui os produtos ainda não tinham um fornecedor associado, foi associado mais a frente). Para cada janela foram implementadas diversas validações.

No dia 18 de abril de 2020, foi feita uma reunião com o cliente para ver o projeto, da reunião resultaram novas funcionalidades a acrescentar no projeto e alterações algumas necessárias ao que já estava implementado até ao momento.

#### **9.10. Implementação – (22/04/2020 a 28/04/2020)**

Ao listar os dados dos utentes já registados na aplicação, percebemos que visualmente ficava melhor efetuar a concatenação do código postal em vez de estar separado, pois na base de dados, estamos a armazenar o código postal separado em dois campos, um designado por *codigoPostalPrefixo* e outro designado por *codigoPostalSuffixo*.

Foi feita uma análise profunda aos novos requisitos do cliente. Assim sendo foi necessário efetuar uma análise do DER, do modelo lógico e do físico da base de dados de forma a poder acoplar os novos dados.

#### **9.11. Implementação – (29/04/2020 a 05/05/2020)**

Nas janelas para adicionar utente e avaliação objetiva foram colocados os novos dados pedidos pelo cliente, sem estarem ligados a base de dados (foram apenas colocados no layout). Foi criada uma janela para a avaliação objetivo do bebe sem os campos estarem a ser registados na base de dados.

Foi feita a reformulação do DER, do modelo lógico e físico da base de dados para acoplar todos os novos dados pedidos pelo cliente.

Foi feita a implementação física da tabela *AvaliacaoObjetivaBebe*, e alteração da tabela dos utentes para conter os novos dados pedidos.

### **9.12. Implementação – (06/05/2020 a 12/05/2020)**

Implementação física das tabelas para guardar os métodos contraceptivos, os tipos de parto e aleitamento existentes. Estas tabelas são, Aleitamento, MetodoContraceptivo e Parto. Alteração da tabela *AvaliaçãoObjetiva* para armazenar os novos dados pedidos pelo cliente. Implementadas as funcionalidades para adicionar e alterar métodos contraceptivos, tipos de parto e aleitamento.

Na janela para registar novos utentes foi adicionado o campo sexo, plano nacional de vacinação. Foi adicionado também um campo de acordos, isto é, se o utente tiver seguro de saúde, se pertencer a um subsistema de saúde ou número do Sistema Nacional de Saúde. Estas modificações tiveram reflexo também na base de dados.

Foi concluída a janela da avaliação objetiva do bebé e as suas respetivas validações.

De acordo com os requisitos do cliente, a janela para registar utentes e a que mostra os detalhes dos utentes sofreram alterações, sendo assim foi necessário acrescentar mais validações a estas novas funcionalidades.

Devido à grande quantidade de novos dados para inserir na base de dados, tivemos necessidade de nos reunirmos com o professor Rui Oliveira no dia 11 de maio de 2020, para ver o nosso DER e dar uma explicação de qual seria a melhor forma de alterar a base de dados com os novos dados.

### **9.13. Implementação – (13/05/2020 a 19/05/2020)**

Implementação do modelo físico da tabela profissões. De seguida foram implementadas as janelas AdicionarProfissao e VerEditarProfissao. Estas janelas foram adicionadas na janela de InserirUtente e EditarUtente. E foram efetuadas as devidas validações.

Ao efetuar os testes de usabilidade do sistema, e ao inserir os dados, foi detetado um erro com os novos dados, como nem todos os campos são obrigatórios, quando se inseria nas tabelas um campo vazio às vezes ficava *NULL* e outras vezes ficava vazio, e quando havia uma tentativa de leitura de dados aparecia exceções. Devido a este erro, foi feita a devida correção em todo o projeto. Os dados quando não são obrigatórios, as vezes registava a *NULL* e outras vezes a vazio, e para garantir a conformidade, cada vez que os

dados não são obrigatórios regista a *NULL* na base de dados, e ao fazer a leitura dos dados, se os mesmos forem *NULL*, o registo aparece vazio.

Foi adicionado em todo o projeto a opção do *errorProvider*, para quem vai usar a aplicação, saiba exatamente quais os campos obrigatórios que se esqueceu de preencher, os campos que estão mal preenchidos. Com a inserção do *errorProvider* no projeto, foram detetadas validações que faltavam e então foram adicionadas essas validações.

Ao inserir dados no sistema, surgia uma exceção relacionada com os campos contacto e NIF. Estes dados na base de dados estavam a ser guardados como dados do tipo *double*, o que provocava por vezes exceções na inserção e leitura dos dados. Para resolver este problema alteramos o tipo de dados destes campos para o tipo inteiro, e efetuamos as alterações necessárias nas janelas que contém estes campos.

Aquando estas alterações e validações feitas, foram feitos novos testes com a inserção de mais dados, e foi detetado um erro no campo email da tabela Paciente. O campo email continha a *Constraint Unique*. Esta restrição, faz com que os dados inseridos sejam únicos e não seja possível que existam 2 emails iguais registados. Se o campo não fosse preenchido, ficava registado na base de dados o valor *NULL*, e ao inserir dois utentes com o campo email vazio, aparecia uma mensagem de erro, pois supostamente aquele email já estava registado, com o valor *NULL*. Esta restrição teve que ser retirada, pois há uma grande maioria dos utentes (idosos) que não tem email, logo este campo não podia conter a *Constraint Unique*. Retirando esta restrição, o campo quando não é preenchido fica a *NULL*. A validação do campo email a *UNIQUE* passou a ser feita unicamente no código da janela, de modo a garantir que é um email válido e único.

De modo a limpar todos os campos de uma janela, se o utilizador assim entender, foi implementada uma função *limparCampos()*, de modo a limpar todos os campos sem ter que sair da janela. A funcionalidade foi implementada em todo o projeto.

Até à presente data, a janela das despesas não estava funcional, tinha apenas o layout definido, então foi desenvolvido o código para o colocar funcional.

Devido ao aparecimento de exceções causadas por chaves estrangeiras na base de dados, foi corrigido o código relativo às encomendas e linha de encomenda.

Ao testar a edição do nome do aleitamento, tipo de parto e método contraceptivo passou a existir partes da avaliação objetiva e avaliação subjetiva do bebé que deixou de funcionar. Por serem nomes muito específicos, os nomes destes campos têm que ser exatamente

iguais aos que estão no código. De modo a não permanecer este erro, pois é muito normal que o utilizador ao inserir dados se engane, foi decidido não permitir alterar o nome destes campos. Se ao querer realizar a avaliação objetiva, os tipos de aleitamento, parto e os métodos contraceptivos não estiverem registados na base de dados como está no código, são registados internamente pela aplicação para que funcione corretamente. Esta decisão foi tomada apenas para esta primeira fase do projeto, pois quando o sistema estiver em produção será corrigido este problema, e ficará a funcionar corretamente.

#### **9.14. Implementação – (20/05/2020 a 26/05/2020)**

Foi detetado que o ID das encomendas estava a ser registado a 0, então foi necessário corrigir este problema. Foram inseridas novas validações na encomenda.

Implementou-se a funcionalidade de visualizar as encomendas feitas e criação da janela das despesas e os respetivos filtros para ser possível efetuar as pesquisas.

#### **9.15. Implementação – (02/05/2020 a 02/06/2020)**

Para ser possível adicionar vários tipos de análises laboratoriais, foi criada a respetiva janela. Foi implementada a funcionalidade para editar os tipos de análises laboratoriais e foi feita a conclusão das validações na opção para finalizar a encomenda, de modo a atualizar o número dos produtos em *stock* quando uma encomenda é finalizada.

#### **9.16. Implementação – (03/06/2020 a 09/06/2020)**

Implementação da tabela Tratamento e desenvolvimento do respetivo *layout* e código para a funcionalidade adicionar tratamento e adicionar a lista para visualizar os tipos de tratamento. Criação da janela para se registar os diferentes tratamentos que o utente fez. Por facilitar o código, teve-se necessidade de criar a classe TratamentoPaciente para fazer a ligação à base de dados. À base de dados adicionou-se as tabelas TipoQueimadura e tipoUlcera. De seguida, implementou-se as janelas para adicionar o tipo de queimadura e o tipo de úlcera. Cada uma, regista os tipos de queimadura existentes e os tipos de úlceras existentes, respetivamente. Criou-se a classe Queimadura e Úlcera para estabelecer ligação à base de dados.

Foi criada a janela que permite registar os produtos utilizados numa consulta e a respetiva quantidade. Numa inserção de dados, detetou-se um erro na janela

LinhaEncomenda, pois estava a inserir mal os dados. Havia um erro no código de inserir, tinha os campos da base de dados mal.

### **9.17. Implementação – (10/06/2020 a 16/06/2020)**

Desenvolvimento da janela e respetivo código para visualizar análises laboratoriais do Paciente, em que é possível também registar os dados das análises que o paciente realizou. De seguida, desenvolveu-se a funcionalidade que permite registar a medicação que o paciente toma, que inclui o nome do medicamento, quantidades e período do dia em que toma.

Foi implementada a janela para registar os dados do tratamento de excisões e a classe TratamentosAnteriores, para ser possível estabelecer ligação à base de dados. Com esta classe auxiliar, criou-se a janela para listar os tratamentos anteriores efetuados pelo utente.

Ajustaram-se os filtros da funcionalidade que permitem listar todas as despesas já registadas.

No seguimento dos últimos pedidos do cliente, implementaram-se as funcionalidades que permitem registar várias atitudes terapêuticas. Como cada atitude terapêutica tem campos muito específicos, criou-se janelas diferentes para cada uma. Essas funcionalidades são adicionar algaliação, aspiração secreção, cateterismo, colheita urina, medicação, ver os tipos de atitudes terapêuticas (que consiste em listar as atitudes terapêuticas que o paciente já tenha realizado), adicionar atitudes terapêuticas (esta janela consiste em atitudes terapêuticas que não têm campos para preencher, e pode-se selecionar várias atitudes simultaneamente), teste avaliação acuidade visual, desbridamento, drenagem de locas, ENG, flebografia, inalação, lavagem auricular, lavagem ocular, lavagem vesical, monitorização ECG, pressoterapia, suturas, vacinação, tricotomia, zaragatoa orofaringe, espirometria, teste de Combur. Quando o utente é do sexo feminino existem todas as atitudes terapêuticas referidas anteriormente e mais estas que só são efetuadas apenas nas mulheres, como a colocação DIU, colpocitologia, implante contraceptivo. Para cada uma destas atitudes terapêuticas foi desenvolvendo a funcionalidade para ver os dados anteriores, caso o utente já tenha realizado aquela atitude terapêutica. Para cada atitude terapêutica foram desenvolvidas as seguintes janelas, ver algaliação, ver aspiração secreções, ver cateterismo, ver colheita de urina, ver desbridamento, ver drenagem de locas, ver ENG, ver espirometria, ver flebografia, ver imagem (é apresentada uma imagem da anatomia do corpo humano onde é possível registar a localização da dor, ou da atitude terapêutica

efetuada ou do tratamento efetuado), ver inalações, ver medicação, ver vacinação, ver teste avaliação acuidade visual, ver colocação DIU, ver implante contraceptivo, ver lavagem ocular, ver lavagem auricular, ver lavagem vesical, ver monitorização ECG, ver várias atitudes efetuadas pelo utente, ver pressoterapia, ver suturas, ver tricotomia, ver zaragatoa orofaringe, ver colpocitologia, ver teste Combur.

Para cada atitude terapêutica foi necessário, implementar as respetivas tabelas na base de dados, essas tabelas são as seguintes, Algaliacao, AspiracaoSecrecoes, Cateterismo, ColheitaUrina, Desbridamento, DrenagemLocas, ENG, Espirometria, Flebografia, Inalacoes, Medicacao, Vacinacao, Atitude, AcuidadeVisual, ColocacaoDIU, ImplanteContraceptivo, LavagemAuricular, LavagemOcular, LavagemVesical, MonitorizacaoECG, Pressoterapia, Suturas, Tricotomia, ZaragatoaOrofaringe, Colpocitologia, TesteCombur.

Foram desenvolvidas as classes para cada uma das atitudes para se estabelecer ligação à base de dados.

No que diz respeito ao registo dos vários tratamentos, foram desenvolvidas as funcionalidades para adicionar outros tratamentos (que permite que o enfermeiro tenha acesso a mais tipos de tratamentos possíveis de realizar), registar os dados do tratamento doppler fetal e ver os registos do doppler fetal já realizados anteriormente.

### **9.18. Implementação – (17/06/2020 a 23/06/2020)**

Desenvolvimento da janela que permite registar a localização da dor, através duma imagem do corpo humano. Esta janela permite que ao clicar em qualquer lugar da imagem apareça um campo de escrita para que o enfermeiro possa registar a informação que pretende sobre a dor, pode também registar observações e guardar esta informação. Para tornar este pedido do cliente funcional, foi necessário efetuar muita pesquisa e muitos testes até obter o resultado final.

De seguida, implementou-se a funcionalidade que permite registar os dados do audiograma. Para registar esta atitude terapêutica, foi-nos solicitado pelo cliente para ser possível desenhar círculos em qualquer sítio da imagem e imprimir essa mesma imagem. Estas funcionalidades relacionadas com as imagens e manipulação das mesmas, foi uma das maiores inovações que este sistema tem, e requereu muita pesquisa e tempo perdido em testes para que funcionasse da forma correta. Como para as atitudes terapêuticas

anteriores, foi necessário implementar as respetivas tabelas na base de dados (tabela audiograma e localizacaoDor).

### **9.19. Implementação – (24/06/2020 a 30/06/2020)**

Ao longo desta semana foram desenvolvidas as funcionalidades que dizem respeito aos tratamentos. Como existem diferentes tipos de tratamentos com campos muito específicos, tal como nas atitudes terapêuticas, foi necessário criar janelas diferentes para cada um, tabelas e classes para estabelecer a ligação à base de dados. Para cada tratamento foi necessária a criação de tabelas distintas, essas tabelas são, DopplerFetal, LocalizacaoDorDopplerFetal, LocalizacaoDorDopplerArterialVenoso, TipoQueimadura, TipoUlcera, Tratamento, TratamentoExcisoes, TratamentoMaosPes, TratamentoPaciente. Foram implementadas as seguintes janelas que permitem registar os dados dos seguintes tratamentos, Doppler Arterial Venoso, Dor no corpo, Onicocriptoses, tratamento mãos e pés. Para cada uma destas opções de tratamentos, foi desenvolvida a opção para listar os dados dos tratamentos anteriores realizados.

Ao realizar algumas inserções de dados, foi detetado um erro na funcionalidade que permite editar os dados de um utente e de um enfermeiro, em que os campos, NIF, email e número do SNS são únicos, e não é permitido alterar esses campos, mas ao alterar outros campos e tentar guardar as alterações feitas, aparecia uma mensagem de erro em como aqueles dados já estavam registados. Para resolver este problema, teve que se verificar se os dados tinham sido alterados ou não, e desta forma já não aparece a mensagem de erro.

### **9.20. Implementação – (01/07/2020 a 10/07/2020)**

Testes as funcionalidades integrais do programa, para deteção de eventuais erros não detetados.

Conclusão do relatório. Descrição detalhada da base de dados e elaboração do manual de utilizador.



## 10. Erros de desenvolvimento ao longo do projeto

A tabela seguinte apresenta alguns dos erros que foram encontrados ao longo do desenvolvimento do projeto e como foram resolvidos. Decidimos apresentá-los sob forma de tabela para uma melhor perceção.

Designação	Descrição	Resolução
<b>Erro1</b>	Desenho inicial do DER.	Reunião com o professor Rui Oliveira para corrigir o que estava errado e otimizar ao máximo a base de dados.
<b>Erro2</b>	Escolher a melhor plataforma gratuita para implementar a base de dados.	Implementação da base de dados localmente, que para efeitos de desenvolvimento temos o efeito pretendido.
<b>Erro3</b>	Depois da funcionalidade para iniciar sessão estar implementada, a funcionalidade para registar enfermeiro deixou de funcionar.	Alocação à base de dados estava mal feita.
<b>Erro4</b>	Encriptação da <i>password</i> .	Correção na função que faz a encriptação da <i>password</i> .
<b>Erro5</b>	Tipo de campo contacto era <i>double</i> e causava problemas na inserção de dados no Enfermeiro.	Alteração na base de dados e na classe Enfermeiro, do tipo de dados do contacto para inteiro.
<b>Erro6</b>	Qualquer enfermeiro com	Associámos o paciente ao

	sessão iniciada tinha acesso a todos os utentes registados, mesmo que não tivesse sido ele a registar.	enfermeiro através da chave estrangeira ( <i>IdEnfermeiro</i> ), e desta forma cada enfermeiro só tem acesso aos dados dos seus utentes.
<b>Erro7</b>	Funcionalidade para recuperar <i>password</i> e <i>username</i> não funcionava com o <i>Gmail</i> .	Alterou-se uns parâmetros relacionados com o <i>Gmail</i> .
<b>Erro8</b>	Gravar datas na base de dados.	Registar as datas e fazer a leitura das mesmas em formato americano (mm/dd/YYYY) e só depois fazer a conversão para o formato de data para o europeu (dd/mm/YYYY) é que ficou correto.
<b>Erro9</b>	Janela para visualizar consultas por paciente é redundante.	Eliminou-se a janela, visto que estávamos a repetir aquela funcionalidade.
<b>Erro10</b>	Eliminar dados de tabela, seleccionando uma linha da <i>DataGridView</i> .	Alterou-se a propriedade <i>allowUsersToAddRows</i> para false.
<b>Erro11</b>	Quando a <i>dataGridView</i> estava vazia e eram adicionados dados posteriormente, ao ir à janela onde se pretendia visualizar os dados inseridos, dava erro de execução na	Efetuiu-se o <i>binding</i> à <i>dataGridView</i> .

	<i>dataGridView.</i>	
<b>Erro12</b>	Campo código-postal aparecia separado na <i>dataGridView</i> .	Concatenar o campo código-postal e registar este campo, em 2 campos separados na base de dados, <i>codigoPostalPrefixo</i> e <i>codigoPostalSufixo</i> .
<b>Erro13</b>	Campos não obrigatórios ficam registados a <i>NULL</i> ou vazios na base de dados. Ao tentar ler estes dados, dava exceção.	Inserção de dados não obrigatórios ficam registados a <i>NULL</i> na base de dados. Assim quando são lidos já não aparece exceção.
<b>Erro14</b>	Ao inserir dados que tenham o campo NIF e contacto no sistema aparece exceção.	Alteração do tipo destes campos para int.
<b>Erro15</b>	Campo email com restrição de <i>Constraint Unique</i> . Esta restrição faz com o que os dados sejam únicos, mas o email não é um campo obrigatório, logo quando são inseridos 2 utilizadores sem email, ficam registados com o valor <i>NULL</i> , e por este motivo dava exceção por já existir um valor igual.	Retirou-se a restrição e desta forma, o email quando não é preenchido fica com o valor <i>NULL</i> e deixa de ser obrigatório. A validação deste campo passou a ser feita apenas para validar se é um email válido.
<b>Erro16</b>	Exceções devido a chaves estrangeiras na base de dados.	Correção de código relativo às encomendas e à linha de encomenda.

<b>Erro17</b>	Editar dados do tipo de parto, tipo de aleitamento e métodos contraceptivos se não forem escritos de maneira exatamente igual dá erro.	Não permitir que o utilizador altere o nome destes campos.
<b>Erro18</b>	ID das encomendas estava a ser registado a 0.	Feitas novas validações no código.
<b>Erro19</b>	Círculos que se desenhavam no audiograma não ficavam no lugar onde o rato clicava.	Ajustar imagem à resolução do ecrã.
<b>Erro20</b>	Impressão da imagem do audiograma ficava cortada.	Ajustar imagem à resolução do ecrã e definir tamanho fixo para a imagem.
<b>Erro21</b>	Inserção errada de dados na Linha de Encomenda.	O inserte estava mal feito. Foi corrigido.
<b>Erro22</b>	Editar dados de Enfermeiro, em que os campos NIF, email e número de SNS não são permitidos alterar. Mas ao guardar os outros campos alterados, aparecia uma exceção pois aqueles dados já estavam registados.	Foi necessário verificar se os dados tinham sido alterados ou não, e desta forma já não aparece a mensagem de erro.

**Tabela 5 - Erros detetados ao longo do desenvolvimento**

## 11. Testes

Do ponto de vista de engenharia de *software* a realização de testes implica um plano de testes e se os testes foram bem-sucedidos ou não. Nesse sentido, aplicamos um tipo de testes que achamos mais importantes para verificar a consistência do *software* e se o que está implementado está de acordo com o que foi solicitado por parte do cliente. Esses conjuntos de testes vão ser descritos ao longo deste capítulo. <sup>[11] [12]</sup>

Em primeiro lugar, realizamos testes de Usabilidade.

Em segundo lugar, realizamos testes de Aceitação.

E em último lugar testes de Sistema.

### 11.1. Testes de Usabilidade

Estes tipos de teste têm por objetivo verificar a estética, acessibilidade e consistência da aplicação. Recebemos *feedback* por parte do cliente acerca do nosso sistema, que testou a aplicação à nossa frente e a reação foi bastante positiva, pois desta forma conseguimos retirar todas as emoções por parte de quem está a testar o sistema. <sup>[7]</sup>

De uma forma geral o cliente (administrador da clínica Siltes Saúde) concordou que os recursos eram de fácil acesso e intuitivos e quando lhe pedíamos para efetuar uma operação sem ainda conhecer bem o software sempre se obteve sucesso sem ser necessária a indicação das operações a realizar. <sup>[7]</sup>

### 11.2. Testes de Aceitação

Os testes de aceitação foram feitos pelo cliente (administrador da clínica Siltes Saúde) na última reunião que tivemos. Percorremos todas as funcionalidades do sistema e todos os requisitos foram cumpridos, o que deixou o cliente satisfeito. <sup>[8][9]</sup>

Apenas foram detetados erros ortográficos, a nível de termos específicos de enfermagem, e que foram corrigidos de imediato. <sup>[8][9]</sup>

### **11.3. Testes de Sistema**

O principal objetivo dos testes de sistema é verificar que ao executar as várias funcionalidades com diferentes utilizadores, sob o ponto de vista de utilizador final o software funciona sem erros e consistentemente. <sup>[10]</sup>

## 12. Conclusão

Antes de tomar a decisão de abraçar este projeto tínhamos consciência da complexidade que o projeto tinha.

Apesar da complexidade do software, pareceu-nos que esta ideia se poderia integrar bem no projeto informático e decidimos embarcar na aventura.

A maior motivação por submeter esta proposta para projeto informático e não outro, deveu-se ao facto deste projeto “não ficar na gaveta”, como podem ficar outros projetos. Trabalhar com um cliente real que nos exige coisas, e poder ver a sua satisfação ao ser-lhe apresentado o trabalho, é sem dúvida a melhor motivação que podemos ter.

Depois de todos os sacrifícios para conseguir concluir este projeto com uma versão estável, todos os esforços foram recompensados ao ver o cliente bastante satisfeito com o trabalho desenvolvido. O medo do desconhecido, pois não temos qualquer conhecimento na área da saúde, hoje tornam-nos mais seguras em situações semelhantes.

A unidade curricular de Projeto Informático é totalmente diferente de todas as unidades curriculares que tivemos contacto durante a licenciatura. Pois nesta unidade curricular existe o orientador, para nos orientar e não para nos explicar todos os problemas e dúvidas que temos, aqui é valorizada a autonomia. Todas as horas investidas em aprender novas técnicas e a encontrar erros ajudaram-nos a perceber em que consiste desenvolver um projeto com uma dimensão destas.

A maior recompensa que podemos ter é ver o nosso software implementado na clínica Siltes Saúde a funcionar em ambiente de produção, e quem sabe noutras clínicas futuramente, pois como referido pelo cliente, na área de enfermagem, o único sistema conhecido é o do Sistema Nacional de Saúde.

Em último lugar, fazemos questão de agradecer mais uma vez ao nosso orientador professor Paulo Cordeiro, por todo o ânimo e sugestões que nos transmitiu em todas as reuniões que tivemos ao longo do semestre.

## 13. Trabalho Futuro

Quando este projeto chegar à fase de produção, existem várias áreas onde já sabemos que podemos e devemos melhorar e acrescentar.

Neste momento podemos identificar as seguintes funcionalidades que falta implementar ou melhorar.

Em primeiro lugar, melhorar a funcionalidade para prescrever medicação. Para os utentes idosos, imprimir a guia de medicação com imagem da quantidade de comprimidos que deve tomar, isto é, quando a quantidade é um comprimido, aparecer a imagem de um comprimido, se for metade de um comprimido, aparecer a imagem de metade de um comprimido.

Em segundo lugar, quando se regista os dados de um bebé, ter a opção de associar esse bebé à mãe (se a mãe for utente na clínica).

Em terceiro lugar, alterar a base de dados, para se poder organizar os dados dos utentes por processos. Isto significa que temos que acrescentar a opção de “Alta” e “Número do Processo”. Quando um paciente tem alta, aquele processo fica encerrado. Caso o utente volte mais tarde, inicia-se um novo processo.

Em quarto lugar, implementar estatísticas, que façam a relação das despesas com as receitas da clínica. Estatísticas que permitam ao administrador fazer um estudo de quais doenças são mais comuns, quais os tratamentos mais realizados e atitudes terapêuticas, qual a maior faixa etária que frequenta a clínica, a duração média de uma consulta, etc.

Em quinto lugar, cifrar todos os dados registados no software.

Em sexto lugar, pretendemos alocar a base de dados num servidor online. Para esta funcionalidade, teremos de nos reunir com o cliente para decidir onde alocar os dados, consoante o que está disposto a pagar, e dependendo da quantidade de dados que necessita de alocar.

Em último, realizar a manutenção do sistema sempre que for solicitado.



Seguramente que podemos melhorar em muito mais áreas e acrescentar ainda mais funcionalidades, e para tal esperemos continuar a contar com o apoio e disponibilidade da administração para nos receber sempre que possível.

## 14. Bibliografia

[1] MJV Technology & Innovation. 2020, *Modelo Cascata X Modelo Ágil: Qual Usar Em Seu Negócio?*, Disponível na Internet:

<https://www.mjvinnovation.com/pt-br/blog/modelo-cascata-modelo-agil/>

[2] *O modelo em cascata*, Medium, Disponível na Internet:  
<https://medium.com/contexto-delimitado/o-modelo-em-cascata-f2418addaf36>

[3] Nathália Tameirão, Metodologias ágeis: o que são e quais os principais tipos, Disponível na Internet:

<https://sambatech.com/blog/insights/metodos-ageis/>

[4] Desenvolvimento ágil de software, Disponível na Internet:

[https://pt.wikipedia.org/wiki/Desenvolvimento\\_%C3%A1gil\\_de\\_software](https://pt.wikipedia.org/wiki/Desenvolvimento_%C3%A1gil_de_software)

[5] Metodologia de Desenvolvimento de Software, Disponível na Internet:

<https://www.devmedia.com.br/metodologia-de-desenvolvimento-de-software/1903>

[6] Pinheiro, Metodologias de Desenvolvimento de Software, Disponível na Internet:

<https://pt.slideshare.net/alvarofpinheiro/metodologias-de-desenvolvimento-de-software>

[7] Teste de Usabilidade – Quando meu software é aparentemente bom para o cliente? | Matera, Disponível na Internet:

<http://www.matera.com/blog/post/teste-de-usabilidade-quando-meu-software-e-aparentemente-bom-para-o-cliente>

[8] Teste de aceitação, Disponível na Internet:

[https://pt.wikipedia.org/wiki/Teste\\_de\\_aceita%C3%A7%C3%A3o](https://pt.wikipedia.org/wiki/Teste_de_aceita%C3%A7%C3%A3o)

[9] Sistemas, Teste de Aceitação: problemas, desafios e abordagens, *Disponível na Internet:*

<https://pt.slideshare.net/synergiaufmg/teste-de-aceitao-problemas-desafios-e-abordagens>

[10] Teste de sistema, *Disponível na Internet:*

[https://pt.wikipedia.org/wiki/Teste\\_de\\_sistema](https://pt.wikipedia.org/wiki/Teste_de_sistema)

[11] Testar.me, *Disponível na Internet:*

<https://www.testar.me/teste-de-software>

[12] Tipos de teste de software: saiba o que é o teste e quais são os principais, *Disponível na Internet:*

<https://blog.cronapp.io/tipos-de-teste-de-software/>

[13] Definindo a arquitetura de um projeto de software, *Disponível na Internet:*

[http://www.macoratti.net/12/11/net\\_arq1.htm](http://www.macoratti.net/12/11/net_arq1.htm)

Error Blink:

<https://docs.microsoft.com/en-us/dotnet/api/system.windows.forms.errorblinkstyle?view=netcore-3.1>

Para converter os dias para meses, para a avaliação objetivo do bebé apenas estar disponíveis até aos 36 meses:

<https://www.c-sharpcorner.com/forums/convert-days-to-months>

Consulta do manual de endereçamento dos CTT porque reparamos a morada não estava a ser remetida de forma correta:

[https://www.ctt.pt/application/themes/pdfs/empresas/manual\\_enderecamento\\_abr\\_2018.pdf](https://www.ctt.pt/application/themes/pdfs/empresas/manual_enderecamento_abr_2018.pdf)

Link consultado para tentar resolver exceção de não guardar valores alterados:

<https://stackoverflow.com/questions/963601/datagridview-value-does-not-gets-saved-if-selection-is-not-lost-from-a-cell>

Ajustar imagens:

<https://stackoverflow.com/questions/3752451/enter-key-pressed-event-handler>

[https://social.msdn.microsoft.com/Forums/pt-BR/c040ed85-15e4-4a6f-a8d9-](https://social.msdn.microsoft.com/Forums/pt-BR/c040ed85-15e4-4a6f-a8d9-1c6209326ca8/c-apagar-ultimo-caracter-de-uma-string?forum=504)

[1c6209326ca8/c-apagar-ultimo-caracter-de-uma-string?forum=504](https://social.msdn.microsoft.com/Forums/pt-BR/c040ed85-15e4-4a6f-a8d9-1c6209326ca8/c-apagar-ultimo-caracter-de-uma-string?forum=504)

Resize de impressão:

[https://stackoverflow.com/questions/10442269/scaling-a-system-drawing-bitmap-to-a-](https://stackoverflow.com/questions/10442269/scaling-a-system-drawing-bitmap-to-a-given-size-while-maintaining-aspect-ratio)

[given-size-while-maintaining-aspect-ratio](https://stackoverflow.com/questions/10442269/scaling-a-system-drawing-bitmap-to-a-given-size-while-maintaining-aspect-ratio)

Abrir ficheiro:

[https://pt.stackoverflow.com/questions/60836/como-faço-para-abrir-um-arquivo-de-um-](https://pt.stackoverflow.com/questions/60836/como-faço-para-abrir-um-arquivo-de-um-diretório-específico-em-pdf-no-visual-bas)  
[diretório-específico-em-pdf-no-visual-bas](https://pt.stackoverflow.com/questions/60836/como-faço-para-abrir-um-arquivo-de-um-diretório-específico-em-pdf-no-visual-bas)

[https://www.codeguru.com/csharp/.net/net\\_general/generating-a-pdf-document-using-c-](https://www.codeguru.com/csharp/.net/net_general/generating-a-pdf-document-using-c-.net-and-itext-7.html)  
[.net-and-itext-7.html](https://www.codeguru.com/csharp/.net/net_general/generating-a-pdf-document-using-c-.net-and-itext-7.html)

## 15. Anexos

### 15.1. Anexo A – Requisitos Funcionais

Neste anexo é possível ver os requisitos funcionais resultantes das reuniões com o administrador da clínica e que foram implementados.

Designação	Caso de Uso	Descrição	Prioridade
<b>US1</b>	Registar Enfermeiro	A aplicação deverá permitir registar enfermeiros.	Alta
<b>US2</b>	Criar Listagem de Enfermeiros Registados	A aplicação deverá permitir listar todos os enfermeiros registados.	Alta
<b>US3</b>	Registar Produtos associados a um fornecedor	A aplicação deverá permitir registar produtos do fornecedor correspondente.	Alta
<b>US4</b>	Criar Listagem dos produtos.	A aplicação deverá permitir listar os produtos registados.	Alta
<b>US5</b>	Registar Fornecedores	A aplicação deverá permitir registar fornecedores.	Alta
<b>US6</b>	Criar Listagem de Fornecedores Registados	A aplicação deverá permitir listar todos os fornecedores registados.	Alta
<b>US7</b>	Pesquisar fornecedor	A aplicação deverá permitir pesquisar um fornecedor pelo nome.	Alta
<b>US8</b>	Registar Encomendas	A aplicação deverá permitir registar as encomendas efetuadas.	Alta
<b>US9</b>	Criar Listagem de	A aplicação deverá permitir	Alta

	Encomendas	listar todas as encomendas.	
<b>US10</b>	Finalizar uma encomenda.	A aplicação deverá permitir finalizar uma encomenda, registando a data de entrega real.	Alta
<b>US11</b>	Visualizar encomendas.	A aplicação deverá permitir visualizar as encomendas realizadas, mostrando os dados do fornecedor, data de registo da encomenda, data de entrega prevista, o total com iva e sem iva.	Alta
<b>US12</b>	Registar Despesas da Clínica	A aplicação deverá permitir registar as despesas da clínica.	Alta
<b>US13</b>	Criar Listagem de despesas	A aplicação deverá permitir listar as despesas registadas.	Alta
<b>US14</b>	Registar Tipo de Encomenda	A aplicação deverá permitir registar diferentes tipos de encomendas.	Alta
<b>US15</b>	Pesquisar despesas	A aplicação deverá permitir pesquisar por data, entre duas datas ou pelo tipo de despesa.	Alta
<b>US16</b>	Criar Listagem de despesas.	A aplicação deverá permitir criar listagem de despesas do mês corrente.	Alta
<b>US17</b>	Pesquisar um Fornecedor	A aplicação deverá permitir pesquisar um fornecedor quando está a fazer uma encomenda.	Alta

**Tabela 5 – Requisitos Funcionais do Administrador**

<b>Designação</b>	<b>Caso de Uso</b>	<b>Descrição</b>	<b>Prioridade</b>
<b>US18</b>	Registar Utente	A aplicação deverá permitir que um enfermeiro registe todos os dados de um utente.	Alta
<b>US19</b>	Imprimir o Consentimento do uso de Dados	A aplicação deverá permitir imprimir o consentimento de dados para que o utente possa assinar, para aceitar o uso dos seus dados.	Alta
<b>US20</b>	Criar Listagem dos Utentes	A aplicação deverá permitir listar todos os utentes daquele enfermeiro.	Alta
<b>US21</b>	Pesquisar Utentes	A aplicação deverá permitir pesquisar utentes pelo seu nome ou NIF.	Alta
<b>US22</b>	Criar Listagem dos detalhes dos Utentes	A aplicação deverá permitir listar as consultas realizadas, análises laboratoriais, avaliação objetivo, cirurgias, exames, doenças e alergias.	Alta
<b>US23</b>	Editar Dados do Utente	A aplicação deverá permitir editar os dados pessoais do utente.	Alta
<b>US24</b>	Marcar Consulta	A aplicação deverá permitir efetuar a marcação de consultas, numa data e numa determinada hora.	Alta
<b>US25</b>	Criar Listagem das consultas marcadas	A aplicação deverá permitir listar todas as consultas marcadas para aquele dia.	Alta
<b>US26</b>	Iniciar uma consulta sem marcação	A aplicação deverá permitir iniciar uma consulta a qualquer hora, sem marcação.	Alta
<b>US27</b>	Criar Listagem dos detalhes das	A aplicação deverá permitir listar os	Alta

	consultas	detalhes das consultas anteriores.	
<b>US28</b>	Registrar a Localização da dor durante a consulta	A aplicação deverá permitir registar a localização da dor através de uma imagem.	Alta
<b>US29</b>	Criar Listagem do registo da localização da dor durante a consulta	A aplicação deverá permitir listar todos os registos das localizações das dores anteriores.	Alta
<b>US30</b>	Registrar Doenças do utente	A aplicação deverá permitir registar doenças do utente e a data do diagnóstico.	Alta
<b>US31</b>	Criar Listagem das doenças	A aplicação deverá permitir listar as doenças registadas referentes a cada utente.	Alta
<b>US32</b>	Pesquisar Doenças	A aplicação deverá permitir pesquisar pelo nome as doenças já registadas.	Alta
<b>US33</b>	Registrar uma nova Doença	A aplicação deverá permitir adicionar os dados de uma doença, como o nome e os sintomas.	Alta
<b>US34</b>	Criar Listagem das doenças registadas	A aplicação deverá permitir listar as doenças registadas.	Alta
<b>US35</b>	Editar uma doença	A aplicação deverá permitir editar os dados de uma doença já registada.	Alta
<b>US36</b>	Registrar uma Alergia do utente	A aplicação deverá permitir registar as alergias do utente e a data do diagnóstico.	Alta
<b>US37</b>	Pesquisar Alergia	A aplicação deverá permitir pesquisar a alergia pelo nome.	Alta
<b>US38</b>	Registrar uma nova Alergia	A aplicação deverá permitir registar os dados de uma alergia, como o	Alta



		nome e os sintomas.	
<b>US39</b>	Criar Listagem das alergias registadas	A aplicação deverá permitir listar as doenças registadas.	Alta
<b>US40</b>	Editar uma alergia	A aplicação deverá permitir editar os dados de uma alergia já registada.	Alta
<b>US41</b>	Registar uma Cirurgia do utente	A aplicação deverá permitir registar cirurgias que o utente tenha realizado, a data e observações.	Alta
<b>US42</b>	Registar uma nova Cirurgia	A aplicação deverá permitir registar um tipo de cirurgia.	Alta
<b>US43</b>	Criar Listagem das cirurgias registadas	A aplicação deverá permitir listar todas as cirurgias que o utente tenha realizado.	Alta
<b>US44</b>	Pesquisar Cirurgia	A aplicação deverá permitir pesquisar a cirurgia pelo nome.	Alta
<b>US45</b>	Registar um exame do utente	A aplicação deverá permitir registar exames que o utente tenha realizado, data, designação e observações.	Alta
<b>US46</b>	Registar um novo Exame	A aplicação deverá permitir registar um tipo de exame.	Alta
<b>US47</b>	Criar listagem dos exames registados	A aplicação deverá permitir listar todos os exames que o utente tenha realizado.	Alta
<b>US48</b>	Pesquisar Exame	A aplicação deverá permitir pesquisar o exame pelo nome.	Alta
<b>US49</b>	Registar Avaliação objetiva do utente	A aplicação deverá permitir registar a avaliação objetiva do utente.	Alta
<b>US50</b>	Criar listagem das avaliações objetivo registadas.	A aplicação deverá permitir listar todas as avaliações objetivo do utente.	Alta

<b>US51</b>	Registrar Métodos Contracetivos	A aplicação deverá permitir registar vários métodos contracetivos.	Alta
<b>US52</b>	Registrar Análises Laboratoriais efetuadas pelo utente	A aplicação deverá permitir registar análises laboratoriais efetuadas pelo utente, a data em que foram feitas, resultados e observações.	Alta
<b>US53</b>	Criar listagem das análises laboratoriais registadas	A aplicação deverá permitir listar todas as análises laboratoriais efetuadas pelo utente.	Alta
<b>US54</b>	Registrar Análises Laboratoriais	A aplicação deverá permitir registar vários tipos de análises laboratoriais	Alta
<b>US55</b>	Registrar Tratamentos	A aplicação deverá permitir registar vários tipos de tratamentos efetuados pelo utente.	Alta
<b>US56</b>	Criar listagem dos detalhes dos tratamentos registados anteriormente.	A aplicação deverá permitir listar os detalhes dos tratamentos anteriores.	Alta
<b>US57</b>	Registrar Excisões	A aplicação deverá permitir registar os dados das excisões.	Alta
<b>US58</b>	Registrar Ferida Cirúrgica	A aplicação deverá permitir registar os detalhes da ferida cirúrgica.	Alta
<b>US59</b>	Registrar Ferida Traumática	A aplicação deverá permitir registar os detalhes da ferida traumática.	Alta
<b>US60</b>	Registrar Doppler arterial/venoso	A aplicação deverá permitir registar a localização onde foi feito este tratamento.	Alta
<b>US61</b>	Registrar Doppler	A aplicação deverá	Alta

	fetal	permitir registar os detalhes do doppler fetal.	
<b>US62</b>	Criar listagem do histórico do doppler fetal.	A aplicação deverá permitir listar todos os detalhes deste tratamento efetuado anteriormente.	Alta
<b>US63</b>	Registar Espirometria	A aplicação deverá permitir registar os detalhes da espirometria.	Alta
<b>US64</b>	Criar listagem do histórico da espirometria.	A aplicação deverá permitir listar todos os detalhes deste tratamento efetuado anteriormente.	Alta
<b>US65</b>	Registar onicocriptoses, onicomicoses e pé diabético.	A aplicação deverá permitir registar a localização destes tratamentos.	Alta
<b>US66</b>	Registar Vacinação	A aplicação deverá permitir registar todos os detalhes da vacinação efetuada.	Alta
<b>US67</b>	Criar listagem do histórico das vacinações	A aplicação deverá permitir listar todos os detalhes das vacinações do utente.	Alta
<b>US68</b>	Registar tipos de Queimaduras	A aplicação deverá permitir registar diferentes tipos de queimaduras.	Alta
<b>US69</b>	Criar listagem dos tipos de Queimaduras	A aplicação deverá permitir listar todos os tipos de queimaduras inseridos.	Alta
<b>US70</b>	Registar Queimadura Iónica	A aplicação deverá permitir registar os detalhes da queimadura iónica.	Alta
<b>US71</b>	Registar Queimadura Química	A aplicação deverá permitir registar os detalhes da queimadura química.	Alta
<b>US72</b>	Registar Queimadura Solar	A aplicação deverá permitir registar os detalhes da	Alta

		queimadura solar.	
<b>US73</b>	Registrar Queimadura Térmica	A aplicação deverá permitir registrar os detalhes da queimadura térmica.	Alta
<b>US74</b>	Registrar tipos de Úlceras	A aplicação deverá permitir registrar diferentes tipos de úlceras.	Alta
<b>US75</b>	Criar listagem dos tipos de Úlceras	A aplicação deverá permitir	Alta
<b>US76</b>	Registrar Úlcera Arterial	A aplicação deverá permitir registrar os detalhes da úlcera arterial.	Alta
<b>US77</b>	Registrar Úlcera Mista	A aplicação deverá permitir registrar os detalhes da úlcera mista.	Alta
<b>US78</b>	Registrar Úlcera de Pressão	A aplicação deverá permitir registrar os detalhes da úlcera de pressão.	Alta
<b>US79</b>	Registrar Úlcera Venosa	A aplicação deverá permitir registrar os detalhes da úlcera venosa.	Alta
<b>US80</b>	Registrar a medicação prescrita	A aplicação deverá permitir registrar a medicação prescrita na consulta, nome medicamento, outras indicações, período do dia e quantidade.	Alta
<b>US81</b>	Imprimir a medicação prescrita	A aplicação deverá permitir imprimir a medicação prescrita pelo enfermeiro.	Alta
<b>US82</b>	Criar listagem da medicação prescrita	A aplicação deverá permitir listar a medicação prescrita mais recente.	Alta
<b>US83</b>	Criar listagem do histórico da medicação prescrita.	A aplicação deverá permitir listar todo o histórico da medicação que já foi prescrita àquele utente.	Alta
<b>US84</b>	Pesquisar	A aplicação deverá	Alta

	Medicação prescrita	permitir pesquisar a medicação que já foi prescrita por data ou pelo nome do medicamento.	
<b>US85</b>	Registrar atitudes terapêuticas.	A aplicação deverá permitir registrar várias atitudes terapêuticas.	Alta
<b>US86</b>	Registrar colheitas.	A aplicação deverá permitir registrar a colheita de expetoração, exsudado zaragatoa, fezes parasitológico, fezes sangue oculto, sangue.	Alta
<b>US87</b>	Registrar outras atitudes terapêuticas.	A aplicação deverá permitir registrar a atitude enema limpeza e lavagem gástrica.	Alta
<b>US88</b>	Criar Listagem do histórico das atitudes terapêuticas.	A aplicação deverá permitir listar as atitudes terapêuticas realizadas pelo utente.	Alta
<b>US89</b>	Registrar a administração de medicação.	A aplicação deverá permitir registrar a administração de medicação efetuada ao utente.	Alta
<b>US90</b>	Criar Listagem do histórico da administração de medicação.	A aplicação deverá permitir listar os dados das medicações administradas ao utente.	Alta
<b>US91</b>	Registrar a algaliação.	A aplicação deverá permitir registrar a algaliação efetuada ao utente.	Alta
<b>US92</b>	Criar Listagem do histórico das algaliações.	A aplicação deverá permitir listar os dados das algaliações feitas ao utente.	Alta
<b>US93</b>	Registrar as aspirações de secreções.	A aplicação deverá permitir registrar as aspirações de	Alta

		secreções efetuadas ao utente.	
<b>US94</b>	Criar Listagem do histórico das aspirações de secreções.	A aplicação deverá permitir listar os dados das aspirações de secreções feitas ao utente.	Alta
<b>US95</b>	Registar dados no audiograma.	A aplicação deverá permitir registar círculos numa imagem do audiograma.	Alta
<b>US96</b>	Imprimir dados do audiograma.	A aplicação deverá permitir imprimir os detalhes do audiograma do utente.	Alta
<b>US97</b>	Registar cateterismo.	A aplicação deverá permitir registar os detalhes do cateterismo.	Alta
<b>US98</b>	Criar Listagem do histórico dos cateterismos.	A aplicação deverá permitir listar os detalhes cateterismos efetuados ao utente.	Alta
<b>US99</b>	Registar colheita de sangue de diagnóstico precoce.	A aplicação deverá permitir registar os dados da colheita de sangue precoce.	Alta
<b>US100</b>	Criar Listagem do histórico das colheitas de sangue precoce.	A aplicação deverá permitir listar os detalhes das colheitas de sangue precoce.	Alta
<b>US101</b>	Registar colheita de urina.	A aplicação deverá permitir registar os detalhes da colheita de urina efetuada ao utente.	Alta
<b>US102</b>	Criar Listagem do histórico das colheitas de urina.	A aplicação deverá permitir listar os detalhes das colheitas de urina anteriores.	Alta
<b>US103</b>	Registar dados da crioterapia.	A aplicação deverá permitir registar a localização onde foi aplicada a crioterapia.	Alta

<b>US104</b>	Criar Listagem do histórico da crioterapia.	A aplicação deverá permitir listar a localização e observações da crioterapia.	Alta
<b>US105</b>	Registrar Drenagem de Locas	A aplicação deverá permitir registrar os detalhes da drenagem de locas.	Alta
<b>US106</b>	Criar Listagem do histórico da drenagem de locas.	A aplicação deverá permitir listar os detalhes das drenagens de locas efetuadas ao utente.	Alta
<b>US107</b>	Registrar Desbridamento.	A aplicação deverá permitir registrar os detalhes do desbridamento realizado ao utente.	Alta
<b>US108</b>	Criar Listagem do histórico do desbridamento.	A aplicação deverá permitir listar os detalhes do desbridamento.	Alta
<b>US109</b>	Registrar ENG.	A aplicação deverá permitir registrar os dados do ENG realizado ao utente.	Alta
<b>US110</b>	Criar Listagem do histórico de ENG.	A aplicação deverá permitir listar os detalhes dos ENG efetuados.	Alta
<b>US111</b>	Registrar Flebografia.	A aplicação deverá permitir registrar os dados da flebografia realizada ao utente.	Alta
<b>US112</b>	Criar Listagem do histórico das flebografias.	A aplicação deverá permitir listar os detalhes das flebografias anteriores.	Alta
<b>US113</b>	Registrar Inalações.	A aplicação deverá permitir registrar os dados das inalações efetuadas ao utente.	Alta
<b>US114</b>	Criar Listagem do histórico das inalações.	A aplicação deverá permitir listar os detalhes das inalações anteriores.	Alta
<b>US115</b>	Registrar lavagem auricular.	A aplicação deverá permitir registrar os	Alta

		dados da lavagem auricular efetuada ao utente.	
<b>US116</b>	Criar Listagem do histórico das lavagens auriculares.	A aplicação deverá permitir listar as lavagens auriculares anteriores.	Alta
<b>US117</b>	Registar lavagem ocular.	A aplicação deverá permitir registar os dados da lavagem ocular.	Alta
<b>US118</b>	Criar Listagem do histórico das lavagens oculares.	A aplicação deverá permitir listar as lavagens oculares anteriores.	Alta
<b>US119</b>	Registar Lavagem Vesical.	A aplicação deverá permitir registar os dados da lavagem vesical.	Alta
<b>US120</b>	Criar Listagem do histórico das lavagens vesicais.	A aplicação deverá permitir listar as lavagens vesicais anteriores.	Alta
<b>US121</b>	Registar Monitorização ECG	A aplicação deverá permitir registar os dados da monitorização ECG.	Alta
<b>US122</b>	Criar Listagem do histórico da monitorização ECG.	A aplicação deverá permitir listar as monitorizações ECG anteriores.	Alta
<b>US123</b>	Registar Pressoterapia.	A aplicação deverá permitir registar os dados da Pressoterapia efetuada ao utente.	Alta
<b>US124</b>	Criar Listagem do histórico da Pressoterapia.	A aplicação deverá permitir listar as pressoterapias anteriores.	Alta
<b>US125</b>	Registar Sutura	A aplicação deverá permitir registar os detalhes da sutura efetuada ao utente.	Alta
<b>US126</b>	Criar Listagem do histórico das suturas.	A aplicação deverá permitir listar as suturas anteriores.	Alta
<b>US127</b>	Registar Teste <i>Combur</i>	A aplicação deverá permitir registar os resultados do teste	Alta



		de <i>Combur</i> .	
<b>US128</b>	Criar Listagem do histórico dos testes de <i>Combur</i> .	A aplicação deverá permitir listar os dados do teste de <i>Combur</i> anteriores.	Alta
<b>US129</b>	Registrar Teste de avaliação acuidade visual.	A aplicação deverá permitir registar os resultados do teste de avaliação acuidade visual efetuado ao utente.	Alta
<b>US130</b>	Criar Listagem do histórico dos testes de avaliação acuidade visual.	A aplicação deverá permitir listar os testes de avaliação acuidade visual anteriores.	Alta
<b>US131</b>	Registrar Tricotomia	A aplicação deverá permitir registar os dados da tricotomia efetuada ao utente.	Alta
<b>US132</b>	Criar Listagem do histórico da tricotomia.	A aplicação deverá permitir listar os dados das tricotomias anteriores.	Alta
<b>US133</b>	Registrar Zaragatoa Orofaringe	A aplicação deverá permitir registar os dados da Zaragatoa Orofaringe.	Alta
<b>US134</b>	Criar Listagem do histórico da zaragatoa orofaringe.	A aplicação deverá permitir listar os dados das zaragotas orofaringe anteriores.	Alta
<b>US135</b>	Registrar Colpocitologia	A aplicação deverá permitir registar os dados da Colpocitologia efetuada à utente.	Alta
<b>US136</b>	Criar Listagem do histórico das colpocitologias.	A aplicação deverá permitir listar as colpocitologias anteriores.	Alta
<b>US137</b>	Registrar colocação de DIU	A aplicação deverá permitir registar os dados da colocação do DIU efetuado à utente.	Alta
<b>US138</b>	Criar Listagem do histórico das	A aplicação deverá permitir listar as	Alta

	colocações de DIU.	colocações de DIU anteriores.	
<b>US139</b>	Registar Implante Contracetivo Dérmico.	A aplicação deverá permitir registar os dados do implante contracetivo dérmico.	Alta
<b>US140</b>	Criar Listagem do histórico dos implantes contracetivos dérmicos.	A aplicação deverá permitir listar as colocações de implantes contracetivos dérmicos anteriores.	Alta
<b>US141</b>	Registar avaliação objetiva do bebé até 36 meses.	A aplicação deverá permitir registar os dados da avaliação objetiva do bebé.	Alta
<b>US142</b>	Criar Listagem do histórico das avaliações objetivas do bebé.	A aplicação deverá permitir listar as avaliações objetivas anteriores do bebé.	Alta
<b>US143</b>	Registar um tipo de Aleitamento.	A aplicação deverá permitir registar um tipo de aleitamento e observações.	Alta
<b>US144</b>	Criar Listagem dos tipos de Aleitamento	A aplicação deverá permitir listar os tipos de aleitamento registados.	Alta
<b>US145</b>	Editar Tipo de Aleitamento.	A aplicação deverá permitir editar as observações de um tipo de aleitamento.	Alta
<b>US146</b>	Pesquisar um tipo de Aleitamento.	A aplicação deverá permitir pesquisar um tipo de aleitamento pelo nome.	Alta
<b>US147</b>	Registar Aleitamento materno.	A aplicação deverá permitir registar o tipo de aleitamento materno e observações.	Alta
<b>US148</b>	Registar Aleitamento Artificial.	A aplicação deverá permitir registar o tipo de aleitamento artificial e observações.	Alta
<b>US149</b>	Registar Aleitamento	A aplicação deverá permitir registar o	Alta

	Misto.	tipo de aleitamento misto.	
<b>US150</b>	Registrar um tipo de parto.	A aplicação deverá permitir registrar um tipo de parto e observações.	Alta
<b>US151</b>	Registrar eutócico.	A aplicação deverá permitir registrar um parto do tipo eutócico.	Alta
<b>US152</b>	Registrar distócico.	A aplicação deverá permitir registrar um parto do tipo distócico, com 2 subtipos: fórceps e ventosa.	Alta
<b>US153</b>	Registrar cesariana.	A aplicação deverá permitir registrar um parto do tipo cesariana.	Alta
<b>US154</b>	Editar tipo de parto.	A aplicação deverá permitir editar as observações de um tipo de parto.	Alta
<b>US155</b>	Pesquisar um tipo de parto.	A aplicação deverá permitir pesquisar um tipo de parto pelo nome.	Alta
<b>US156</b>	Criar Listagem dos tipos de Parto	A aplicação deverá permitir listar os tipos de parto existentes.	Alta
<b>US157</b>	Registrar os produtos utilizados numa consulta.	A aplicação deverá permitir registrar os produtos utilizados numa consulta.	Alta
<b>US158</b>	Registrar os produtos utilizados numa consulta associados a um fornecedor.	A aplicação deverá permitir registrar os produtos utilizados associados a um fornecedor.	Alta
<b>US159</b>	Criar Listagem dos fornecedores.	A aplicação deverá permitir listar os fornecedores existentes.	Alta
<b>US160</b>	Criar Listagem dos produtos em <i>stock</i> .	A aplicação deverá permitir listar os produtos em <i>stock</i> do fornecedor	Alta

		selecionado e a quantidade de cada produto.	
<b>US161</b>	Criar listagem dos produtos utilizados na consulta.	A aplicação deverá permitir listar os produtos utilizados na consulta.	Alta
<b>US162</b>	Registar a quantidade utilizada de cada produto.	A aplicação deverá permitir registar a quantidade utilizada de cada produto na consulta.	Alta
<b>US163</b>	Decrementar os produtos utilizados.	A aplicação deverá permitir decrementar a quantidade de produtos no stock existente na clínica.	Alta
<b>US164</b>	Retirar produtos usados na consulta.	A aplicação deverá permitir retirar um produto que não tenha sido utilizado na consulta.	Alta
<b>US165</b>	Registar o valor da consulta.	A aplicação deverá permitir registar o valor da consulta.	Alta
<b>US166</b>	Alterar hora ou data de uma consulta.	A aplicação deverá permitir alterar a hora ou data de uma consulta.	Alta
<b>US167</b>	Pesquisar Consultas	A aplicação deverá permitir pesquisar consultas registadas por data, nome do utente ou NIF do utente.	Alta
<b>US168</b>	Criar Listagem das consultas registadas.	A aplicação deverá permitir listar as consultas registadas.	Alta
<b>US169</b>	Desmarcar consulta agendada.	A aplicação deverá permitir desmarcar consulta agendada.	Alta
<b>US170</b>	Editar dados do enfermeiro.	A aplicação deverá permitir alterar os dados do enfermeiro com sessão iniciada.	Alta
<b>US171</b>	Alterar password do enfermeiro.	A aplicação deverá permitir alterar password do enfermeiro com sessão iniciada.	Alta

<b>US172</b>	Criar listagem das consultas do dia.	A aplicação deverá permitir listar as consultas do dia e iniciar consulta.	Alta
<b>US173</b>	Registrar Tipo de Atitude terapêutica.	A aplicação deverá permitir registrar um tipo de atitude terapêutica.	Alta
<b>US174</b>	Registrar um método contraceptivo.	A aplicação deverá permitir registrar um método contraceptivo e observações.	Alta
<b>US175</b>	Criar listagem dos métodos contraceptivos.	A aplicação deverá permitir listar os métodos contraceptivos registrados.	Alta
<b>US176</b>	Editar métodos contraceptivos.	A aplicação deverá permitir editar os métodos contraceptivos, o nome e observações.	Alta
<b>US177</b>	Pesquisar método contraceptivo	A aplicação deverá permitir pesquisar um método contraceptivo pelo nome.	Alta
<b>US178</b>	Registrar profissões.	A aplicação deverá permitir registrar uma profissão.	Alta
<b>US179</b>	Criar Listagem das profissões.	A aplicação deverá permitir listar as profissões registradas.	Alta
<b>US180</b>	Editar profissões.	A aplicação deverá permitir editar o nome das profissões.	Alta
<b>US181</b>	Pesquisar profissões.	A aplicação deverá permitir pesquisar a profissão pelo nome.	Alta
<b>US182</b>	Recuperar <i>username</i> do utilizador.	A aplicação deverá permitir recuperar o <i>username</i> do utilizador através do Gmail.	Alta
<b>US183</b>	Enviar e-mail ao utilizador, para recuperar o <i>username</i> .	A aplicação deverá permitir enviar um email ao utilizador para que este receba	Alta

		um código para alterar o <i>username</i> .	
<b>US184</b>	Recuperar <i>password</i> do utilizador.	A aplicação deverá permitir recuperar a <i>password</i> do utilizador através do Gmail.	Alta
<b>US185</b>	Enviar e-mail ao utilizador, para recuperar a <i>password</i> .	A aplicação deverá permitir enviar um email ao utilizador para que este receba um código para alterar a <i>password</i> .	Alta
<b>US186</b>	<i>Login</i>	A aplicação deverá permitir aos vários tipos de utilizador efetuar o <i>login</i> .	Alta
<b>US187</b>	Primeiro Acesso	A aplicação deverá permitir que o enfermeiro ao iniciar sessão pela primeira vez possa definir a sua <i>password</i> e o seu <i>username</i> .	Alta
<b>US188</b>	Denições de Utilizador	A aplicação tem de permitir ao enfermeiro editar os seus dados pessoais.	Alta
<b>US189</b>	Denições de Utilizador	A aplicação tem de permitir ao enfermeiro alterar a sua <i>password</i> .	Alta
<b>US190</b>	Iniciar uma consulta com marcação	A aplicação deverá permitir iniciar uma consulta a um dia/hora marcado,	Alta
<b>US191</b>	Ver consultas do próprio dia	A aplicação deverá permitir que o enfermeiro visualize as consultas do dia atual no menu principal.	Alta

Tabela 6 - Requisitos Funcionais do Administrador e Utilizador Normal

## 15.2. Anexo B - Mockups

Neste Anexo é possível ver as *mockups* realizadas para o projeto, para que o administrador da clínica pudesse ter uma noção de como seria a aplicação.

The mockup shows a receipt form with two columns. The left column is titled 'Despesas da Clínica:' and lists items like 'luz - 100€', 'água - 80€', 'medicamentos', and 'terapias'. The right column is titled 'Receitas Recebidas' and lists items like 'ultrassom - 50€' and 'ultrassom - 30€'. Both columns have a 'Total: \_\_\_\_\_' line at the bottom. A 'Saldo Total: \_\_\_\_\_' line is at the very bottom.

**Figura 4 – mockup das despesas e receitas**

The mockup shows a user registration form. It includes fields for 'Nome Completo:', 'Data Nascimento:' (with a calendar icon), 'Sexo:' (with radio buttons for 'M' and 'F'), 'Morada:', 'Contacto:', 'E-mail:', and 'Uf:'. There is a 'Profissão:' section with a list of professions: Assistente Social, Bibliotecário, Bombeiro, Cuidador, Enfermeiro, Dentista, Economista, Eletricista, Fisioterapeuta, Farmacêutico, Médico, and Odontólogo. At the bottom, there are two buttons: 'Guardar' (with a green checkmark) and 'Cancelar' (with a red X).

**Figura 5 – mockup do registo dos Utentes**

### 15.3. Anexo C – Base de Dados

A base de dados utilizada no projeto desenvolvido, foi desenvolvida em SQL, foi criada localmente devido ao cliente ainda não saber que tipo de base de dados quer usar para armazenar os dados. Futuramente será adaptada para o que o cliente desejar.




#### Detalhe das tabelas

A base de dados tem o nome ‘SiltesSaude’ e é composta por 71 tabelas. Que são detalhadas uma a uma de seguida.

- **AdministrarMedicacao:** esta tabela armazena informação relativa aos medicamentos e a forma como o enfermeiro os vai administrar no paciente. A tabela possui como chave primária três campos: IdAtitude, IdPaciente e data, para que a um determinado paciente possa ser administrado a medicação em datas diferentes. Como chave estrangeira possui dois campos: IdAtitude e IdPaciente. A tabela possui os seguintes atributos:
  - **IdAtitude:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdAtitude da tabela Atitude (a atitude ‘Administrar Medicação’ pode ser administrada diversas vezes). Regista o id’s da atitude ‘Administrar Medicação’ da tabela Atitude. O atributo é do tipo inteiro e é obrigatório.
  - **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode ter várias administrações de medicamentos). Regista os id’s dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - **data:** é chave primária. Regista a data que o enfermeiro escolhe para administração da medicação. O atributo é do tipo data e é obrigatório.
  - **PO:** (pela boca) regista se a administração da medicação é de forma oral. O atributo é do tipo *string* com tamanho 50, e não é obrigatório.
  - **retal:** (pela boca) regista se a administração da medicação é de forma retal. O atributo é do tipo *string* com tamanho 50, e não é obrigatório.




- **intradérmica:** tipo de administrar a medicação. O enfermeiro regista se é este tipo de administração ou não. O atributo é do tipo *string*, com tamanho 50 e não é obrigatório.
- **intramuscular:** tipo de administrar a medicação. O enfermeiro regista se é este tipo de administração ou não. O atributo é do tipo *string*, com tamanho 50 e não é obrigatório.
- **endovenosa:** tipo de administrar a medicação. O enfermeiro regista se é este tipo de administração ou não. O atributo é do tipo *string*, com tamanho 50 e não é obrigatório.
- **subcutânea:** tipo de administrar a medicação. O enfermeiro regista se é este tipo de administração ou não. O atributo é do tipo *string*, com tamanho 50 e não é obrigatório.
- **topicoViaCutânea:** administração da medicação do tipo tópico que pode ser Via Cutânea. O enfermeiro regista se é este tipo de administração ou não. O atributo é do tipo *string*, com tamanho 50 e não é obrigatório.
- **topicoEfeitoLocal:** administração da medicação do tipo tópico que pode ser de Efeito Local. O enfermeiro regista se é este tipo de administração ou não. O atributo é do tipo *string*, com tamanho 50 e não é obrigatório.
- **Observações:** escrever observações. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

<b>AdministrarMedicacao</b>			
	Column Name	Data Type	Allow Nulls
	IdAtitude	int	<input type="checkbox"/>
	IdPaciente	int	<input type="checkbox"/>
	data	date	<input type="checkbox"/>
	PO	nvarchar(50)	<input checked="" type="checkbox"/>
	retal	nvarchar(50)	<input checked="" type="checkbox"/>
	intradermica	nvarchar(50)	<input checked="" type="checkbox"/>
	intramuscular	nvarchar(50)	<input checked="" type="checkbox"/>
	endovenosa	nvarchar(50)	<input checked="" type="checkbox"/>
	subcutanea	nvarchar(50)	<input checked="" type="checkbox"/>
	topicoViaCutanea	nvarchar(50)	<input checked="" type="checkbox"/>
	topicoEfeitoLocal	nvarchar(50)	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 6 - tabela AdministrarMedicacao**


- **AgendamentoConsulta:** esta tabela armazena informação relativa a marcação das consultas. A tabela tem como chave primária o atributo IdMarcacao. A tabela possui 2 campos como chave estrangeira: IdPaciente e IdEnfermeiro. A tabela possui os seguintes atributos:
  - **IdMarcacao:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **horaProximaConsulta:** guarda a hora da marcação. O atributo é do tipo string com tamanho máximo e é obrigatório.
  - **dataProximaConsulta:** guarda a data da consulta marcada. O atributo é do tipo data e é obrigatório.
  - **IdPaciente:** é chave estrangeira, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode ter diversas consultas). Regista os id's dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - **IdEnfermeiro:** é chave estrangeira, tem uma relação de 1 para N com o atributo IdEnfermeiro da tabela Enfermeiro (um enfermeiro pode ter várias consultas para realizar).
  - **ConsultaRealizada:** o campo regista se a consulta foi ou não realizada. Quando se marca a consulta o valor do campo é FALSE. Quando se realiza

a consulta e é dada como terminada o valor do campo passa a TRUE. O atributo é do tipo *boolean*, e é obrigatório.

AgendamentoConsulta			
	Column Name	Data Type	Allow Nulls
	IdMarcacao	int	<input type="checkbox"/>
	horaProximaConsulta	nvarchar(MAX)	<input type="checkbox"/>
	dataProximaConsulta	date	<input type="checkbox"/>
	IdPaciente	int	<input type="checkbox"/>
	IdEnfermeiro	int	<input type="checkbox"/>
	ConsultaRealizada	bit	<input type="checkbox"/>
			<input type="checkbox"/>

**Figura 7 - tabela AgendamentoConsulta**

- **Aleitamento:** esta tabela armazena informação relativa aos tipos de aleitamento possíveis num bebé. A tabela tem como chave primária o atributo IdAleitamento. A tabela possui os seguintes atributos:
  - **IdAleitamento:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **tipoAleitamento:** regista os vários tipos de aleitamento que podem existir. O atributo é do tipo *string*, tem tamanho 50 e é obrigatório.
  - **Observacoes:** regista as observações que o enfermeiro quiser registar. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.

Aleitamento			
	Column Name	Data Type	Allow Nulls
	IdAleitamento	int	<input type="checkbox"/>
	tipoAleitamento	nvarchar(50)	<input type="checkbox"/>
	Observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 8 - tabela Aleitamento**

- **Alergia:** esta tabela armazena informação relativa aos tipos de alergias possíveis. A tabela tem como chave primária o atributo IdAlergia. A tabela possui os seguintes atributos:
  - **IdAlergia:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **Nome:** o enfermeiro regista os nomes das alergias que existem. O atributo é do tipo *string* com tamanho máximo e é obrigatório.
  - **Sintomas:** o enfermeiro regista os sintomas que determinada alergia tem nos pacientes. O atributo é do tipo *string* com tamanho máximo e não é obrigatório.

Alergia			
	Column Name	Data Type	Allow Nulls
🔑	IdAlergia	int	<input type="checkbox"/>
	Nome	nvarchar(MAX)	<input type="checkbox"/>
	Sintomas	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 9 - tabela Alergia**

- **AlergiaPaciente:** esta tabela armazena informação relativa aos tipos de alergias que um paciente pode ter. A tabela tem como chave primária os atributos IdAlergia, IdPaciente e data. A tabela possui 2 campos como chave estrangeira: IdPaciente e IdAlergia. A tabela possui os seguintes atributos:
  - **IdAlergia:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdAlergia da tabela Alergia (uma alergia pode ser registada diversas vezes, em diversos pacientes). Regista os id's das atitudes da tabela Alergia. O atributo é do tipo inteiro e é obrigatório.
  - **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode ter várias alergias). Regista os id's dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - **data:** é chave primária. Regista a data que o enfermeiro escolhe para registo ou descoberta da alergia. O atributo é do tipo data e é obrigatório.

- **observações:** o enfermeiro regista observações relativas às alergias do paciente. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

AlergiaPaciente			
	Column Name	Data Type	Allow Nulls
🔑	IdAlergia	int	<input type="checkbox"/>
🔑	IdPaciente	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 10 – tabela AlergiaPaciente**

- **Algaliacao:** esta tabela armazena informação relativa aos tipos de algaliação possíveis. Algaliação é se o paciente precisa de usar um saco para as suas necessidades fisiológicas. A tabela tem como chave primária três atributos: IdAtitude, IdPaciente e data. Como chaves estrangeiras tem os atributos IdAtitude e IdPaciente. A tabela possui os seguintes atributos:
  - **IdAtitude:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdAtitude da tabela Atitude (a atitude ‘Algaliação’ atitude pode ser administrada diversas vezes). Regista o id’s da atitude ‘Algaliação’ da tabela Atitude. O atributo é do tipo inteiro e é obrigatório.
  - **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode ter várias algaliações). Regista os id’s dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - **data:** é chave primária. Regista a data que o enfermeiro escolhe para registo ou da algaliação. O atributo é do tipo data e é obrigatório.
  - **silastic:** regista a percentagem do tipo de material silicone. O atributo é do tipo inteiro, e não é obrigatório.
  - **folley:** regista se o material é *folley* ou não (borracha). O atributo é do tipo *string* com tamanho 30, e não é obrigatório.

- **tresVias:** regista se o material é algália (3 pontas) ou não (algália é uma sonda vesical, ou seja, um tubo ou sistema de tubos que se utiliza para esvaziar e recolher a urina da bexiga). O atributo é do tipo *string* com tamanho 30, e não é obrigatório.
- **dataProximaRealgaliacao:** o enfermeiro regista a data da próxima algáliação que o paciente precisa de fazer. O atributo é do tipo *date* e não é obrigatório.
- **observações:** o enfermeiro regista observações relativas às alergias do paciente. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

Algaliacao			
	Column Name	Data Type	Allow Nulls
🔑	IdAtitude	int	<input type="checkbox"/>
🔑	IdPaciente	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
	silastic	int	<input checked="" type="checkbox"/>
	folley	nvarchar(30)	<input checked="" type="checkbox"/>
	tresVias	nvarchar(30)	<input checked="" type="checkbox"/>
	dataProximaRealgaliacao	date	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 11 - tabela Algaliacao**

- **analisesLaboratoriais:** esta tabela armazena informação relativa aos tipos de análises laboratoriais possíveis. A tabela tem como chave primária o atributo IdAnalisesLaboratoriais. A tabela possui os seguintes atributos:
  - **IdAnalisesLaboratoriais:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **NomeAnalise:** o enfermeiro regista os nomes das análises laboratoriais que existem. O atributo é do tipo *string* com tamanho máximo e é obrigatório.
  - **Observacoes:** o enfermeiro regista observações relativas a análise laboratorial que deseja. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

analisesLaboratoriais			
	Column Name	Data Type	Allow Nulls
🔑	IdAnalisesLaboratoriais	int	<input type="checkbox"/>
	NomeAnalise	nvarchar(MAX)	<input type="checkbox"/>
	Observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 12- tabela analisesLaboratoriais**

- **analisesLaboratoriaisPaciente:** esta tabela armazena informação relativa às análises laboratoriais que um paciente realizou. A tabela tem como chave primária os atributos IdAlergia, IdPaciente e data. A tabela possui 2 campos como chave estrangeira: IdPaciente e IdAlergia. A tabela possui os seguintes atributos:
  - **IdAnalisesLaboratoriais:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdAnalisesLaboratoriais da tabela analisesLaboratoriais (uma análise laboratorial pode ser registada diversas vezes, em diversos pacientes). Regista os id's das atitudes da tabela Alergia. O atributo é do tipo inteiro e é obrigatório.
  - **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode realizar várias análises Laboratoriais). Regista os id's dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - **data:** é chave primária. Enfermeiro regista a data em que o paciente realizou as analises laboratoriais. O atributo é do tipo data e é obrigatório.
  - **resultados:** o enfermeiro regista os resultados das análises laboratoriais do paciente. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório. O campo não é obrigatório porque nem sempre são registados os resultados. As vezes o enfermeiro apenas pretende registar as análises que foram realizadas.

- **observações:** o enfermeiro regista observações relativas às análises laboratoriais do paciente. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

analisesLaboratoriaisPaciente			
	Column Name	Data Type	Allow Nulls
🔑	IdAnalisesLaboratoriais	int	<input type="checkbox"/>
🔑	IdPaciente	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
	resultados	nvarchar(MAX)	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 13 - tabela analisesLaboratoriaisPaciente**

- **AspiracaoSecrecao:** esta tabela armazena informação relativa à Aspiração de Secreção (sucção de fluidos) que um paciente realizou. A tabela tem como chave primária os atributos IdAtitude, IdPaciente e data. A tabela possui 2 campos como chave estrangeira: IdAtitude e IdPaciente. A tabela possui os seguintes atributos:
  - **IdAtitude:** IdAtitude: é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdAtitude da tabela Atitude (a atitude ‘Aspiração Secreção’ pode ser administrada diversas vezes). Regista o id’s da atitude ‘Aspiração Secreção’ da tabela Atitude. O atributo é do tipo inteiro e é obrigatório.
  - **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode realizar várias análises Laboratoriais). Regista os id’s dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - **data:** é chave primária. Enfermeiro regista a data em que o paciente realizou a aspiração de secreção. O atributo é do tipo data e é obrigatório.
  - **aspiracao:** o enfermeiro regista a secção de fluídos feita. O atributo é do tipo *string* com tamanho máximo e não é obrigatório.




- **observações:** o enfermeiro regista observações ao procedimento realizado ao paciente. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

AspiracaoSecrecao			
	Column Name	Data Type	Allow Nulls
🔑	IdAtitude	int	<input type="checkbox"/>
🔑	IdPaciente	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
	aspiracao	nvarchar(MAX)	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 14 – tabela AspiracaoSecrecao**

- **Atitude:** esta tabela armazena informação relativa aos tipos de atitudes terapêuticas possíveis. A tabela tem como chave primária o atributo IdAtitude. Neste caso, o cliente pediu atitudes terapêuticas específicas, com campos específicos, mas esta tabela foi criada a pensar na possibilidade de poder haver mais atitudes terapêuticas.
- A tabela possui os seguintes atributos:
  - **IdAtitude:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **nomeAtitude:** o enfermeiro regista os nomes das atitudes terapêuticas que existem. O atributo é do tipo *string* com tamanho máximo e é obrigatório.
  - **Observacoes:** o enfermeiro regista observações relativas a análise laboratorial que deseje. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

Atitude			
	Column Name	Data Type	Allow Nulls
	IdAtitude	int	<input type="checkbox"/>
	nomeAtitude	nvarchar(MAX)	<input type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 15 - tabela Atitude**

- AvaliacaoObjetiva:** esta tabela armazena informação relativa a avaliação objetiva de um paciente com mais de 36 meses. A tabela tem como chave primária o atributo IdAvaliacaoObjetiva. A tabela possui 2 campos como chave estrangeira: IdPaciente. E IdMetodoContracetivo A tabela possui os seguintes atributos:
  - IdAvaliacaoObjetiva:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - data:** guarda a data de realização da avaliação objetivo. O atributo é do tipo data e é obrigatório.
  - peso:** regista o peso do paciente. O atributo é do tipo numérico, com tamanho de 5 dígitos e 2 casas decimais, e é obrigatório.
  - altura:** regista a altura do paciente. O atributo é do tipo inteiro, e é obrigatório.
  - IdPaciente:** é chave estrangeira, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode realizar várias avaliações objetivo). Regista os id's dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - pressaoArterial:** regista a pressão arterial paciente. O atributo é do tipo inteiro e não é obrigatório.
  - frequenciaCardiaca:** regista a frequência cardíaca do paciente. O atributo é do tipo inteiro e não é obrigatório.
  - temperatura:** regista a temperatura corporal do paciente. O atributo é do tipo numérico, com tamanho de 5 dígitos e 2 casas decimais, e não é obrigatório.

- **saturacaoOxigenio**: regista a saturação do sangue no oxigénio. O atributo é do tipo inteiro e não é obrigatório.
- **dataUltimaMestruacao**: regista a data da última menstruação do paciente. O atributo é do tipo data e não é obrigatório. (este atributo é só para pacientes em que o sexo é do tipo 'Feminino').
- **menopausa**: regista com que idade o paciente teve a menopausa. O atributo é do tipo inteiro, e não é obrigatório. (este atributo é só para pacientes em que o sexo é do tipo 'Feminino').
- **IdMetodoContracetivo**: é chave estrangeira, tem uma relação de 1 para N com o atributo IdMetodoContracetivo da tabela MetodoContracetivo (um paciente pode usar vários métodos contraceptivos ao longo da vida). Regista os id's dos Metodos Contraceptivos da tabela MetodoContracetivo. O atributo é do tipo inteiro e não é obrigatório.
- **DIU**: registar se o paciente usa DIU ou não. O atributo é do tipo string, com tamanho 10, e não é obrigatório. (este atributo é só para pacientes em que o sexo é do tipo 'Feminino').
- **concentracaoGlicoseSangue**: regista a concentração de glicose no sangue do paciente. O atributo é do tipo inteiro e não é obrigatório. (este atributo é só para pacientes em que o sexo é do tipo 'Feminino').
- **AC**: regista a auscultação cardíaca. O atributo é do tipo inteiro, e não é obrigatório. (este atributo é só para pacientes em que o sexo é do tipo 'Feminino').
- **AP**: regista a auscultação pulmonar. O atributo é do tipo inteiro, e não é obrigatório. (este atributo é só para pacientes em que o sexo é do tipo 'Feminino').
- **INR**: regista o tempo necessário para o sangue coagular relativamente a um valor médio. O atributo é do tipo inteiro, e não é obrigatório. (este atributo é só para pacientes em que o sexo é do tipo 'Feminino').
- **Menarca**: regista com que idade o paciente teve a primeira menstruação. O atributo é do tipo inteiro, e não é obrigatório. (este atributo é só para pacientes em que o sexo é do tipo 'Feminino').
- **Gravidez**: regista quantas gravidezes o paciente já teve. O atributo é do tipo inteiro, e não é obrigatório. (este atributo é só para pacientes em que o sexo é do tipo 'Feminino').

- **filhosVivos**: regista quantos filhos vivos o paciente já tem. O atributo é do tipo inteiro, e não é obrigatório. (este atributo é só para pacientes em que o sexo é do tipo 'Feminino').
- **abortos**: regista quantos abortos o paciente já teve. O atributo é do tipo inteiro, e não é obrigatório. (este atributo é só para pacientes em que o sexo é do tipo 'Feminino').
- **observações**: escrever observações relativas à avaliação objetivo. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

AvaliacaoObjetiva			
	Column Name	Data Type	Allow Nulls
?	IdAvaliacaoObjetiva	int	<input type="checkbox"/>
	data	date	<input type="checkbox"/>
	peso	numeric(5, 2)	<input type="checkbox"/>
	altura	int	<input type="checkbox"/>
	IdPaciente	int	<input type="checkbox"/>
	pressaoArterial	int	<input type="checkbox"/>
	frequenciaCardiaca	int	<input checked="" type="checkbox"/>
	temperatura	numeric(5, 2)	<input checked="" type="checkbox"/>
	saturacaoOxigenio	int	<input checked="" type="checkbox"/>
	dataUltimaMestruacao	date	<input checked="" type="checkbox"/>
	menopausa	int	<input checked="" type="checkbox"/>
	IdMetodoContraceptivo	int	<input checked="" type="checkbox"/>
	DIU	nvarchar(10)	<input checked="" type="checkbox"/>
	concentracaoGlicoseSan...	int	<input checked="" type="checkbox"/>
	AC	int	<input checked="" type="checkbox"/>
	AP	int	<input checked="" type="checkbox"/>
	INR	int	<input checked="" type="checkbox"/>
	Menarca	int	<input checked="" type="checkbox"/>
	gravidez	int	<input checked="" type="checkbox"/>
	filhosVivos	int	<input checked="" type="checkbox"/>
	abortos	int	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Figura 16 - tabela AvaliacaoObjetivo

- **AvaliacaoObjetivaBebe:** esta tabela armazena informação relativa a avaliação objetiva de um paciente com menos de 36 meses. A tabela tem como chave primária o atributo **IdAvaliacaoObjetivaBebe**. A tabela possui 3 campos como chave estrangeira: **IdPaciente**, **IdTipoAleitamento** e **IdTipoParto**. A tabela possui os seguintes atributos:
  - **IdAvaliacaoObjetivaBebe:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **dataRegistro:** guarda a data de realização da avaliação objetivo. O atributo é do tipo data e é obrigatório.
  - **Peso:** regista o peso do paciente. O atributo é do tipo numérico, com tamanho de 5 dígitos e 2 casas decimais, e é obrigatório.
  - **Altura:** regista a altura do paciente. O atributo é do tipo inteiro, e é obrigatório.
  - **pressaoArterial:** regista a pressão arterial paciente. O atributo é do tipo inteiro e não é obrigatório.
  - **frequenciaCardiaca:** regista a frequência cardíaca do paciente. O atributo é do tipo inteiro e não é obrigatório.
  - **temperatura:** regista a temperatura corporal do paciente. O atributo é do tipo numérico, com tamanho de 5 dígitos e 2 casas decimais, e não é obrigatório.
  - **saturacaoOxigenio:** regista a saturação do sangue no oxigénio. O atributo é do tipo inteiro e não é obrigatório.
  - **INR:** regista o ‘international normalized’. O atributo é do tipo inteiro, e não é obrigatório.
  - **Perimetro:** regista o perímetro do paciente. O atributo é do tipo inteiro, e não é obrigatório.
  - **IdTipoAleitamento:** é chave estrangeira, tem uma relação de 1 para N com o atributo **IdAleitamento** da tabela **Aleitamento**. (um paciente pode ter vários tipos de aleitamento). Regista os id’s do Tipo de Aleitamento da tabela **Aleitamento**. O atributo é do tipo inteiro e não é obrigatório.
  - **nomeLeiteArtificial:** regista o nome do leite quando é do tipo ‘Artificial’. O atributo é do tipo *string*, e não é obrigatório.

- **IdTipoParto:** é chave estrangeira, tem uma relação de com o atributo IdParto da tabela Parto. Regista os id's do Tipo de parto da tabela Parto. O atributo é do tipo inteiro e não é obrigatório.
- **partoDistocico:** regista se o parto do tipo 'Distócico' foi feito com ventosas ou fórceps. O atributo é do tipo *string* com tamanho 20, e não é obrigatório.
- **epidural:** regista se o parto do bebe teve epidural ou não. O atributo é do tipo *string* com tamanho 20, e não é obrigatório.
- **episiotomia:** regista se durante o parto do bebé foi feito ou não corte vaginal à mãe. O atributo é do tipo *string* com tamanho 20, e não é obrigatório.
- **reaminacaoFetal:** regista se o bebe precisa de reanimação fetal logo após o parto ou não. O atributo é do tipo *string* com tamanho 20, e não é obrigatório.
- **indiceAPGAR:** regista o índice de avaliação neurológica (choro, abertura de olhos, ao fim de um determinado tempo). O atributo é do tipo *string* com tamanho 20, e não é obrigatório.
- **Fototerapia:** regista se o bebé recebeu luz ultravioleta ou não (quando um bebé vem amarelo). O atributo é do tipo *string* com tamanho 20, e não é obrigatório.
- **observações:** escrever observações relativas à avaliação objetivo. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.
- **IdPaciente:** é chave estrangeira, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode realizar várias avaliações objetivas). Regista os id's dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.

AvaliacaoObjetivaBebe			
	Column Name	Data Type	Allow Nulls
?	IdAvaliacaoObjetivaBebe	int	<input type="checkbox"/>
	dataRegisto	date	<input type="checkbox"/>
	Peso	numeric(5, 2)	<input type="checkbox"/>
	Altura	int	<input type="checkbox"/>
	pressaoArterial	int	<input type="checkbox"/>
	frequenciaCardiaca	int	<input checked="" type="checkbox"/>
	temperatura	numeric(5, 2)	<input checked="" type="checkbox"/>
	saturacaoOxigenio	int	<input checked="" type="checkbox"/>
	INR	int	<input checked="" type="checkbox"/>
	Perimetro	int	<input checked="" type="checkbox"/>
	IdTipoAleitamento	int	<input checked="" type="checkbox"/>
	nomeLeiteArtificial	nvarchar(50)	<input checked="" type="checkbox"/>
	IdTipoParto	int	<input checked="" type="checkbox"/>
	partoDistocico	nvarchar(20)	<input checked="" type="checkbox"/>
	epidural	nvarchar(20)	<input checked="" type="checkbox"/>
	episiotomia	nvarchar(20)	<input checked="" type="checkbox"/>
	reanimacaoFetal	nvarchar(20)	<input checked="" type="checkbox"/>
	indiceAPGAR	nvarchar(20)	<input checked="" type="checkbox"/>
	Fototerapia	nvarchar(20)	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
	IdPaciente	int	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 17 - tabela AvaliacaoObjetivoBebe**

- **Cateterismo:** esta tabela armazena informação relativa ao cateterismo (picar cateter endovenosa). A tabela tem como chave primária três atributos: IdAtitude, IdPaciente e data. Como chaves estrangeiras tem os atributos IdAtitude e IdPaciente. A tabela possui os seguintes atributos:
  - **IdAtitude:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdAtitude da tabela Atitude (a atitude ‘Cateterismo’ atitude pode ser administrada diversas vezes). Regista o id’s da atitude ‘Cateterismo’ da tabela Atitude. O atributo é do tipo inteiro e é obrigatório.


- **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode ter várias algaliações). Regista os id's dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
- **data:** é chave primária. Regista a data que o paciente colocou o cateter. O atributo é do tipo data e é obrigatório.
- **cateterismo:** regista o picar do cateter por endovenosa. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório
- **observações:** o enfermeiro regista observações relativas ao cateterismo do paciente. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

Cateterismo			
	Column Name	Data Type	Allow Nulls
🔑	IdAtitude	int	<input type="checkbox"/>
🔑	IdPaciente	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
	cateterismo	nvarchar(MAX)	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 18 - tabela Cateterismo**




- **Cirurgia:** esta tabela armazena informação relativa a cirurgias possíveis. A tabela tem como chave primária o atributo: IdCirurgia. A tabela possui os seguintes atributos:
  - **IdCirurgia:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **Nome:** regista o nome da cirurgia. O atributo é do tipo *string* com tamanho máximo, e é obrigatório.
  - **Caracterização:** regista complicações, o objetivo da cirurgia, etc. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.



Cirurgia			
	Column Name	Data Type	Allow Nulls
	IdCirurgia	int	<input type="checkbox"/>
	Nome	nvarchar(MAX)	<input type="checkbox"/>
	Caracterizacao	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 19 - tabela Cirurgia**

- **CirurgiaPaciente:** esta tabela regista as cirurgias que determinado paciente pode ter realizado. A tabela tem como chave primária três atributos: IdCirurgia, IdPaciente e data. Como chaves estrangeiras tem os atributos IdCirurgia e IdPaciente. A tabela possui os seguintes atributos:
  - **IdCirurgia:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdCirurgia da tabela Cirurgia (determinada cirurgia pode ser realizada diversas vezes). Regista o id's das cirurgias da tabela Cirurgia. O atributo é do tipo inteiro e é obrigatório.
  - **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode ter várias cirurgias). Regista os id's dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - **data:** é chave primária. Regista a data que o paciente realizou a cirurgia. O atributo é do tipo data e é obrigatório.
  - **observações:** o enfermeiro regista observações relativas às cirurgias do paciente. O atributo é do tipo string, com tamanho máximo e não é obrigatório.

CirurgiaPaciente			
	Column Name	Data Type	Allow Nulls
	IdCirurgia	int	<input type="checkbox"/>
	IdPaciente	int	<input type="checkbox"/>
	data	date	<input type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 20 – CirurgiaPaciente**

- **ColheidadeSanguePrecoce:** esta tabela regista a colheita de sangue precoce do paciente (um bebé – picada do pezinho, até aos 5 dias). A tabela tem como chave primária três atributos: IdAtitude, IdPaciente e data. Como chaves estrangeiras tem os atributos IdAtitude e IdPaciente. A tabela possui os seguintes atributos:
  - **IdAtitude:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdAtitude da tabela Atitude (a atitude ‘Colheita de Sangue Precoce’ pode ser administrada diversas vezes). Regista o id’s da atitude ‘Colheita de Sangue Precoce’ da tabela Atitude. O atributo é do tipo inteiro e é obrigatório.
  - **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode ter várias colheitas de sangue precoce). Regista os id’s dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - **data:** é chave primária. Regista a data que o enfermeiro escolhe para registo da colheita de sangue precoce. O atributo é do tipo data e é obrigatório.
  - **idadeDias:** regista a idade do paciente em dias. O atributo é do tipo inteiro e é obrigatório.
  - **observações:** o enfermeiro regista observações relativas à colheita de sangue precoce do paciente. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

ColheidadeSanguePrecoce			
	Column Name	Data Type	Allow Nulls
🔑	IdAtitude	int	<input type="checkbox"/>
🔑	IdPaciente	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
	idadeDias	int	<input type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 21 - tabela ColheidadeSanguePrecoce**

- **ColheitaUrina:** esta tabela armazena informação de quando o paciente faz a colheita normal de urina. A tabela tem como chave primária três atributos: IdAtitude, IdPaciente e data. Como chaves estrangeiras tem os atributos IdAtitude e IdPaciente. A tabela possui os seguintes atributos:
  - **IdAtitude:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdAtitude da tabela Atitude (a atitude ‘Colheita de Urina’ atitude pode ser realizada diversas vezes). Regista o id’s da atitude ‘Colheita de Urina’ da tabela Atitude. O atributo é do tipo inteiro e é obrigatório.
  - **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode realizar uma colheita de urina diversas vezes). Regista os id’s dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - **data:** é chave primária. Regista a data que o enfermeiro escolhe para registo da colheita de urina. O atributo é do tipo data e é obrigatório.
  - **exameSumario:** regista se o paciente fez a analise normal urina. O atributo é do tipo *string* com tamanho 200 e não é obrigatório.
  - **urocultura:** regista se o paciente fez a colheita urina, para exame bacteriológico. O atributo é do tipo *string* com tamanho 500 e não é obrigatório.

- **vinteQuatroHoras**: regista se o paciente fez a colheita de urina para 24h. O atributo é do tipo *string* com tamanho 50 e não é obrigatório.
- **observacoes**: o enfermeiro regista observações relativas à colheita de urina do paciente. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

ColheitaUrina			
	Column Name	Data Type	Allow Nulls
🔑	IdAtitude	int	<input type="checkbox"/>
🔑	IdPaciente	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
	exameSumario	nvarchar(200)	<input checked="" type="checkbox"/>
	urocultura	nvarchar(500)	<input checked="" type="checkbox"/>
	vinteQuatroHoras	nvarchar(50)	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 22 - tabela ColheitaUrina**

- **ColocacaoDIU**: esta tabela armazena informação relativa a colocação do DIU na mulher. A tabela tem como chave primária três atributos: IdAtitude, IdPaciente e data. Como chaves estrangeiras tem os atributos IdAtitude e IdPaciente. Esta tabela só regista os dados se o sexo do paciente for 'Feminino'. A tabela possui os seguintes atributos:
  - **IdAtitude**: é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdAtitude da tabela Atitude (a atitude 'Colocação do DIU' atitude pode ser realizada diversas vezes). Regista o id's da atitude 'Colocação do DIU' da tabela Atitude. O atributo é do tipo inteiro e é obrigatório.
  - **IdPaciente**: é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela. Regista os id's dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - **data**: é chave primária. Regista a data que o paciente coloca o DIU. O atributo é do tipo data e é obrigatório.

- **observações:** o enfermeiro registra observações relativas à colocação do DIU do paciente. O atributo é do tipo string, com tamanho máximo e não é obrigatório.

ColocacaoDIU			
	Column Name	Data Type	Allow Nulls
🔑	IdAtitude	int	<input type="checkbox"/>
🔑	IdPaciente	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
	dataColocacaoDIU	date	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 23 - tabela ColocacaoDIU**

- **Colpocitologia:** esta tabela armazena informação relativa à colheita de células do colo do útero para exames. (a tabela registra informação se o paciente for do sexo ‘Feminino’). A tabela tem como chave primária três atributos: IdAtitude, IdPaciente e data. Esta tabela só registra os dados se o sexo do paciente for ‘Feminino’. Como chaves estrangeiras tem os atributos IdAtitude e IdPaciente. A tabela possui os seguintes atributos:
  - **IdAtitude:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdAtitude da tabela Atitude (a atitude ‘Colpocitologia’ atitude pode ser realizada diversas vezes). Registra o id’s da atitude ‘Colpocitologia’ da tabela Atitude. O atributo é do tipo inteiro e é obrigatório.
  - **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente. Registra os id’s dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - **data:** é chave primária. Registra a data que o enfermeiro escolhe para registro da colpocitologia. O atributo é do tipo data e é obrigatório.
  - **dum:** registra o dia da última menstruação. O atributo é do tipo data e não é obrigatório.

- **metodoContracetivoOral:** regista qual o método contraceutivo oral que o paciente toma. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.
- **metodoContracetivoDIUData:** regista a data de colocação do DIU. O atributo é do tipo *data* e não é obrigatório.
- **metodoContracetivoImplante:** regista que implante o paciente coloca. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.
- **metodoContracetivoImplanteData:** regista a data de colocação do implante. O atributo é do tipo *data* e não é obrigatório.
- **metodoContracetivoAnelVaginalData:** regista a data de colocação do anel vaginal. O atributo é do tipo *data* e não é obrigatório.
- **metodoContracetivoPreservativos:** regista se o paciente usa ou não preservativos. O atributo é do tipo *string* com tamanho 50 e não é obrigatório.
- **metodoContracetivoIntramuscular:** regista o tipo de método contraceutivo intramuscular que o paciente usa. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.
- **metodoContracetivoInstramuscularData:** regista a data de colocação do implante intramuscular. O atributo é do tipo *data* e não é obrigatório.
- **metodoContracetivoLaqTrompasData:** regista a data de laqueação das trompas do paciente. O atributo é do tipo *data* e não é obrigatório.
- **metodoContracetivoPessarioData:** regista a data de realização do pessário (método contraceutivo colocado no útero que evita a gravidez) do paciente. O atributo é do tipo *data* e não é obrigatório.
- **observacoes:** o enfermeiro regista observações relativas às alergias do paciente. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

Colpocitologia			
	Column Name	Data Type	Allow Nulls
🔑	IdAtitude	int	<input type="checkbox"/>
🔑	IdPaciente	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
	dum	date	<input checked="" type="checkbox"/>
	metodoContraceptivoOral	nvarchar(MAX)	<input checked="" type="checkbox"/>
	metodoContraceptivoDIU...	date	<input checked="" type="checkbox"/>
	metodoContraceptivoImp...	nvarchar(MAX)	<input checked="" type="checkbox"/>
	metodoContraceptivoImp...	date	<input checked="" type="checkbox"/>
	metodoContraceptivoAne...	date	<input checked="" type="checkbox"/>
	metodoContraceptivoPres...	nvarchar(50)	<input checked="" type="checkbox"/>
	metodoContraceptivoIntr...	nvarchar(MAX)	<input checked="" type="checkbox"/>
	metodoContraceptivoInst...	date	<input checked="" type="checkbox"/>
	metodoContraceptivoLaq...	date	<input checked="" type="checkbox"/>
	metodoContraceptivoPes...	date	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 24 - tabela Colpocitologia**

- **Consulta:** esta tabela regista dados gerais da consulta do paciente. A tabela tem como chave primária o atributo: IdConsulta. Como chaves estrangeiras tem os atributos IdPaciente e IdEnfermeiro. A tabela possui os seguintes atributos:
  - **dataConsulta:** regista a data de realização da consulta. O atributo é do tipo data e é obrigatório.
  - **horaInicioConsulta:** regista a hora de início da consulta. O atributo é do tipo string com tamanho 50 e é obrigatório.
  - **historiaAtual:** regista a razão da consulta. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.
  - **sintomatologia:** regista os sintomas do paciente. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.
  - **sinais:** regista o que se observa na consulta. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.


- **escalaDor:** regista o grau de dor do paciente. O atributo é do tipo string com tamanho 50, e não é obrigatório.
- **idPaciente:** é chave estrangeira, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode ter várias consultas). Regista os id's dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
- **idEnfermeiro:** é chave estrangeira, tem uma relação de 1 para N com o atributo IdEnfermeiro da tabela Enfermeiro (um enfermeiro realiza diversas consultas). Regista os id's dos Enfermeiros da tabela Enfermeiro. O atributo é do tipo inteiro e é obrigatório.
- **valorConsulta:** regista o valor da consulta que o paciente vai pagar. O atributo é do tipo numérico, com tamanho de 6 dígitos e 2 casas decimais, e é obrigatório
- **horaFimConsulta:** regista a hora de fim da consulta. O atributo é do tipo string com tamanho 50 e é obrigatório.
- **diagnostico:** regista o diagnóstico que o enfermeiro regista do paciente. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.



Consulta			
	Column Name	Data Type	Allow Nulls
?	IdConsulta	int	<input type="checkbox"/>
	dataConsulta	date	<input type="checkbox"/>
	horalnicioConsulta	varchar(50)	<input type="checkbox"/>
	historiaAtual	nvarchar(MAX)	<input checked="" type="checkbox"/>
	sintomatologia	nvarchar(MAX)	<input checked="" type="checkbox"/>
	sinais	nvarchar(MAX)	<input checked="" type="checkbox"/>
	escalaDor	varchar(50)	<input checked="" type="checkbox"/>
	idPaciente	int	<input type="checkbox"/>
	idEnfermeiro	int	<input type="checkbox"/>
	valorConsulta	decimal(6, 2)	<input type="checkbox"/>
	horaFimConsulta	nvarchar(50)	<input type="checkbox"/>
	diagnostico	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 25 - tabela Consulta**

- **ConsultaProdutoStock:** esta tabela regista os produtos armazenados em stock que são usados durante a consulta. A tabela tem como chave primária o atributo: IdConsultaProdutoStock. Como chaves estrangeiras tem o atributo IdProdutoStock. A tabela possui os seguintes atributos:
  - **IdConsultaProdutoStock:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **IdProdutoStock:** é chave estrangeira, tem uma relação de 1 para N com o atributo IdProdutoStock da tabela ProdutoStock (podem ser usados vários produtos na mesma consulta). Regista o id do produto pretendido ProdutoStock. O atributo é do tipo inteiro e é obrigatório.
  - **quantidadeUsada:** regista a quantidade usada na consulta de casa produto. O atributo é do tipo *string* com tamanho máximo, e é obrigatório.
  - **observacoes:** o enfermeiro regista observações relativas aos produtos usados durante a consulta. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

ConsultaProdutoStock			
	Column Name	Data Type	Allow Nulls
	IdConsultaProdutoStock	int	<input type="checkbox"/>
	IdProdutoStock	int	<input type="checkbox"/>
	quantidadeUsada	int	<input type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 26 - tabela ConsultaProdutoStock**

- **Crioterapia:** esta tabela regista o tratamento pelo frio. A tabela tem como chave primária 3 atributos: IdAtitude, IdPaciente e data. Como chaves estrangeiras tem os atributos IdAtitude e IdPaciente. A tabela possui os seguintes atributos:
  - **IdAtitude:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdAtitude da tabela Atitude (a atitude ‘Crioterapia’ atitude pode ser realizada diversas vezes). Regista o id’s da atitude ‘Crioterapia’ da tabela Atitude. O atributo é do tipo inteiro e é obrigatório.
  - **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode realizar diversas crioterapias). Regista os id’s dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - **data:** é chave primária. Regista a data que o cliente realizou a crioterapia. O atributo é do tipo data e é obrigatório.
  - **localizacao:** regista a localização no corpo onde foi realizada a crioterapia. O atributo é do tipo *string* com tamanho máximo, e é obrigatório.
  - **observacoes:** o enfermeiro regista observações relativas ao procedimento realizado. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

Crioterapia			
	Column Name	Data Type	Allow Nulls
🔑	IdAtitude	int	<input type="checkbox"/>
🔑	IdPaciente	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
	localizacao	nvarchar(MAX)	<input type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 27 - tabela Crioterapia**

- **Desbridamento:** esta tabela armazena informação relativa à retirada de tecidos mortos com a lamina de *bist*. A tabela tem como chave primária três atributos: IdAtitude, IdPaciente e data. Como chaves estrangeiras tem os atributos IdAtitude e IdPaciente. A tabela possui os seguintes atributos:
  - **IdAtitude:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdAtitude da tabela Atitude (a atitude ‘Desbridamento’ atitude pode ser administrada diversas vezes). Regista o id’s da atitude ‘Desbridamento’ da tabela Atitude. O atributo é do tipo inteiro e é obrigatório.
  - **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode ter várias algaliações). Regista os id’s dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - **data:** é chave primária. Regista a data que o paciente realizou o desbridamento. O atributo é do tipo data e é obrigatório.
  - **autolico:** regista os produtos que se coloca numa compressa e vem tecidos mortos. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.
  - **Enzimático:** regista os produtos que se coloca numa compressa e vem tecidos mortos. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório
  - **cirurgico:** regista os produtos que se coloca numa compressa e vem tecidos mortos, com material cortante. O atributo é do tipo string com tamanho

máximo, e não é obrigatório O atributo é do tipo *string* com tamanho máximo, e não é obrigatório

- **observações:** o enfermeiro regista observações relativas ao procedimento realizado. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

Desbridamento			
	Column Name	Data Type	Allow Nulls
🔑	IdAtitude	int	<input type="checkbox"/>
🔑	IdPaciente	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
	autolico	nvarchar(MAX)	<input checked="" type="checkbox"/>
	enzimatico	nvarchar(MAX)	<input checked="" type="checkbox"/>
	cirurgico	nvarchar(MAX)	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 28 - tabela Desbridamento**

- **Despesa:** esta tabela armazena informação relativa as despesas da clínica. A tabela tem como chave primária o atributo: IdDespes. Como chaves estrangeiras tem os atributos idTipoDespesa e idEncomenda. A tabela possui os seguintes atributos:
  - **IdDespesa:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **data:** Regista a data em que a despesa ocorreu. O atributo é do tipo data e é obrigatório.
  - **valor:** regista o valor da despesa. O atributo é do tipo numérico, com tamanho de 6 dígitos e 2 casas decimais, e é obrigatório.
  - **observacoes:** o enfermeiro regista observações relativas ao procedimento realizado. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.
  - **idTipoDespesa:** é chave estrangeira, tem uma relação de 1 para N com o atributo IdTipoDespesa da tabela tipoDespesa (as despesas ocorrem várias


vezes). Regista o id's do tipo de despesa que ocorre. O atributo é do tipo inteiro e é obrigatório.

- **idEncomenda:** é chave estrangeira, tem uma relação de 1 para N com o atributo IdEncomenda da tabela Encomenda (a encomenda ocorre várias vezes). Regista o id's da encomenda que ocorre. O atributo é do tipo inteiro e não é obrigatório.

Despesa			
	Column Name	Data Type	Allow Nulls
🔑	IdDespesa	int	<input type="checkbox"/>
	data	date	<input type="checkbox"/>
	valor	numeric(6, 2)	<input type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
	idTipoDespesa	int	<input type="checkbox"/>
	idEncomenda	int	<input checked="" type="checkbox"/>
			<input type="checkbox"/>




**Figura 29 - tabela Despesa**

- **Doenca:** esta tabela armazena informação relativa a doenças possíveis. A tabela tem como chave primária o atributo: IdDoenca. A tabela possui os seguintes atributos:
  - **IdDoenca:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **Nome:** regista o nome da doença. O atributo é do tipo *string* com tamanho máximo, e é obrigatório.
  - **Sintomas:** regista os sintomas que a doença pode apresentar. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.

Doença			
	Column Name	Data Type	Allow Nulls
	IdDoenca	int	<input type="checkbox"/>
	Nome	nvarchar(MAX)	<input type="checkbox"/>
	Sintomas	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>


**Figura 30 - tabela Doenca**

- **DoencaPaciente:** esta tabela armazena informação relativa às doenças que o paciente pode ter. A tabela tem como chave primária três atributos: IdDoenca, IdPaciente e data. Como chaves estrangeiras tem os atributos IdDoenca e IdPaciente. A tabela possui os seguintes atributos:
  - **IdDoenca:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdDoenca da tabela Doenca (a doença pode ocorrer diversas vezes no paciente). Regista o id's da doença, da tabela Doença. O atributo é do tipo inteiro e é obrigatório.
  - **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode ter várias algaliações). Regista os id's dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - **data:** é chave primária. Regista a data que o paciente teve a doença. O atributo é do tipo data e é obrigatório.
  - **observacoes:** o enfermeiro regista observações relativas à doença do paciente. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

DoencaPaciente			
	Column Name	Data Type	Allow Nulls
	IdDoenca	int	<input type="checkbox"/>
	IdPaciente	int	<input type="checkbox"/>
	data	date	<input type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 31 - tabela DoencaPaciente**

- **DopplerFetal:** esta tabela armazena informação relativa ao Doppler Fetal (sonda para ouvir o coração do feto). A tabela tem como chave primária o atributo: IdDopplerFetal. Como chaves estrangeiras tem o atributo IdPaciente. A tabela possui os seguintes atributos:
  - **idDopplerFetal:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **IdPaciente:** é chave estrangeira, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente. Regista os id's dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - **ig:** regista a idade gestacional (semanas de gravidez). O atributo é do tipo inteiro e não é obrigatório.
  - **dppData:** Regista a data provável do parto. O atributo é do tipo data, e não é obrigatório.
  - **dppcData:** Regista a data provável do parto corrigida. O atributo é do tipo data, e não é obrigatório.
  - **primeiraEcografia:** Regista a data da primeira ecografia ao feto. O atributo é do tipo data, e não é obrigatório.
  - **escalaDor:** regista o grau de dor do paciente. O atributo é do tipo *string* com tamanho 50, e não é obrigatório.
  - **observacoes:** o enfermeiro regista observações relativas à doença do paciente. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

DopplerFetal			
	Column Name	Data Type	Allow Nulls
	IdDopplerFetal	int	<input type="checkbox"/>
	IdPaciente	int	<input type="checkbox"/>
	dataRegisto	date	<input type="checkbox"/>
	ig	int	<input checked="" type="checkbox"/>
	dppData	date	<input checked="" type="checkbox"/>
	dppcData	date	<input checked="" type="checkbox"/>
	primeiraEcografia	date	<input checked="" type="checkbox"/>
	escalaDor	nvarchar(50)	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 32 - tabela DopplerFetal**

- **DrenagemLocas:** esta tabela armazena informação relativa à drenagem das locas (retirar fluidos por aspiração, pressão negativa, gravidade). A tabela tem como chave primária três atributos: IdAtitude, IdPaciente e data. Como chaves estrangeiras tem os atributos IdAtitude e IdPaciente. A tabela possui os seguintes atributos:
  - **IdAtitude:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdAtitude da tabela Atitude (a atitude ‘Drenagem das Locas’ atitude pode ser administrada diversas vezes). Regista o id’s da atitude ‘Drenagem das Locas’ da tabela Atitude. O atributo é do tipo inteiro e é obrigatório.
  - **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode realizar várias drenagens das locas). Regista os id’s dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - **data:** é chave primária. Regista a data que o enfermeiro escolhe para registo da drenagem das locas. O atributo é do tipo data e é obrigatório.




- **drenagemLocas:** regista a informação sobre o que foi feito durante o procedimento. O atributo é do tipo *string* com tamanho máximo, e é obrigatório.
- **observacoes:** o enfermeiro regista observações relativas ao procedimento realizado. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

DrenagemLocas			
	Column Name	Data Type	Allow Nulls
🔑	IdAtitude	int	<input type="checkbox"/>
🔑	IdPaciente	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
	drenagemLocas	nvarchar(MAX)	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 33 - tabela DrenagemLocas**

- **Encomenda:** esta tabela guarda o registo das encomendas de produtos feitas para a clínica. Como chave primária tem o atributo: IdEncomenda e como chave estrangeira tem o atributo idFornecedor. A tabela tem os seguintes atributos:
  - **IdEncomenda:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **idFornecedor:** é chave estrangeira, tem uma relação de 1 para N com o atributo IdFornecedor da tabela Fornecedor. Regista o id dos Fornecedores da tabela Fornecedor. O atributo é do tipo inteiro e é obrigatório.
  - **NFatura:** Regista o número da fatura. O atributo é do tipo *string* com tamanho 100, e não é obrigatório.
  - **dataRegistoEncomenda:** regista a data em que a encomenda foi feita. O atributo é do tipo data e é obrigatório.
  - **dataEntregaPrevista:** regista a data em que a encomenda será previsivelmente entregue. O atributo é do tipo data e não é obrigatório.
  - **dataEntregaReal:** regista a data em que a encomenda foi realmente entregue. O atributo é do tipo data e não é obrigatório.

- **pago:** regista se a encomenda foi para ou não. Inicialmente, quando a encomenda é feita o campo é guardado a FALSE. O atributo é do tipo *boolean* e não é obrigatório.

Encomenda			
	Column Name	Data Type	Allow Nulls
	IdEncomenda	int	<input type="checkbox"/>
	idFornecedor	int	<input type="checkbox"/>
	Nfatura	nvarchar(100)	<input type="checkbox"/>
	dataRegistoEncomenda	date	<input type="checkbox"/>
	dataEntregaPrevista	date	<input checked="" type="checkbox"/>
	dataEntregaReal	date	<input checked="" type="checkbox"/>
	pago	bit	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 34 - tabela Encomenda**

- **Enfermeiro:** esta tabela regista os dados dos enfermeiros que trabalham na clínica. A tabela tem como chave primária o atributo: IdEnfermeiro. A tabela possui os seguintes atributos:
  - **IdEnfermeiro:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **nome:** regista o nome do enfermeiro. O atributo é do tipo *string* com tamanho 50, e é obrigatório.
  - **funcao:** regista a função do enfermeiro na clínica. O atributo é do tipo *string* com tamanho 50, e não é obrigatório.
  - **contacto:** regista o contacto telefónico do enfermeiro. O atributo é do tipo numérico com tamanho 9 e 0 casas decimais, e não é obrigatório.
  - **dataNascimento:** regista a data de nascimento do enfermeiro. O atributo é do tipo data e não é obrigatório.
  - **username:** regista o *username* do enfermeiro. Quando o enfermeiro é criado o Administrador define um *username*. Quando o enfermeiro inicia sessão pela primeira vez é obrigado a mudar a palavra passe. O atributo é do tipo *string* com tamanho 15, e é obrigatório.

- **password:** regista a password do enfermeiro. Quando o enfermeiro é registado tem uma palavra passe *default*: 'User1234\*', quando o enfermeiro inicia sessão é obrigado a mudar a palavra passe e não pode ser a palavra passe por *default*. O atributo é do tipo *string* com tamanho 15, e é obrigatório.
- **email:** regista o email do enfermeiro para recuperação do *username* e da palavra passe. O atributo é do tipo *string* com tamanho 50, e é obrigatório.
- **permissoao:** regista o tipo de permissão que o utilizador tem para aceder aos dados. O atributo é do tipo inteiro e é obrigatório.
- **passwordDefault:** este atributo regista se a *password default* está a *TRUE* ou *FALSE*. Quando um enfermeiro é registado a *password* está a *password default* e este atributo está a *TRUE*. Quando o enfermeiro inicia sessão pela primeira vez é obrigado a mudar a palavra passe e o campo passa ao valor *FALSE*. O atributo é do tipo *boolean* e é obrigatório.

Enfermeiro			
	Column Name	Data Type	Allow Nulls
🔑	IdEnfermeiro	int	<input type="checkbox"/>
	nome	varchar(50)	<input type="checkbox"/>
	funcao	nvarchar(50)	<input checked="" type="checkbox"/>
	contacto	numeric(9, 0)	<input checked="" type="checkbox"/>
	dataNascimento	date	<input checked="" type="checkbox"/>
	username	nvarchar(15)	<input type="checkbox"/>
	password	nvarchar(50)	<input type="checkbox"/>
	email	nvarchar(50)	<input type="checkbox"/>
	permissoao	int	<input type="checkbox"/>
	passwordDefault	bit	<input type="checkbox"/>
			<input type="checkbox"/>

**Figura 35 - tabela Enfermeiro**

- **ENG:** esta tabela regista os dados do ENG (entubação nasso gástrica, alimentação pela sonda, ou drenagem do conteúdo do estomago). A tabela tem como chave primária três atributos: IdAtitude, IdPaciente e data. Como chaves estrangeiras tem os atributos IdAtitude e IdPaciente. A tabela possui os seguintes atributos:

- **IdAtitude:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdAtitude da tabela Atitude (a atitude ‘ENG’ pode ser realizada diversas vezes). Regista o id’s da atitude ‘ENG’ da tabela Atitude. O atributo é do tipo inteiro e é obrigatório.
- **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode realizar várias vezes o ENG). Regista os id’s dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
- **data:** é chave primária. Regista a data em que o ENG está a ser registado no sistema. O atributo é do tipo data e é obrigatório.
- **numeroENG:** regista o calibre do diâmetro da sonda. O atributo é do tipo inteiro e não é obrigatório.
- **dataENG:** regista a data em que o ENG foi realizado. O atributo é do tipo data e não é obrigatório.
- **observacoes:** o enfermeiro regista observações relativas ao ENG do paciente. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.




ENG			
	Column Name	Data Type	Allow Nulls
🔑	IdAtitude	int	<input type="checkbox"/>
🔑	IdPaciente	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
	numeroENG	int	<input checked="" type="checkbox"/>
	dataENG	date	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 36 - tabela ENG**

- **Espirometria:** esta tabela armazena informação relativa à espirometria (medir a quantidade e o fluxo de ar que entra e sai dos pulmões, medir a capacidade respiratória). A tabela tem como chave primária três atributos: IdAtitude,

IdPaciente e data. Como chaves estrangeiras tem os atributos IdAtitude e IdPaciente. A tabela possui os seguintes atributos:

- **IdAtitude:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdAtitude da tabela Atitude (a atitude ‘Espirometria’ pode ser administrada diversas vezes). Regista o id’s da atitude ‘Espirometria’ da tabela Atitude. O atributo é do tipo inteiro e é obrigatório.
- **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode realizar várias espirometrias). Regista os id’s dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
- **data:** é chave primária. Guarda a data de realização da espirometria. O atributo é do tipo data e é obrigatório.
- **fev:** regista a força expiratória voluntaria. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório
- **fvc:** regista a força voluntaria cardíaca. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório
- **fr:** regista a frequência cardíaca. O atributo é do tipo inteiro, e não é obrigatório.
- **caracteristicaSuperficial:** Regista se a respiração é do tipo superficial. O atributo é do tipo *string* com tamanho 50, e não é obrigatório.
- **caracteristicaProfunda:** Regista se a respiração é do tipo profunda. O atributo é do tipo *string* com tamanho 50, e não é obrigatório.
- **caracteristicaAdbominal:** Regista se a respiração é do tipo abdominal. O atributo é do tipo *string* com tamanho 50, e não é obrigatório.
- **caracteristicaToracica:** Regista se a respiração é do tipo torácica. O atributo é do tipo *string* com tamanho 50, e não é obrigatório.
- **caracteristicaMista:** Regista se a respiração é do tipo mista. O atributo é do tipo *string* com tamanho 50, e não é obrigatório.
- **escalaDor:** regista o grau de dor do paciente. O atributo é do tipo *string* com tamanho 50, e não é obrigatório.
- **observacoes:** o enfermeiro regista observações relativas ao procedimento realizado. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

Espirometria			
	Column Name	Data Type	Allow Nulls
	IdEspirometria	int	<input type="checkbox"/>
	IdPaciente	int	<input type="checkbox"/>
	data	date	<input type="checkbox"/>
	fev	nvarchar(MAX)	<input checked="" type="checkbox"/>
	fvc	nvarchar(MAX)	<input checked="" type="checkbox"/>
	fr	int	<input checked="" type="checkbox"/>
	caracteristicaSuperficial	nvarchar(50)	<input checked="" type="checkbox"/>
	caracteristicaProfunda	nvarchar(50)	<input checked="" type="checkbox"/>
	caracteristicaAdbominal	nvarchar(50)	<input checked="" type="checkbox"/>
	caracteristicaToracica	nvarchar(50)	<input checked="" type="checkbox"/>
	caracteristicaMista	nvarchar(50)	<input checked="" type="checkbox"/>
	escalaDor	nvarchar(50)	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 37 - tabela Espirometria**

- **Exame:** esta tabela armazena informação relativa aos exames que o paciente realiza. A tabela tem como chave primária três atributos: IdPaciente, idTipoExame e data. Como chaves estrangeiras tem os atributos IdAtitude e IdPaciente. A tabela possui os seguintes atributos:
  - **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode realizar vários exames). Regista os id's dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - **idTipoExame:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdTipoExame da tabela tipoExame. Regista os id's dos exames da tabela tipoExame. O atributo é do tipo inteiro e é obrigatório.
  - **data:** é chave primária. Regista a data que o paciente realiza o exame. O atributo é do tipo data e é obrigatório.

- **designacao:** regista a designação relativa ao exame que o paciente realizou. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.
- **observacoes:** regista observações relativas ao exame realizado. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.

Exame			
	Column Name	Data Type	Allow Nulls
🔑	idPaciente	int	<input type="checkbox"/>
🔑	idTipoExame	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
	designacao	nvarchar(MAX)	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 38 - tabela Exame**

- **Flebografia:** esta tabela armazena informação relativa à flebografia (exame ecográfico ao sistema venoso e arterial - sistema circulatório). A tabela tem como chave primária três atributos: IdAtitude, IdPaciente e data. Como chaves estrangeiras tem os atributos IdAtitude e IdPaciente. A tabela possui os seguintes atributos:
  - **IdAtitude:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdAtitude da tabela Atitude (a atitude ‘Flebografia’ pode ser administrada diversas vezes). Regista o id’s da atitude ‘Flebografia’ da tabela Atitude. O atributo é do tipo inteiro e é obrigatório.
  - **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode realizar diversas vezes o exame). Regista os id’s dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - **data:** é chave primária. Regista a data que o enfermeiro escolhe para registo ou da algaliação. O atributo é do tipo data e é obrigatório.

- **flebografia:** regista detalhes relativas ao exame realizado. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.
- **observacoes:** o enfermeiro regista observações relativas ao procedimento realizado. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

Flebografia			
	Column Name	Data Type	Allow Nulls
🔑	IdAtitude	int	<input type="checkbox"/>
🔑	IdPaciente	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
	flebografia	nvarchar(MAX)	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 39 - tabela Flebografia**


- **Fornecedor:** esta tabela regista informação relativa aos fornecedores de produtos para a clínica. A tabela tem como chave primária o atributo: IdFornecedor.
  - **IdFornecedor:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **nif:** guarda o NIF do fornecedor. O atributo é do tipo inteiro e é obrigatório.
  - **nome:** regista o nome do fornecedor. O atributo é do tipo *string* com tamanho 50, e é obrigatório.
  - **contacto:** regista o contacto telefónico do fornecedor. O atributo é do tipo inteiro e é obrigatório.
  - **email:** regista o email do fornecedor. O atributo é do tipo *string* com tamanho 500 e não é obrigatório.
  - **observacoes:** registar observações relativas ao fornecedor a ser registado. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.
  - **rua:** registo da rua da morada do fornecedor. O atributo é do tipo *string* com tamanho máximo e é obrigatório.
  - **numeroMorada:** registo o número da morada do fornecedor. O atributo é do tipo inteiro e não é obrigatório.



- **andarPiso:** regista o andar e piso da morada do fornecedor (ex: 1º andar, Esquerdo). O atributo é do tipo *string* com tamanho 50 e não é obrigatório.
- **localidade:** regista a localidade da morada do fornecedor. O atributo é do tipo *string* com tamanho 100 e é obrigatório.
- **bairroLocal:** bairro ou Local da morada. (nem todas as moradas têm esta parte, ex: Bairro da Pedra). O atributo é do tipo *string* com tamanho 100 e não é obrigatório.
- **codPostalPrefixo:** regista o prefixo do código postal, composto por 4 algarismos. O atributo é do tipo *string* com tamanho 4 e é obrigatório.
- **codPostalSufixo:** regista o sufixo do código postal, composto por 3 algarismos. O atributo é do tipo *string* com tamanho 3 e é obrigatório.
- **designacao:** regista a designação relativa à morada do fornecedor (ex: LRA (Leiria)). O atributo é do tipo *string* com tamanho 150 e não é obrigatório.

Nesta tabela para registar a morada correta foi consultado o manual de endereçamento de Portugal dos CTT, disponível no seguinte link:

[https://www.ctt.pt/application/themes/pdfs/empresas/manual\\_enderecamento\\_abr\\_2018.pdf](https://www.ctt.pt/application/themes/pdfs/empresas/manual_enderecamento_abr_2018.pdf)

Fornecedor			
	Column Name	Data Type	Allow Nulls
	IdFornecedor	int	<input type="checkbox"/>
	nif	int	<input type="checkbox"/>
	nome	nvarchar(50)	<input type="checkbox"/>
	contacto	int	<input type="checkbox"/>
	email	nvarchar(500)	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
	rua	nvarchar(MAX)	<input type="checkbox"/>
	numeroMorada	int	<input checked="" type="checkbox"/>
	andarPiso	nvarchar(50)	<input checked="" type="checkbox"/>
	localidade	nvarchar(100)	<input type="checkbox"/>
	bairroLocal	nvarchar(100)	<input checked="" type="checkbox"/>
	codPostalPrefixo	nvarchar(4)	<input type="checkbox"/>
	codPostalSufixo	nvarchar(3)	<input type="checkbox"/>
	designacao	nvarchar(150)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 40 - tabela Fornecedor**

- **ImplanteContracetivo:** esta tabela regista informação relativa ao implante contracetivo (Esta tabela só regista os dados se o sexo do paciente for ‘Feminino’). A tabela tem como chave primária três atributos: IdAtitude, IdPaciente e data. Como chaves estrangeiras tem os atributos IdAtitude e IdPaciente. A tabela possui os seguintes atributos:
  - **IdAtitude:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdAtitude da tabela Atitude (a atitude pode ser administrada diversas vezes). Regista o id’s da atitude ‘Implante Contracetivo’ da tabela Atitude. O atributo é do tipo inteiro e é obrigatório.
  - **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode ter vários implantes contracetivos ao longo da vida). Regista os id’s dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.

- **data:** é chave primária. Regista a data de registo dos dados na tabela durante a consulta. O atributo é do tipo data e é obrigatório.
- **dataColocacao:** é chave primária. Regista a data de colocação do implante contraceptivo. O atributo é do tipo data e não é obrigatório.
- **dataRetirada:** é chave primária. Regista a data de retirada do implante contraceptivo. O atributo é do tipo data e não é obrigatório.
- **observacoes:** o enfermeiro regista observações relativas ao procedimento realizado. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

ImplanteContraceptivo			
	Column Name	Data Type	Allow Nulls
🔑	IdAtitude	int	<input type="checkbox"/>
🔑	IdPaciente	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
	dataColocacao	date	<input checked="" type="checkbox"/>
	dataRetirada	date	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 41 - tabela ImplanteContraceptivo**

- **Inalacoes:** esta tabela armazena informação relativa às inalações (inspiração de um produto). A tabela tem como chave primária três atributos: IdAtitude, IdPaciente e data. Como chaves estrangeiras tem os atributos IdAtitude e IdPaciente. A tabela possui os seguintes atributos:
  - **IdAtitude:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdAtitude da tabela Atitude (a atitude ‘Inalações’ pode ser administrada diversas vezes). Regista o id’s da atitude ‘Inalações’ da tabela Atitude. O atributo é do tipo inteiro e é obrigatório.
  - **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode ter várias inalações). Regista os id’s dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.

- **data:** é chave primária. Regista a data que o enfermeiro escolhe para registo ou das inalações. O atributo é do tipo data e é obrigatório.
- **O2:** regista o produto que foi inspirado por O2 (oxigénio – ar). O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.
- **aerossol:** regista o produto que foi inspirado por aerossóis. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.
- **inaladores:** regista o produto que foi inspirado por inaladores. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.
- **observacoes:** o enfermeiro regista observações relativas às alergias do paciente. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

Inalacoes			
	Column Name	Data Type	Allow Nulls
🔑	IdAtitude	int	<input type="checkbox"/>
🔑	IdPaciente	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
	O2	nvarchar(MAX)	<input checked="" type="checkbox"/>
	aerossol	nvarchar(MAX)	<input checked="" type="checkbox"/>
	inaladores	nvarchar(MAX)	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 42 - tabela Inalacoes**

- **LavagemAuricular:** regista a lavagem auricular do paciente (lavagem dos ouvidos). A tabela tem como chave primária três atributos: IdAtitude, IdPaciente e data. Como chaves estrangeiras tem os atributos IdAtitude e IdPaciente. A tabela possui os seguintes atributos:
  - **IdAtitude:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdAtitude da tabela Atitude (a atitude ‘Lavagem Auricular’ pode ser realizada diversas vezes). Regista o id’s da atitude ‘Lavagem Auricular’ da tabela Atitude. O atributo é do tipo inteiro e é obrigatório.

- **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente. Regista os id's dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
- **data:** é chave primária. Regista a data que o paciente realizou a lavagem auricular. O atributo é do tipo data e é obrigatório.
- **ouvidoDireito:** regista se a lavagem é no ouvido direito. O atributo é do tipo *string* com tamanho 50, e não é obrigatório.
- **ouvidoEsquerdo:** regista se a lavagem é no ouvido esquerdo. O atributo é do tipo *string* com tamanho 50, e não é obrigatório.
- **ambos:** registar se a lavagem é em ambos os ouvidos. O atributo é do tipo *string* com tamanho 50, e não é obrigatório.
- **observacoes:** o enfermeiro regista observações relativa a lavagem auricular do paciente. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

LavagemAuricular			
	Column Name	Data Type	Allow Nulls
🔑	IdAtitude	int	<input type="checkbox"/>
🔑	IdPaciente	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
	ouvidoDireito	nvarchar(50)	<input checked="" type="checkbox"/>
	ouvidoEsquerdo	nvarchar(50)	<input checked="" type="checkbox"/>
	ambos	nvarchar(50)	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 43 - tabela LavagemAuricular**

- **LavagemOcular:** regista a lavagem ocular do paciente (lavagem dos olhos com soro). A tabela tem como chave primária três atributos: IdAtitude, IdPaciente e data. Como chaves estrangeiras tem os atributos IdAtitude e IdPaciente. A tabela possui os seguintes atributos:
  - **IdAtitude:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdAtitude da tabela Atitude (a atitude 'Lavagem Ocular' pode ser realizada diversas vezes). Regista o id's da

atitude 'Lavagem Ocular' da tabela Atitude. O atributo é do tipo inteiro e é obrigatório.

- **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente. Regista os id's dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
- **data:** é chave primária. Regista a data em que o paciente realizou a lavagem ocular. O atributo é do tipo data e é obrigatório.
- **olhoDireito:** regista se a lavagem é no olho direito. O atributo é do tipo *string* com tamanho 50, e não é obrigatório.
- **olhoEsquerdo:** regista se a lavagem é no olho esquerdo. O atributo é do tipo *string* com tamanho 50, e não é obrigatório.
- **ambos:** registar se a lavagem é em ambos os olhos. O atributo é do tipo *string* com tamanho 50, e não é obrigatório.
- **observacoes:** o enfermeiro regista observações relativa a lavagem ocular do paciente. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.




LavagemOcular			
	Column Name	Data Type	Allow Nulls
🔑	IdAtitude	int	<input type="checkbox"/>
🔑	IdPaciente	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
	olhoDireito	nvarchar(50)	<input checked="" type="checkbox"/>
	olhoEsquerdo	nvarchar(50)	<input checked="" type="checkbox"/>
	ambos	nvarchar(50)	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 44 - tabela LavagemOcular**

- **LavagemOcular:** regista a lavagem ocular do paciente (lavagem da vesícula). A tabela tem como chave primária três atributos: IdAtitude, IdPaciente e data. Como

chaves estrangeiras tem os atributos **IdAtitude** e **IdPaciente**. A tabela possui os seguintes atributos:

- **IdAtitude**: é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo **IdAtitude** da tabela **Atitude** (a atitude ‘Lavagem Ocular’ pode ser realizada diversas vezes). Regista o id’s da atitude ‘Lavagem Ocular’ da tabela **Atitude**. O atributo é do tipo inteiro e é obrigatório.
- **IdPaciente**: é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo **IdPaciente** da tabela **Paciente**. Regista os id’s dos Pacientes da tabela **Paciente**. O atributo é do tipo inteiro e é obrigatório.
- **data**: é chave primária. Regista a data em que o paciente realizou a lavagem vesical. O atributo é do tipo **date** e é obrigatório.
- **lavagemVesical**: regista especificações da lavagem vesical. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.
- **observacoes**: o enfermeiro regista observações relativa a lavagem vesical do paciente. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

LavagemVesical			
	Column Name	Data Type	Allow Nulls
	IdAtitude	int	<input type="checkbox"/>
	IdPaciente	int	<input type="checkbox"/>
	data	date	<input type="checkbox"/>
	lavagemVesical	nvarchar(MAX)	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 45 - tabela LavagemVesical**

- **LinhaEncomenda**: esta tabela regista informação sobre os produtos que se pretende encomendar. A tabela tem como chave primária o atributo: **IdLinhaEncomenda**. Como chaves estrangeiras tem os atributos **idProdutoStock** e **IdEncomenda**. A tabela possui os seguintes atributos:

- **IdLinhaEncomenda:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
- **quantidade:** guarda a quantidade que se quer encomendar dos produtos. O atributo é do tipo inteiro e é obrigatório.
- **idProdutoStock:** este atributo é chave estrangeira, tem uma relação de 1 para N com o atributo IdProdutoStock da tabela ProdutoStock (um produto é encomendado várias vezes) Regista os id's dos produtos da tabela ProdutoStock. O atributo é do tipo inteiro e é obrigatório.
- **idEncomenda:** este atributo é chave estrangeira, tem uma relação de 1 para N com o atributo idEncomenda da tabela Encomenda. Regista os id's das encomendas da tabela Encomenda. O atributo é do tipo inteiro e é obrigatório.

LinhaEncomenda			
	Column Name	Data Type	Allow Nulls
🔑	IdLinhaEncomenda	int	<input type="checkbox"/>
	quantidade	int	<input type="checkbox"/>
	idProdutoStock	int	<input type="checkbox"/>
	idEncomenda	int	<input type="checkbox"/>
			<input type="checkbox"/>

**Figura 46 - tabela LinhaEncomenda**

- **LocalizacaoDor:** esta tabela regista a dor nas mãos e pés. A tabela tem como chave primária os atributos: IdTratamentoMaosPes, IdPaciente e data. Como chaves estrangeiras tem os atributos IdTratamentoMaosPes e IdPaciente. A tabela possui os seguintes atributos:
  - **IdTratamentoMaosPes:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdTratamentoMaosPes da tabela TratamentoMaosPes. Regista os id's do tratamento da tabela TratamentoMaosPes. O atributo é do tipo inteiro e é obrigatório



- **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente. Regista os id's dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
- **data:** é chave primária. Regista a data em que a dor foi detetada na consulta. O atributo é do tipo data e é obrigatório.
- **localizacao:** regista a localização da dor do paciente. O atributo é do tipo *string* com tamanho máximo, e é obrigatório.
- **observacoes:** o enfermeiro regista observações relativas às queixas do paciente. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

LocalizacaoDor			
	Column Name	Data Type	Allow Nulls
🔑	IdTratamentoMaosPes	int	<input type="checkbox"/>
🔑	IdPaciente	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
	localizacao	nvarchar(MAX)	<input type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 47 - tabela LocalizacaoDor**

- **LocalizacaoDorConsulta:** esta tabela regista a localização da dor do paciente durante a consulta. A tabela tem como chave primária os atributos: IdLocalizacaoDor, IdPaciente e data. Como chave estrangeira tem o atributo IdPaciente. A tabela possui os seguintes atributos:
  - **IdLocalizacaoDor:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente. Regista os id's dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.




- **data:** é chave primária. Regista a data em que a dor foi detetada na consulta. O atributo é do tipo data e é obrigatório.
- **localizacao:** regista a localização da dor do paciente. O atributo é do tipo *string* com tamanho máximo, e é obrigatório.
- **observacoes:** o enfermeiro regista observações relativas às queixas de dor do paciente. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

LocalizacaoDorConsulta			
	Column Name	Data Type	Allow Nulls
🔑	IdLocalizacaoDor	int	<input type="checkbox"/>
🔑	IdPaciente	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
	localizacao	nvarchar(MAX)	<input type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 48 - tabela LocalizacaoDorConsulta**




- **LocalizacaoDorDopplerArterialVenoso:** esta tabela regista em que local foi usado a sonda para ouvir as veias a contrair. A tabela tem como chave primária os atributos: IdTratamento, IdPaciente e data. Como chave estrangeira tem o atributo IdPaciente. A tabela possui os seguintes atributos:
  - **IdTratamento:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente. Regista os id's dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - **data:** é chave primária. Regista a data em que a dor foi detetada na consulta. O atributo é do tipo data e é obrigatório.
  - **localizacao:** regista a localização onde foi usada a sonda. O atributo é do tipo *string* com tamanho máximo, e é obrigatório.

- **observacoes:** o enfermeiro regista observações relativas às queixas de dor do paciente. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

LocalizacaoDorDopplerArterialVenoso			
	Column Name	Data Type	Allow Nulls
	IdTratamento	int	<input type="checkbox"/>
	IdPaciente	int	<input type="checkbox"/>
	data	date	<input type="checkbox"/>
	localizacao	nvarchar(MAX)	<input type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 49 - tabela LocalizacaoDorDopplerArterialVenoso**

- **LocalizacaoDorDopplerFetal:** esta tabela regista onde foi feita o exame. A tabela tem como chave primária os atributos: IdTratamento, IdPaciente e data. Como chave estrangeira tem o atributo IdPaciente. A tabela possui os seguintes atributos:
  - **IdTratamento:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente. Regista os id's dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - **data:** é chave primária. Regista a data em que a dor foi detetada na consulta. O atributo é do tipo data e é obrigatório.
  - **localizacao:** regista a localização do exame e eventuais dores. O atributo é do tipo *string* com tamanho máximo, e é obrigatório.
  - **observacoes:** o enfermeiro regista observações relativas às queixas de dor do paciente. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

LocalizacaoDorDopplerFetal			
	Column Name	Data Type	Allow Nulls
	IdTratamento	int	<input type="checkbox"/>
	IdPaciente	int	<input type="checkbox"/>
	data	date	<input type="checkbox"/>
	localizacao	nvarchar(MAX)	<input type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 50 - tabela LocalizacaoDorDopplerFetal**

- LocalizacaoDorEspisiotomia** esta tabela regista a localização da dor para realizar a prova de função pulmonar. A tabela tem como chave primária os atributos: IdTratamento, IdPaciente e data. Como chave estrangeira tem o atributo IdPaciente. A tabela possui os seguintes atributos:
  - IdTratamento:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente. Regista os id's dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - data:** é chave primária. Regista a data em que a dor foi detetada na consulta. O atributo é do tipo data e é obrigatório.
  - localizacao:** regista a localização da dor do paciente. O atributo é do tipo *string* com tamanho máximo, e é obrigatório.
  - observacoes:** o enfermeiro regista observações relativas às queixas de dor do paciente. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

LocalizacaoDorEpisiotomia			
	Column Name	Data Type	Allow Nulls
🔑	IdTratamento	int	<input type="checkbox"/>
🔑	IdPaciente	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
	localizacao	nvarchar(MAX)	<input type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>




**Figura 51 - tabela LocalizacaoDorEspisiotomia**

- **LocalizacaoDorTratamento:** esta tabela regista a localização da dor do paciente durante os tratamentos. A tabela tem como chave primária os atributos: IdLocalizacaoDorTratamento, IdPaciente e data. Como chave estrangeira tem o atributo IdPaciente. A tabela possui os seguintes atributos:
  - **IdLocalizacaoDorTratamento:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente. Regista os id's dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - **data:** é chave primária. Regista a data em que a dor foi detetada na consulta. O atributo é do tipo data e é obrigatório.
  - **localizacao:** regista a localização da dor do paciente. O atributo é do tipo string com tamanho máximo, e é obrigatório.
  - **observacoes:** o enfermeiro regista observações relativas às queixas de dor do paciente. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

LocalizacaoDorTratamento			
	Column Name	Data Type	Allow Nulls
🔑	IdLocalizacaoDorTratamento	int	<input type="checkbox"/>
🔑	IdPaciente	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
	localizacao	nvarchar(MAX)	<input type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 52 – tabela LocalizacaoDorTratamento**

- **LocalizacaoDorVacinacao:** esta tabela regista a localização da dor do paciente durante a consulta. A tabela tem como chave primária os atributos: IdTratamento, IdPaciente e data. Como chave estrangeira tem o atributo IdPaciente. A tabela possui os seguintes atributos:
  - **IdTratamento:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente. Regista os id's dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - **data:** é chave primária. Regista a data em que a dor foi detetada na consulta. O atributo é do tipo data e é obrigatório.
  - **localizacao:** regista a localização da dor do paciente. O atributo é do tipo *string* com tamanho máximo, e é obrigatório.
  - **observacoes:** o enfermeiro regista observações relativas às queixas de dor do paciente. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.


LocalizacaoDorVacinacao			
	Column Name	Data Type	Allow Nulls
	IdTratamento	int	<input type="checkbox"/>
	IdPaciente	int	<input type="checkbox"/>
	data	date	<input type="checkbox"/>
	localizacao	nvarchar(MAX)	<input type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 53 - tabela LocalizacaoDorVacinacao**

- **Medicacao:** esta tabela regista a medicação do paciente. A tabela tem como chave primária o atributo: IdMedicacao. Como chave estrangeira tem o atributo IdPaciente. A tabela possui os seguintes atributos:
  - **IdMedicacao:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **medicamentos:** regista os medicamentos que o paciente deve tomar. O atributo é do tipo *string* com tamanho máximo, e é obrigatório.
  - **jejum:** regista se o paciente toma os medicamentos ao jejum. O atributo é do tipo *string* com tamanho 50 e não é obrigatório.
  - **pequenoAlmoco:** regista se o paciente toma os medicamentos ao pequeno almoço. O atributo é do tipo *string* com tamanho 50 e não é obrigatório.
  - **Almoco:** regista se o paciente toma os medicamentos ao almoço. O atributo é do tipo *string* com tamanho 50 e não é obrigatório.
  - **lanche:** regista se o paciente toma os medicamentos ao lanche. O atributo é do tipo *string* com tamanho 50 e não é obrigatório.
  - **jantar:** regista se o paciente toma os medicamentos ao jantar. O atributo é do tipo *string* com tamanho 50 e não é obrigatório.
  - **deitar:** regista se o paciente toma os medicamentos ao deitar. O atributo é do tipo *string* com tamanho 50 e não é obrigatório.
  - **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente. Regista os id's dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.


- **data:** é chave primária. Regista a data em que a prescrição dos medicamentos é passada. O atributo é do tipo *data* e é obrigatório.
- **quantidadeJejum:** regista a quantidade de medicamentos que o paciente tem de tomar ao jejum. O atributo é do tipo *string* com tamanho 500 e não é obrigatório.
- **quantidadePequenoAlmoco:** regista a quantidade de medicamentos que o paciente tem de tomar ao pequeno almoço. O atributo é do tipo *string* com tamanho 500 e não é obrigatório.
- **quantidadeAlmoco:** regista a quantidade de medicamentos que o paciente tem de tomar ao almoço. O atributo é do tipo *string* com tamanho 500 e não é obrigatório.
- **quantidadeLanche:** regista a quantidade de medicamentos que o paciente tem de tomar ao lanche. O atributo é do tipo *string* com tamanho 500 e não é obrigatório.
- **quantidadeJantar:** regista a quantidade de medicamentos que o paciente tem de tomar ao jantar. O atributo é do tipo *string* com tamanho 500 e não é obrigatório.
- **quantidadeDeitar:** regista a quantidade de medicamentos que o paciente tem de tomar ao deitar. O atributo é do tipo *string* com tamanho 500 e não é obrigatório.
- **observacoes:** o enfermeiro regista outras indicações que possam ser necessárias relativas aos medicamentos. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.



Medicacao			
	Column Name	Data Type	Allow Nulls
	IdMedicacao	int	<input type="checkbox"/>
	medicamentos	nvarchar(MAX)	<input type="checkbox"/>
	jejum	nvarchar(50)	<input checked="" type="checkbox"/>
	pequenoAlmoco	nvarchar(50)	<input checked="" type="checkbox"/>
	almoco	nvarchar(50)	<input checked="" type="checkbox"/>
	lanche	nvarchar(50)	<input checked="" type="checkbox"/>
	jantar	nvarchar(50)	<input checked="" type="checkbox"/>
	deitar	nvarchar(50)	<input checked="" type="checkbox"/>
	IdPaciente	int	<input checked="" type="checkbox"/>
	data	date	<input type="checkbox"/>
	quantidadeJejum	nvarchar(500)	<input checked="" type="checkbox"/>
	quantidadePequenoAlm...	nvarchar(500)	<input checked="" type="checkbox"/>
	quantidadeAlmoco	nvarchar(500)	<input checked="" type="checkbox"/>
	quantidadeLanche	nvarchar(500)	<input checked="" type="checkbox"/>
	quantidadeJantar	nvarchar(500)	<input checked="" type="checkbox"/>
	quantidadeDeitar	nvarchar(500)	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>




**Figura 54 - tabela Medicacao**

- **MetodoContracetivo:** esta tabela armazena informação relativa aos métodos contraceptivos existentes. A tabela tem como chave primária o atributo: IdMetodoContracetivo. A tabela possui os seguintes atributos:
  - **IdMetodoContracetivo:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **nomeMetodoContracetivo:** regista o nome dos métodos contraceptivos existentes. O atributo é do tipo *string* com tamanho máximo, e é obrigatório.
  - **observacoes:** o enfermeiro regista observações relativas aos métodos contraceptivos. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

MetodoContracetivo			
	Column Name	Data Type	Allow Nulls
	IdMetodoContracetivo	int	<input type="checkbox"/>
	nomeMetodoContracetivo	nvarchar(MAX)	<input type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 55 - tabela MetodoContracetivo**

- **MonitorizacaoECG:** esta tabela armazena informação relativa ao ECG (eletrocardiograma). A tabela tem como chave primária três atributos: IdAtitude, IdPaciente e data. Como chaves estrangeiras tem os atributos IdAtitude e IdPaciente. A tabela possui os seguintes atributos:
  - **IdAtitude:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdAtitude da tabela Atitude (a atitude ‘Monitorização ECG’ pode ser administrada diversas vezes). Regista o id’s da atitude ‘Monitorização ECG’ da tabela Atitude. O atributo é do tipo inteiro e é obrigatório.
  - **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente. Regista os id’s dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - **data:** é chave primária. Regista a data que o paciente realizou o exame. O atributo é do tipo data e é obrigatório.
  - **monitorizacaoECG:** registar dados específicos relativos ao exame. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.
  - **observacoes:** o enfermeiro regista observações relativas exame do paciente. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

MonitorizacaoECG			
	Column Name	Data Type	Allow Nulls
	IdAtitude	int	<input type="checkbox"/>
	IdPaciente	int	<input type="checkbox"/>
	data	date	<input type="checkbox"/>
	monitorizacaoECG	nvarchar(MAX)	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 56 - tabela MonitorizacaoECG**

- **Paciente:** esta tabela armazena informações dos pacientes. A tabela tem como chave primária o atributo IdPaciente. Como chaves estrangeiras tem o IdEnfermeiro e IdProfissao. (cada paciente tem o seu enfermeiro). A tabela possui os seguintes atributos:
  - **IdPaciente:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **Nome:** regista o nome do paciente. O atributo é do tipo *string* com tamanho 50, e é obrigatório
  - **DataNascimento:** regista a data de nascimento do paciente. O atributo é do tipo data e é obrigatório.
  - **Email:** regista o email do paciente. O atributo é do tipo *string* com tamanho 500, e não é obrigatório.
  - **Contacto:** regista o contacto telefónico do paciente. O atributo é do tipo inteiro e é obrigatório.
  - **Nif:** guarda o NIF do paciente. O atributo é do tipo inteiro e é obrigatório.
  - **Rua:** registo da rua da morada do paciente. O atributo é do tipo *string* com tamanho máximo e é obrigatório.
  - **NumeroCasa:** registo o número da morada do paciente. O atributo é do tipo inteiro e não é obrigatório.
  - **Andar:** regista o andar e piso da morada do paciente (ex: 1º andar, Esquerdo). O atributo é do tipo *string* com tamanho 50 e não é obrigatório.
  - **localidade:** regista a localidade da morada do paciente. O atributo é do tipo *string* com tamanho 100 e é obrigatório.

- **bairroLocal:** bairro ou Local da morada. (nem todas as moradas têm esta parte, ex: Bairro da Pedra). O atributo é do tipo *string* com tamanho 100 e não é obrigatório.
- **codPostalPrefixo:** regista o prefixo do código postal, composto por 4 algarismos. O atributo é do tipo *string* com tamanho 4 e é obrigatório.
- **codPostalSufixo:** regista o sufixo do código postal, composto por 3 algarismos. O atributo é do tipo *string* com tamanho 3 e é obrigatório.
- **designacao:** regista a designação relativa à morada do fornecedor (ex: LRA (Leiria)). O atributo é do tipo *string* com tamanho 500 e não é obrigatório.
- **IdEnfermeiro:** é chave estrangeira, tem uma relação de N para 1 com o atributo IdPaciente da tabela Paciente. Regista os id's dos Enfermeiros da tabela Enfermeiro. O atributo é do tipo inteiro e é obrigatório.
- **Acordo:** regista o tipo de acordo que o paciente quer com a clínica (se vai usar seguro, subsistema, ou é através do SNS). O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.
- **NomeSeguradora:** se o Acordo for com uma seguradora, este atributo regista o nome da seguradora. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.
- **NumeroApoliceSeguradora:** se o Acordo for com uma seguradora, este atributo regista o número da apólice do seguro. O atributo é do tipo inteiro, e não é obrigatório.
- **NomeSubsistema:** se o Acordo for um subsistema, este atributo regista o nome do subsistema. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.
- **NumeroSubsistema:** se o Acordo for com um subsistema, este atributo regista o número do subsistema. O atributo é do tipo inteiro, e não é obrigatório.
- **NumeroSNS:** se o Acordo for SNS, este atributo regista o número do SNS do paciente. O atributo é do tipo inteiro, e não é obrigatório.
- **Sexo:** este atributo regista o sexo (Feminino, Masculino, Indefinido) do paciente. O atributo é do tipo *string* com tamanho 50, e não é obrigatório.
- **PlanoVacinao:** este atributo regista se o plano de vacinação do paciente está atualizado ou não. O atributo é do tipo *string* com tamanho 50, e não é obrigatório.

- **IdProfissão:** é chave estrangeira, tem uma relação de 1 para N com o atributo IdProfissao da tabela Profissao. Regista o id da profissão que o paciente tem, da tabela Profissão. O atributo é do tipo inteiro e não é obrigatório.

Nesta tabela para registar a morada correta foi consultado o manual de endereçamento de Portugal dos CTT, disponível no seguinte link:

[https://www.ctt.pt/application/themes/pdfs/empresas/manual\\_enderecamento\\_abr\\_2018.pdf](https://www.ctt.pt/application/themes/pdfs/empresas/manual_enderecamento_abr_2018.pdf)


Paciente			
	Column Name	Data Type	Allow Nulls
	IdPaciente	int	<input type="checkbox"/>
	Nome	nvarchar(50)	<input type="checkbox"/>
	DataNascimento	date	<input type="checkbox"/>
	Email	nvarchar(500)	<input checked="" type="checkbox"/>
	Contacto	int	<input type="checkbox"/>
	Nif	int	<input type="checkbox"/>
	Rua	nvarchar(MAX)	<input type="checkbox"/>
	NumeroCasa	int	<input checked="" type="checkbox"/>
	Andar	nvarchar(1000)	<input checked="" type="checkbox"/>
	localidade	nvarchar(1000)	<input type="checkbox"/>
	bairroLocal	nvarchar(1000)	<input checked="" type="checkbox"/>
	codPostalPrefixo	nvarchar(4)	<input type="checkbox"/>
	codPostalSufixo	nvarchar(3)	<input type="checkbox"/>
	designacao	nvarchar(500)	<input checked="" type="checkbox"/>
	IdEnfermeiro	int	<input type="checkbox"/>
	Acordo	nvarchar(MAX)	<input type="checkbox"/>
	NomeSeguradora	nvarchar(MAX)	<input checked="" type="checkbox"/>
	NumeroApoliceSeguradora	int	<input checked="" type="checkbox"/>
	NomeSubsistema	nvarchar(MAX)	<input checked="" type="checkbox"/>
	NumeroSubsistema	int	<input checked="" type="checkbox"/>
	NumeroSNS	int	<input checked="" type="checkbox"/>
	Sexo	nvarchar(50)	<input checked="" type="checkbox"/>
	PlanoVacinao	nvarchar(50)	<input checked="" type="checkbox"/>
	IdProfissao	int	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Figura 57 - tabela Paciente

- **Parto:** esta tabela regista os possíveis tipos de parto. Esta tabela tem como chave primária o atributo IdParto. A tabela possui os seguintes atributos:
  - **IdParto:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **tipoParto:** regista os possíveis tipos de parto. O atributo é do tipo *string* com tamanho máximo, e é obrigatório.

- **Observacoes:** registo observações relativas ao tipo de parto. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.

Parto			
	Column Name	Data Type	Allow Nulls
🔑	IdParto	int	<input type="checkbox"/>
	tipoParto	nvarchar(MAX)	<input type="checkbox"/>
	Observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 58 - tabela Parto**

- **Pressoterapia:** esta tabela regista informação sobre a pressoterapia (terapia por pressão, com ligaduras para as pernas). A tabela tem como chave primária três atributos: IdAtitude, IdPaciente e data. Como chaves estrangeiras tem os atributos IdAtitude e IdPaciente. A tabela possui os seguintes atributos:
  - **IdAtitude:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdAtitude da tabela Atitude (a atitude ‘Pressoterapia’ pode ser administrada diversas vezes). Regista o id’s da atitude ‘Pressoterapia’ da tabela Atitude. O atributo é do tipo inteiro e é obrigatório.
  - **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode realizar várias pressoterapias). Regista os id’s dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - **data:** é chave primária. Regista a data que a pressoterapia foi realizada ao paciente. O atributo é do tipo data e é obrigatório.
  - **membrosInferiores:** regista se a terapia por pressão é nos membros inferiores. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.
  - **membrosSuperiores:** regista se a terapia por pressão é nos membros superiores. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.


- **observacoes:** registo observações relativas ao procedimento. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.

Pressoterapia			
	Column Name	Data Type	Allow Nulls
🔑	IdAtitude	int	<input type="checkbox"/>
🔑	IdPaciente	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
	membrosInferiores	nvarchar(MAX)	<input checked="" type="checkbox"/>
	membrosSuperiores	nvarchar(MAX)	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 59 - tabela Pressoterapia**


- **ProdutoStock:** esta tabela regista os produtos que há em stock. A tabela tem como chave primária o atributo IdProdutoStock. Como chave estrangeira tem o atributo IdFornecedor. A tabela possui os seguintes atributos:
  - **IdProdutoStock:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **NomeProduto:** regista o nome do produto que está em stock. O atributo é do tipo string com tamanho máximo e é obrigatório.
  - **quantidadeArmazenada:** regista a quantidade que está armazenada em stock de determinado produto. O atributo é do tipo inteiro, e é obrigatório.
  - **precoUnitario:** regista o preço por unidade de cada produto. O atributo é do tipo numérico, com tamanho de 5 dígitos e 2 casas decimais, e é obrigatório.
  - **IdFornecedor:** é chave estrangeira, tem uma relação de 1 para N com o atributo IdFornecedor da tabela Paciente (um fornecedor pode ter fornecedor vários produtos. Regista os id's dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - **observacoes:** o enfermeiro regista observações relativas ao procedimento realizado. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.



ProdutoStock			
	Column Name	Data Type	Allow Nulls
	IdProdutoStock	int	<input type="checkbox"/>
	NomeProduto	nvarchar(MAX)	<input type="checkbox"/>
	quantidadeArmazenada	int	<input type="checkbox"/>
	taxaIVA	int	<input type="checkbox"/>
	precoUnitario	numeric(5, 2)	<input type="checkbox"/>
	Observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
	IdFornecedor	int	<input type="checkbox"/>
			<input type="checkbox"/>

**Figura 60 - tabela ProdutoStock**

- **Profissao:** esta tabela armazena informação relativa as profissões dos pacientes. A tabela tem como chave primária o atributo: IdProfissao. A tabela possui os seguintes atributos:
  - **IdProfissao:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **nomeProfissao:** regista o nome da profissão. O atributo é do tipo *string* com tamanho máximo, e é obrigatório.

Profissao			
	Column Name	Data Type	Allow Nulls
	IdProfissao	int	<input type="checkbox"/>
	nomeProfissao	nvarchar(MAX)	<input type="checkbox"/>
			<input type="checkbox"/>

**Figura 61 - tabela Profissao**

- **Suturas:** esta tabela regista informação das suturas (cozer). A tabela tem como chave primária três atributos: IdAtitude, IdPaciente e data. Como chaves estrangeiras tem os atributos IdAtitude e IdPaciente. A tabela possui os seguintes atributos:
  - **IdAtitude:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdAtitude da tabela Atitude (a atitude

‘Suturas’ pode ser realizada diversas vezes). Regista o id’s da atitude ‘Suturas’ da tabela Atitude. O atributo é do tipo inteiro e é obrigatório.

- **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode realizar várias suturas). Regista os id’s dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
- **data:** é chave primária. Regista em que data o paciente realizou as suturas. O atributo é do tipo data e é obrigatório.
- **id:** regista o ponto intradérmico. O atributo é do tipo inteiro, e não é obrigatório.
- **natural:** regista se a sutura é natural. O atributo é do tipo inteiro, e não é obrigatório.
- **donatti:** regista se a sutura é *donatti*. O atributo é do tipo inteiro, e não é obrigatório.
- **observacoes:** o enfermeiro regista observações relativas às suturas do paciente. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

Suturas			
	Column Name	Data Type	Allow Nulls
🔑	IdAtitude	int	<input type="checkbox"/>
🔑	IdPaciente	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
	id	int	<input checked="" type="checkbox"/>
	[natural]	int	<input checked="" type="checkbox"/>
	donatti	int	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 62 - tabela Suturas**

- **TesteAcuidadeVisual:** esta tabela regista informação da escala para ver a distância que a pessoa consegue ver (teste de visão). A tabela tem como chave primária três atributos: IdAtitude, IdPaciente e data. Como chaves estrangeiras tem os atributos IdAtitude e IdPaciente. A tabela possui os seguintes atributos:

- **IdAtitude:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdAtitude da tabela Atitude (a atitude ‘Teste de Acuidade Visual’ pode ser realizada diversas vezes). Regista o id’s da atitude ‘Teste de Acuidade Visual’ da tabela Atitude. O atributo é do tipo inteiro e é obrigatório.
- **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode realizar vários testes). Regista os id’s dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
- **data:** é chave primária. Regista em data que o paciente realizou o teste de acuidade visual. O atributo é do tipo data e é obrigatório.
- **testeAcuidadeVisual:** regista informação adicional sobre o teste. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório
- **observacoes:** o enfermeiro regista observações relativas às suturas do paciente. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

TesteAcuidadeVisual			
	Column Name	Data Type	Allow Nulls
🔑	IdAtitude	int	<input type="checkbox"/>
🔑	IdPaciente	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
	testeAcuidadeVisual	nvarchar(MAX)	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 63 - tabela TesteAcuidadeVisual**

- **TesteCombur:** esta tabela regista informação de teste à urina. A tabela tem como chave primária três atributos: IdAtitude, IdPaciente e data. Como chaves estrangeiras tem os atributos IdAtitude e IdPaciente. A tabela possui os seguintes atributos:
  - **IdAtitude:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdAtitude da tabela Atitude (a atitude

‘Teste de *Combur*’ pode ser realizada diversas vezes). Regista o id’s da atitude ‘Teste de *Combur*’ da tabela Atitude. O atributo é do tipo inteiro e é obrigatório.

- **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode realizar vários testes). Regista os id’s dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
- **data:** é chave primária. Regista em que data o paciente realizou o teste de acuidade visual. O atributo é do tipo data e é obrigatório.
- **densidadeV1:** regista a densidade da urina (1000). O atributo é do tipo inteiro, e não é obrigatório.
- **densidadeV2:** regista a densidade da urina (1005). O atributo é do tipo inteiro, e não é obrigatório.
- **densidadeV3:** regista a densidade da urina (1010). O atributo é do tipo inteiro, e não é obrigatório.
- **densidadeV4:** regista a densidade da urina (1015). O atributo é do tipo inteiro, e não é obrigatório.
- **densidadeV5:** regista a densidade da urina (1020). O atributo é do tipo inteiro, e não é obrigatório.
- **densidadeV6:** regista a densidade da urina (1025). O atributo é do tipo inteiro, e não é obrigatório.
- **densidadeV7:** regista a densidade da urina (1030). O atributo é do tipo inteiro, e não é obrigatório.
- **ph:** regista o valor do ph (de 5 a 9). O atributo é do tipo inteiro, e não é obrigatório.
- **leucocitos:** regista os leucócitos (Negativos, 1+, 2+ ou 3+). O atributo e do tipo *string* com tamanho 100, e não é obrigatório.
- **nitritos:** regista os nitritos (Negativos ou positivos). O atributo e do tipo *string* com tamanho 100, e não é obrigatório.
- **proteínas:** regista as proteínas (negativas, 1+, 2+ ou 3+). O atributo e do tipo *string* com tamanho 100, e não é obrigatório.
- **glucose:** regista a glucose (negativa, 1+, 2+, 3, ou 4+). O atributo e do tipo *string* com tamanho 100, e não é obrigatório.


- **cocetonicos**: regista os cocetonicos (negativos, 1+, 2+ ou 3+). O atributo é do tipo *string* com tamanho 100, e não é obrigatório.
- **sangeHemoglobina**: regista o valor de sangue (negativo, 1+, 2+, 3+ ou 4+). O atributo é do tipo *string* com tamanho 100, e não é obrigatório.
- **observacoes**: o enfermeiro regista observações relativas às suturas do paciente. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

TesteCombur			
	Column Name	Data Type	Allow Nulls
🔑	IdAtitude	int	<input type="checkbox"/>
🔑	IdPaciente	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
	densidadeV1	int	<input checked="" type="checkbox"/>
	densidadeV2	int	<input checked="" type="checkbox"/>
	densidadeV3	int	<input checked="" type="checkbox"/>
	densidadeV4	int	<input checked="" type="checkbox"/>
	densidadeV5	int	<input checked="" type="checkbox"/>
	densidadeV6	int	<input checked="" type="checkbox"/>
	densidadeV7	int	<input checked="" type="checkbox"/>
	ph	int	<input checked="" type="checkbox"/>
	[leucocitos ]	nvarchar(100)	<input checked="" type="checkbox"/>
	nitritos	nvarchar(100)	<input checked="" type="checkbox"/>
	proteinas	nvarchar(100)	<input checked="" type="checkbox"/>
	glucose	nvarchar(100)	<input checked="" type="checkbox"/>
	cocetonicos	nvarchar(100)	<input checked="" type="checkbox"/>
	sangeHemoglobina	nvarchar(100)	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 64 - tabela TesteCombur**


- **tipoDespesa**: está tabela regista os possíveis tipos de despesas que a clínica pode ter. A tabela tem como chave primária o atributo: IdTipoDespesa. A tabela possui os seguintes atributos (só o administrador tem acesso a esta tabela):
  - **IdTipoDespesa**: é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.

- **designacao:** especificação do tipo de despesa. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.
- **observacoes:** regista observações relativas ao tipo de despesa. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

tipoDespesa			
	Column Name	Data Type	Allow Nulls
	IdTipoDespesa	int	<input type="checkbox"/>
	designacao	nvarchar(MAX)	<input type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 65 - tabela tipoDespesa**

- **tipoExame:** está tabela regista os possíveis tipos de despesas que a clínica pode ter. A tabela tem como chave primária o atributo: IdTipoDespesa. A tabela possui os seguintes atributos:
  - **IdTipoDespesa:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **designacao:** especificação do tipo de despesa. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.
  - **observacoes:** regista observações relativas ao tipo de despesa. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

tipoExame			
	Column Name	Data Type	Allow Nulls
	IdTipoExame	int	<input type="checkbox"/>
	nome	nvarchar(MAX)	<input type="checkbox"/>
	categoria	nvarchar(MAX)	<input checked="" type="checkbox"/>
	designacao	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 66 - tabela tipoExame**

- **tipoQueimadura:** está tabela regista os possíveis tipos de queimaduras que um paciente pode ter. A tabela tem como chave primária o atributo: IdTipoQueimadura. A tabela possui os seguintes atributos:
  - **IdTipoQueimadura:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **tipoQueimadura:** especificação do tipo de queimadura. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.
  -


tipoQueimadura			
	Column Name	Data Type	Allow Nulls
	 IdTipoQueimadura	int	<input type="checkbox"/>
	tipoQueimadura	nvarchar(MAX)	<input type="checkbox"/>
			<input type="checkbox"/>

Figura 67 - tabela tipoQueimadura

- **tipoUlcera:** está tabela regista os possíveis tipos de úlceras que o paciente pode ter. A tabela tem como chave primária o atributo: IdTipoUlcera. A tabela possui os seguintes atributos:
  - **IdTipoUlcera:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **tipoUlcera:** especificação do tipo de úlcera. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.



tipoUlcera			
	Column Name	Data Type	Allow Nulls
	 IdTipoUlcera	int	<input type="checkbox"/>
	tipoUlcera	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Figura 68 - tabela tipoUlcera

- **Tratamento:** está tabela regista os possíveis tipos de tratamentos. A tabela tem como chave primária o atributo: IdTratamento. A tabela possui os seguintes atributos:
  - **IdTratamento:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **nomeTratamento:** especificação do nome do tratamento. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.

Tratamento			
	Column Name	Data Type	Allow Nulls
	IdTratamento	int	<input type="checkbox"/>
	nomeTratamento	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 69 - tabela Tratamento**

- **TratamentoExcisoes:** esta tabela regista informação de registar corpos estranhos com material cortante. A tabela tem como chave primária três atributos: IdTratamento, IdPaciente e data. Como chaves estrangeiras tem os atributos IdAtitude e IdPaciente. A tabela possui os seguintes atributos:
  - **IdTratamento:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdTratamento da tabela Tratamento. Regista o id's dos tratamentos da tabela Tratamento. O atributo é do tipo inteiro e é obrigatório.
  - **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode realizar vários tratamentos). Regista os id's dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - **data:** é chave primária. Regista em que data o paciente realizou o tratamento. O atributo é do tipo data e é obrigatório.
  - **numeroTratamento:** regista o número do tratamento (se é o 1º, o 2º, 3º, e consecutivamente). O atributo é do tipo inteiro e não é obrigatório.
  - **corpoEstranho:** regista se tem corpos estranhos. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.




- **dérmica:** regista em que local é o tratamento. O atributo é do tipo *string* com tamanho máximo e não é obrigatório.
- **observacoes:** regista observações relativas ao tratamento. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.
- **dataProximoTratamento:** regista a data do próximo tratamento. O atributo é do tipo *date*, e não é obrigatório.

TratamentoExcisoes			
	Column Name	Data Type	Allow Nulls
🔑	IdTratamento	int	<input type="checkbox"/>
🔑	IdPaciente	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
	numeroTratamento	int	<input checked="" type="checkbox"/>
	corpoEstranho	nvarchar(MAX)	<input checked="" type="checkbox"/>
	dermica	nvarchar(MAX)	<input checked="" type="checkbox"/>
	Observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
	dataProximoTratamento	date	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 70 - tabela TratamentoExcisoes**

- **TratamentoMaosPes:** está tabela regista os possíveis tipos de tratamentos possíveis nas mãos e nos pés. A tabela tem como chave primária o atributo: IdTratamentoMaosPes. A tabela possui os seguintes atributos:
  - **IdTratamentoMaosPes:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **tratamento:** regista o tratamento que o paciente faz. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.
  - **observacoes:** regista observações relativas ao tratamento. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.




TratamentoMaosPes			
	Column Name	Data Type	Allow Nulls
	IdTratamentoMaosPes	int	<input type="checkbox"/>
	tratamento	nvarchar(MAX)	<input type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 71 - tabela TratamentoMaosPes**

- **TratamentoPaciente:** esta tabela registra informação relativa aos tratamentos que o paciente pode realizar. A tabela tem como chave primária três atributos: IdTratamento, IdPaciente e data. Como chaves estrangeiras tem os atributos IdTratamento e IdPaciente. A tabela possui os seguintes atributos:
  - **IdTratamento:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdTratamento da tabela Tratamento. Regista o id's dos tratamentos da tabela Tratamento. O atributo é do tipo inteiro e é obrigatório.
  - **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode realizar vários tratamentos). Regista os id's dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - **data:** é chave primária. Regista em que data o paciente realizou o tratamento. O atributo é do tipo data e é obrigatório.
  - **numeroTratamento:** registra o número do tratamento (se é o 1º, o 2º, 3º, e consecutivamente). O atributo é do tipo inteiro e não é obrigatório.
  - **dimensoes:** registra a dimensão da ferida. O atributo é do tipo *string* com tamanho 50 e não é obrigatório.
  - **grauUlceraPressao:** registra a escala de avaliação da gravidade da úlcera. O atributo é do tipo *string* com tamanho 50 e não é obrigatório.
  - **exsudadoTipo:** registra o fluído orgânico que sai das feridas. O atributo é do tipo inteiro e não é obrigatório.

- **exsudadoQuantidade:** O atributo é do tipo *string* com tamanho 50 e não é obrigatório.
- **exsudadoCheiro:** regista se a ferida tem cheiro. O atributo é do tipo *string* com tamanho 100 e não é obrigatório.
- **tecidoPredominante:** regista os tecidos d ferida. O atributo é do tipo *string* com tamanho 500 e não é obrigatório.
- **areaCircundante:** regista o estado à volta da ferida (inchada, vermelha, escamativa, seca, etc). O atributo é do tipo *string* com tamanho 100 e não é obrigatório.
- **agenteLimpeza:** registo o que é usado para limpar a ferida. O atributo é do tipo *string* com tamanho 100 e não é obrigatório.
- **aplicacaoFerida:** regista os materiais que existem para tratar feridas. O atributo é do tipo *string* com tamanho 100 e não é obrigatório.
- **aplicacaoAreaCircundante:** registo o que é aplicado à volta. O atributo é do tipo *string* com tamanho 100 e não é obrigatório.
- **aplicacaoPenso:** regista se é colocado penso. O atributo é do tipo *string* com tamanho 100 e não é obrigatório.
- **aplicacaoTamanho:** regista o tamanho do penso. O atributo é do inteiro e não é obrigatório.
- **aplicacaoSuportePenso:** regista se o suporte do penso é ligadura ou adesivo. O atributo é do tipo *string* com tamanho 100 e não é obrigatório.
- **ProximoTratamento:** regista a data do próximo tratamento. O atributo é do tipo data e não é obrigatório.
- **Observacoes:** regista observações relativas ao tratamento. O atributo é do tipo *string* com tamanho 50 e não é obrigatório.
- **EscalaDor:** regista a dor do paciente. O atributo é do tipo *string* com tamanho 50 e não é obrigatório.
- **tipoQueimadura:** se o paciente tiver uma queimadura regista o tipo de queimadura. é chave estrangeira, tem uma relação de 1 para N com o atributo idTipoQueimadura da tabela tipoQueimadura. O atributo é do tipo inteiro e é obrigatório.
- **IPTB:** regista o índice de pressão tornozelo braço, que é a comparação da tensão arterial do braço e da perna). O atributo é do tipo *string* com tamanho 50 e não é obrigatório.

- **tipoUlcera:** é chave estrangeira, tem uma relação de 1 para N com o atributo idTipoUlcera da tabela tipoUlcera. O atributo é do tipo inteiro e não é obrigatório.

TratamentoPaciente			
	Column Name	Data Type	Allow Nulls
	IdTratamento	int	<input type="checkbox"/>
	IdPaciente	int	<input type="checkbox"/>
	data	date	<input type="checkbox"/>
	numeroTratamento	int	<input checked="" type="checkbox"/>
	dimensoes	nvarchar(50)	<input checked="" type="checkbox"/>
	grauUlceraPressao	nvarchar(50)	<input checked="" type="checkbox"/>
	exsudadoTipo	nvarchar(50)	<input checked="" type="checkbox"/>
	exsudadoQuantidade	int	<input checked="" type="checkbox"/>
	exsudadoCheiro	nvarchar(100)	<input checked="" type="checkbox"/>
	[tecidoPredominante ]	nvarchar(500)	<input checked="" type="checkbox"/>
	areaCircundante	nvarchar(100)	<input checked="" type="checkbox"/>
	agenteLimpeza	nvarchar(100)	<input checked="" type="checkbox"/>
	aplicacaoFerida	nvarchar(100)	<input checked="" type="checkbox"/>
	aplicacaoAreaCircundante	nvarchar(100)	<input checked="" type="checkbox"/>
	aplicacaoPenso	nvarchar(100)	<input checked="" type="checkbox"/>
	aplicacaoTamanho	int	<input checked="" type="checkbox"/>
	aplicacaoSuportePenso	nvarchar(100)	<input checked="" type="checkbox"/>
	ProximoTratamento	date	<input checked="" type="checkbox"/>
	Observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
	EscalaDor	nvarchar(30)	<input checked="" type="checkbox"/>
	tipoQueimadura	int	<input checked="" type="checkbox"/>
	IPTB	nvarchar(50)	<input checked="" type="checkbox"/>
	tipoUlcera	int	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 72 - tabela TratamentoPaciente**

- **Tricotomia:** esta tabela regista informação da retirada de pêlos. A tabela tem como chave primária três atributos: IdAtitude, IdPaciente e data. Como chaves estrangeiras tem os atributos IdAtitude e IdPaciente. A tabela possui os seguintes atributos:

- **IdAtitude:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdAtitude da tabela Atitude (a atitude ‘Tricotomia’ pode ser realizada diversas vezes). Regista o id’s da atitude ‘Tricotomia’ da tabela Atitude. O atributo é do tipo inteiro e é obrigatório.
- **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode realizar várias tricotomias). Regista os id’s dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
- **data:** é chave primária. Regista em que data o paciente realizou a tricotomia. O atributo é do tipo data e é obrigatório.
- **tricotomia:** regista dados da tricotomia que o paciente realizou. O atributo é do tipo *string* com tamanho máximo, e não é obrigatório.
- **observacoes:** regista observações relativas ao tratamento. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

Tricotomia			
	Column Name	Data Type	Allow Nulls
🔑	IdAtitude	int	<input type="checkbox"/>
🔑	IdPaciente	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
	tricotomia	nvarchar(MAX)	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 73 - tabela Tricotomia**

- **Vacinacao:** esta tabela armazena informação relativa à vacinação do paciente. A tabela tem como chave primária o atributo: IdVacinacao. Como chave estrangeira tem o atributo IdPaciente. A tabela possui os seguintes atributos:
  - **IdVacinacao:** é chave primária. Guarda o id da tabela com incremento automático de 1 em 1. O atributo é do tipo inteiro e é obrigatório.
  - **IdPaciente:** é chave estrangeira, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente. Regista os id’s dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.

- **data:** Regista a data que o paciente foi vacinado. O atributo é do tipo data e é obrigatório.
- **nomeVacina:** regista a vacina que o paciente levou. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.
- **marcaComercial:** regista a marca da vacina que o paciente levou. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.
- **numeroInoculacao:** regista o número da vacina que o paciente levou. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.
- **lote:** regista o lote da vacina que o paciente levou. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.
- **local:** regista o local onde a vacina foi administrada. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.
- **Escalador:** regista a escala da dor do paciente apos ter levado a vacina. O atributo é do tipo *string*, com tamanho 50 e não é obrigatório.
- **observacoes:** regista observações relativas à vacinação do paciente. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

Vacinação			
	Column Name	Data Type	Allow Nulls
🔑	IdVacinacao	int	<input type="checkbox"/>
	IdPaciente	int	<input checked="" type="checkbox"/>
	data	date	<input checked="" type="checkbox"/>
	nomeVacina	nvarchar(MAX)	<input checked="" type="checkbox"/>
	marcaComercial	nvarchar(MAX)	<input checked="" type="checkbox"/>
	numeroInoculacao	nvarchar(MAX)	<input checked="" type="checkbox"/>
	lote	nvarchar(MAX)	<input checked="" type="checkbox"/>
	local	nvarchar(MAX)	<input checked="" type="checkbox"/>
	escalaDor	nvarchar(50)	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 74 - tabela Vacinacao**




- **VariasAtitudes:** esta tabela regista informação relativa a vários tratamentos. A tabela tem como chave primária três atributos: IdAtitude, IdPaciente e data. Como chaves estrangeiras tem os atributos IdAtitude e IdPaciente. A tabela possui os seguintes atributos:
  - **IdAtitude:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdAtitude da tabela. Regista o id's da atitude 'Tricotomia' da tabela Atitude. O atributo é do tipo inteiro e é obrigatório.
  - **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente. Regista os id's dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
  - **data:** é chave primária. Regista em que data o paciente realizou a tricotomia. O atributo é do tipo data e é obrigatório.

VariasAtitudes			
	Column Name	Data Type	Allow Nulls
🔑	IdAtitude	int	<input type="checkbox"/>
🔑	IdPaciente	int	<input type="checkbox"/>
🔑	data	date	<input type="checkbox"/>
			<input type="checkbox"/>

**Figura 75 - tabela VariasAtitudes**

- **ZaragatoaOrofaringe:** esta tabela regista informação relativa ao teste da COVID-19, regista quando o teste é realizado ao paciente. A tabela tem como chave primária três atributos: IdAtitude, IdPaciente e data. Como chaves estrangeiras tem os atributos IdAtitude e IdPaciente. A tabela possui os seguintes atributos:
  - **IdAtitude:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdAtitude da tabela Atitude (a atitude 'Zaragatoa Orofaringe' pode ser realizada diversas vezes). Regista o id's da atitude 'Zaragatoa Orofaringe' da tabela Atitude. O atributo é do tipo inteiro e é obrigatório.

- **IdPaciente:** é chave primária e chave estrangeira ao mesmo tempo, tem uma relação de 1 para N com o atributo IdPaciente da tabela Paciente (um paciente pode realizar vários testes). Regista os id's dos Pacientes da tabela Paciente. O atributo é do tipo inteiro e é obrigatório.
- **data:** é chave primária. Regista em que data o paciente realizou a tricotomia. O atributo é do tipo data e é obrigatório.
- **zaragatoaOrofaringe:** regista informação relativa ao teste. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.
- **observacoes:** regista observações relativas à realização do teste. O atributo é do tipo *string*, com tamanho máximo e não é obrigatório.

ZaragatoaOrofaringe			
	Column Name	Data Type	Allow Nulls
	IdAtitude	int	<input type="checkbox"/>
	IdPaciente	int	<input type="checkbox"/>
	data	date	<input type="checkbox"/>
	zaragatoaOrofaringe	nvarchar(MAX)	<input checked="" type="checkbox"/>
	observacoes	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

**Figura 76 - tabela ZaragatoaOrofaringe**



## 15.4. Anexo D - Autoproposta do projeto

Este anexo contém a autoproposta que foi feita do projeto



DEPARTAMENTO  
DE  
ENGENHARIA  
INFORMÁTICA

### Projeto Informático

## Gestão de Clínica de Enfermagem

### Área Temática:

AIBD    Aplicações Integradas de Base de Dados

DA       Desenvolvimento de Aplicações

**Descrição:** Pretende-se implementar uma aplicação desktop que permita fazer a gestão de uma clínica de enfermagem. A solução será desenvolvida na linguagem de programação C# e terá a orientação da administração da clínica de enfermagem *Siltes Saúde*.

Esta aplicação tem como requisitos, ter um sistema de autenticação, ter um *dashboard* onde se pode ver as consultas do dia atual (nome utente, hora consulta e enfermeiro responsável pelo utente), efetuar a gestão de utentes (podendo inserir, atualizar utentes). Efetuar a gestão de consultas podendo registar os antecedentes clínicos do utente (registar doenças crónicas existentes e antecedentes familiares), registar a medicação crónica (se existir), registar a avaliação objetiva (dados levantados na primeira consulta, exemplo peso, altura, IMC), registar a informação sobre exames laboratoriais que o utente leve para a consulta. Registar os dados das consultas, como sintomatologia, dor, registar o diagnóstico efetuado: plano cuidados e terapêutica prescrita. Registar a data da próxima consulta ou tratamento (contagem de dias) e indicar se o utente tem alta ou não. Efetuar um inventário de todos os materiais e medicamentos existentes na clínica.

Quando os materiais ou medicamentos da clínica chegarem a um determinado limite, a aplicação lança um alarme em como e necessário repor o stock. Para a administração é importante, registar as despesas da clínica de enfermagem bem como as receitas obtidas nas consultas ou tratamento.

**Módulos do projeto** (*2 estudantes em cada módulo e podem definir de 1 até 4 módulos*):

- Módulo I: (*a ser desenvolvido pelo grupo I*)

**Responsáveis pela autoproposta:** Andreia Fernandes nº 2160844 e Beatriz Ribeiro nº 2160796. Esta proposta de projeto surgiu através duma conversa com a administração da clínica que não tem qualquer sistema informático e sente falta de ter um.

**Entidade externa participante:** *Siltes Saúde – Clínica de Enfermagem*

## **Ficha técnica do projeto**

**Projeto Informático:** Gestão de clínica de Enfermagem

**Área:** Sistemas de Informação

**Título:** Siltes Saúde

**Orientador 1 (definido posteriormente pelo responsável da UC):**

**Orientador 2 (definido posteriormente pelo responsável da UC):**

**Sistema operativo:** Windows

**Outro software:** Microsoft Visual Studio

**Software para adquirir:** N/A

**Hardware:** Computador

**Hardware para adquirir:** N/A

**Observações:**



### 15.5. Anexo E - US Desenvolvidas por

Este anexo contém as US, e quem desenvolveu cada uma

<b>Designação</b>	<b>Caso de Uso</b>	<b>Desenvolvido por</b>
<b>US1</b>	Registar Enfermeiro	Beatriz Ribeiro e Andreia Fernandes
<b>US2</b>	Criar Listagem de Enfermeiros Registados	Beatriz Ribeiro e Andreia Fernandes
<b>US3</b>	Registar Produtos associados a um fornecedor	Andreia Fernandes
<b>US4</b>	Criar Listagem dos produtos.	Andreia Fernandes
<b>US5</b>	Registar Fornecedores	Andreia Fernandes
<b>US6</b>	Criar Listagem de Fornecedores Registados	Andreia Fernandes
<b>US7</b>	Pesquisar fornecedor	Andreia Fernandes
<b>US8</b>	Registar Encomendas	Andreia Fernandes
<b>US9</b>	Criar Listagem de Encomendas	Andreia Fernandes
<b>US10</b>	Finalizar uma encomenda.	Andreia Fernandes
<b>US11</b>	Visualizar encomendas.	Andreia Fernandes
<b>US12</b>	Registar Despesas da Clínica	Andreia Fernandes
<b>US13</b>	Criar Listagem de despesas	Andreia Fernandes
<b>US14</b>	Registar Tipo de Encomenda	Andreia Fernandes
<b>US15</b>	Pesquisar despesas	Andreia Fernandes
<b>US16</b>	Criar Listagem de despesas.	Andreia Fernandes
<b>US17</b>	<b>Pesquisar um Fornecedor</b>	<b>Andreia Fernandes</b>

**Tabela 6 – tabela de quem desenvolveu os requisitos de Administrador**

<b>Designação</b>	<b>Caso de Uso</b>	<b>Desenvolvido por</b>
<b>US18</b>	Registar Utente	Beatriz Ribeiro e Andreia Fernandes
<b>US19</b>	Imprimir o Consentimento do uso de Dados	Andreia Fernandes
<b>US20</b>	Criar Listagem dos Utentes	Beatriz Ribeiro e Andreia Fernandes
<b>US21</b>	Pesquisar Utentes	Andreia Fernandes
<b>US22</b>	Criar Listagem dos detalhes dos Utentes	Andreia Fernandes
<b>US23</b>	Editar Dados do Utente	Andreia Fernandes
<b>US24</b>	Marcar Consulta	Andreia Fernandes
<b>US25</b>	Criar Listagem das consultas marcadas	Andreia Fernandes
<b>US26</b>	Iniciar uma consulta sem marcação	Andreia Fernandes
<b>US27</b>	Criar Listagem dos detalhes das consultas	Andreia Fernandes
<b>US28</b>	Registar a Localização da dor	Andreia Fernandes e Beatriz Fernandes
<b>US29</b>	Criar Listagem do registo da localização da dor	Andreia Fernandes e Beatriz Fernandes
<b>US30</b>	Registar Doenças do utente	Andreia Fernandes
<b>US31</b>	Criar Listagem das doenças	Andreia Fernandes
<b>US32</b>	Pesquisar Doenças	Andreia Fernandes
<b>US33</b>	Registar uma nova Doença	Andreia Fernandes
<b>US34</b>	Criar Listagem das doenças registadas	Andreia Fernandes
<b>US35</b>	Editar uma doença	Andreia Fernandes
<b>US36</b>	Registar uma Alergia do utente	Andreia Fernandes
<b>US37</b>	Pesquisar Alergia	Andreia Fernandes
<b>US38</b>	Registar uma nova Alergia	Andreia Fernandes
<b>US39</b>	Criar Listagem das alergias registadas	Andreia Fernandes
<b>US40</b>	Editar uma alergia	Andreia Fernandes
<b>US41</b>	Registar uma Cirurgia do utente	Andreia Fernandes
<b>US42</b>	Registar uma nova Cirurgia	Andreia Fernandes
<b>US43</b>	Criar Listagem das cirurgias registadas	Andreia Fernandes
<b>US44</b>	Pesquisar Cirurgia	Andreia Fernandes
<b>US45</b>	Registar um exame do utente	Andreia Fernandes
<b>US46</b>	Registar um novo Exame	Andreia Fernandes
<b>US47</b>	Criar listagem dos exames registados	Andreia Fernandes
<b>US48</b>	Pesquisar Exame	Andreia Fernandes

<b>US49</b>	Registar Avaliação objetiva do utente	Andreia Fernandes
<b>US50</b>	Criar listagem das avaliações objetivo registadas.	Andreia Fernandes
<b>US51</b>	Registar Métodos Contracetivos	Andreia Fernandes
<b>US52</b>	Registar Análises Laboratoriais efetuadas pelo utente	Andreia Fernandes
<b>US53</b>	Criar listagem das análises laboratoriais registadas	Andreia Fernandes
<b>US54</b>	Registar Análises Laboratoriais	Andreia Fernandes
<b>US55</b>	Registar Tratamentos	Andreia Fernandes
<b>US56</b>	Criar listagem dos detalhes dos tratamentos registados anteriormente.	Andreia Fernandes
<b>US57</b>	Registar Excisões	Andreia Fernandes
<b>US58</b>	Registar Ferida Cirúrgica	Andreia Fernandes
<b>US59</b>	Registar Ferida Traumática	Andreia Fernandes
<b>US60</b>	Registar Doppler arterial/venoso	Andreia Fernandes e Beatriz Fernandes
<b>US61</b>	Registar Doppler fetal	Andreia Fernandes e Beatriz Fernandes
<b>US62</b>	Criar listagem do histórico do doppler fetal.	Andreia Fernandes e Beatriz Fernandes
<b>US63</b>	Registar Espirometria	Andreia Fernandes e Beatriz Fernandes
<b>US64</b>	Criar listagem do histórico da espirometria.	Andreia Fernandes e Beatriz Fernandes
<b>US65</b>	Registar onicocriptoses, onicomicoses e pé diabético.	Andreia Fernandes e Beatriz Fernandes
<b>US66</b>	Registar Vacinação	Andreia Fernandes e Beatriz Fernandes
<b>US67</b>	Criar listagem do histórico das vacinações	Andreia Fernandes e Beatriz Fernandes
<b>US68</b>	Registar tipos de Queimaduras	Andreia Fernandes
<b>US69</b>	Criar listagem dos tipos de Queimaduras	Andreia Fernandes
<b>US70</b>	Registar Queimadura Iónica	Andreia Fernandes
<b>US71</b>	Registar Queimadura Química	Andreia Fernandes
<b>US72</b>	Registar Queimadura Solar	Andreia Fernandes
<b>US73</b>	Registar Queimadura Térmica	Andreia Fernandes
<b>US74</b>	Registar tipos de Úlceras	Andreia Fernandes

<b>US75</b>	Criar listagem dos tipos de Úlceras	Andreia Fernandes
<b>US76</b>	Registar Úlcera Arterial	Andreia Fernandes
<b>US77</b>	Registar Úlcera Mista	Andreia Fernandes
<b>US78</b>	Registar Úlcera de Pressão	Andreia Fernandes
<b>US79</b>	Registar Úlcera Venosa	Andreia Fernandes
<b>US80</b>	Registar a medicação prescrita	Andreia Fernandes
<b>US81</b>	Imprimir a medicação prescrita	Andreia Fernandes
<b>US82</b>	Criar listagem da medicação prescrita	Andreia Fernandes
<b>US83</b>	Criar listagem do histórico da medicação prescrita.	Andreia Fernandes
<b>US84</b>	Pesquisar Medicação prescrita	Andreia Fernandes
<b>US85</b>	Registar atitudes terapêuticas.	Andreia Fernandes
<b>US86</b>	Registar colheitas.	Andreia Fernandes
<b>US87</b>	Registar outras atitudes terapêuticas.	Andreia Fernandes
<b>US88</b>	Criar Listagem do histórico das atitudes terapêuticas.	Andreia Fernandes
<b>US89</b>	Registar a administração de medicação.	Andreia Fernandes
<b>US90</b>	Criar Listagem do histórico da administração de medicação.	Andreia Fernandes
<b>US91</b>	Registar a algaliação.	Andreia Fernandes
<b>US92</b>	Criar Listagem do histórico das algaliações.	Andreia Fernandes
<b>US93</b>	Registar as aspirações de secreções.	Andreia Fernandes
<b>US94</b>	Criar Listagem do histórico das aspirações de secreções.	Andreia Fernandes
<b>US95</b>	Registar dados no audiograma.	Andreia Fernandes e Beatriz Fernandes
<b>US96</b>	Imprimir dados do audiograma.	Andreia Fernandes e Beatriz Fernandes
<b>US97</b>	Registar cateterismo.	Andreia Fernandes
<b>US98</b>	Criar Listagem do histórico dos cateterismos.	Andreia Fernandes
<b>US99</b>	Registar colheita de sangue de diagnóstico precoce.	Andreia Fernandes
<b>US100</b>	Criar Listagem do histórico das colheitas de sangue precoce.	Andreia Fernandes
<b>US101</b>	Registar colheita de urina.	Andreia Fernandes
<b>US102</b>	Criar Listagem do histórico das colheitas de urina.	Andreia Fernandes
<b>US103</b>	Registar dados da crioterapia.	Andreia Fernandes e Beatriz Fernandes

<b>US104</b>	Criar Listagem do histórico da crioterapia.	Andreia Fernandes e Beatriz Fernandes
<b>US105</b>	Registar Drenagem de Locas	Andreia Fernandes
<b>US106</b>	Criar Listagem do histórico da drenagem de locas.	Andreia Fernandes
<b>US107</b>	Registar Desbridamento.	Andreia Fernandes
<b>US108</b>	Criar Listagem do histórico do desbridamento.	Andreia Fernandes
<b>US109</b>	Registar ENG.	Andreia Fernandes
<b>US110</b>	Criar Listagem do histórico de ENG.	Andreia Fernandes
<b>US111</b>	Registar Flebografia.	Andreia Fernandes
<b>US112</b>	Criar Listagem do histórico das flebogrfias.	Andreia Fernandes
<b>US113</b>	Registar Inalações.	Andreia Fernandes
<b>US114</b>	Criar Listagem do histórico das inalações.	Andreia Fernandes
<b>US115</b>	Registar lavagem auricular.	Andreia Fernandes
<b>US116</b>	Criar Listagem do histórico das lavagens auriculares.	Andreia Fernandes
<b>US117</b>	Registar lavagem ocular.	Andreia Fernandes
<b>US118</b>	Criar Listagem do histórico das lavagens oculares.	Andreia Fernandes
<b>US119</b>	Registar Lavagem Vesical.	Andreia Fernandes
<b>US120</b>	Criar Listagem do histórico das lavagens vesicais.	Andreia Fernandes
<b>US121</b>	Registar Monitorização ECG	Andreia Fernandes
<b>US122</b>	Criar Listagem do histórico da monitorização ECG.	Andreia Fernandes
<b>US123</b>	Registar Pressoterapia.	Andreia Fernandes
<b>US124</b>	Criar Listagem do histórico da Pressoterapia.	Andreia Fernandes
<b>US125</b>	Registar Sutura	Andreia Fernandes
<b>US126</b>	Criar Listagem do histórico das suturas.	Andreia Fernandes
<b>US127</b>	Registar Teste <i>Combur</i>	Andreia Fernandes
<b>US128</b>	Criar Listagem do histórico dos testes de <i>Combur</i> .	Andreia Fernandes
<b>US129</b>	Registar Teste de avaliação acuidade visual.	Andreia Fernandes
<b>US130</b>	Criar Listagem do histórico dos testes de avaliação acuidade visual.	Andreia Fernandes
<b>US131</b>	Registar Tricotomia	Andreia Fernandes
<b>US132</b>	Criar Listagem do histórico da tricotomia.	Andreia Fernandes
<b>US133</b>	Registar Zaragatoa Orofaringe	Andreia Fernandes
<b>US134</b>	Criar Listagem do histórico da zaragatoa orofaringe.	Andreia Fernandes
<b>US135</b>	Registar Colpocitologia	Andreia Fernandes
<b>US136</b>	Criar Listagem do histórico das colpocitologias.	Andreia Fernandes



<b>US137</b>	Registrar colocação de DIU	Andreia Fernandes
<b>US138</b>	Criar Listagem do histórico das colocações de DIU.	Andreia Fernandes
<b>US139</b>	Registrar Implante Contracetivo Dérmico.	Andreia Fernandes
<b>US140</b>	Criar Listagem do histórico dos implantes contracetivos dérmicos.	Andreia Fernandes
<b>US141</b>	Registrar avaliação objetiva do bebé até 36 meses.	Andreia Fernandes
<b>US142</b>	Criar Listagem do histórico das avaliações objetivas do bebé.	Andreia Fernandes
<b>US143</b>	Registrar um tipo de Aleitamento.	Andreia Fernandes
<b>US144</b>	Criar Listagem dos tipos de Aleitamento	Andreia Fernandes
<b>US145</b>	Editar Tipo de Aleitamento.	Andreia Fernandes
<b>US146</b>	Pesquisar um tipo de Aleitamento.	Andreia Fernandes
<b>US147</b>	Registrar Aleitamento materno.	Andreia Fernandes
<b>US148</b>	Registrar Aleitamento Artificial.	Andreia Fernandes
<b>US149</b>	Registrar Aleitamento Misto.	Andreia Fernandes
<b>US150</b>	Registrar um tipo de parto.	Andreia Fernandes
<b>US151</b>	Registrar eutócico.	Andreia Fernandes
<b>US152</b>	Registrar distócico.	Andreia Fernandes
<b>US153</b>	Registrar cesariana.	Andreia Fernandes
<b>US154</b>	Editar tipo de parto.	Andreia Fernandes
<b>US155</b>	Pesquisar um tipo de parto.	Andreia Fernandes
<b>US156</b>	Criar Listagem dos tipos de Parto	Andreia Fernandes
<b>US157</b>	Registrar os produtos utilizados numa consulta.	Andreia Fernandes
<b>US158</b>	Registrar os produtos utilizados numa consulta associados a um fornecedor.	Andreia Fernandes
<b>US159</b>	Criar Listagem dos fornecedores.	Andreia Fernandes
<b>US160</b>	Criar Listagem dos produtos em stock.	Andreia Fernandes
<b>US161</b>	Criar listagem dos produtos utilizados na consulta.	Andreia Fernandes
<b>US162</b>	Registrar a quantidade utilizada de cada produto.	Andreia Fernandes
<b>US163</b>	Decrementar os produtos utilizados.	Andreia Fernandes
<b>US164</b>	Retirar produtos usados na consulta.	Andreia Fernandes
<b>US165</b>	Registrar o valor da consulta.	Andreia Fernandes
<b>US166</b>	Alterar hora ou data de uma consulta.	Andreia Fernandes
<b>US167</b>	Pesquisar Consultas	Andreia Fernandes
<b>US168</b>	Criar Listagem das consultas registadas.	Andreia Fernandes

<b>US169</b>	Desmarcar consulta agendada.	Andreia Fernandes
<b>US170</b>	Editar dados do enfermeiro.	Andreia Fernandes
<b>US171</b>	Alterar password do enfermeiro.	Andreia Fernandes
<b>US172</b>	Criar listagem das consultas do dia.	Andreia Fernandes
<b>US173</b>	Registar Tipo de Atitude terapêutica.	Andreia Fernandes
<b>US174</b>	Registar um método contraceptivo.	Andreia Fernandes
<b>US175</b>	Criar listagem dos métodos contraceptivos.	Andreia Fernandes
<b>US176</b>	Editar métodos contraceptivos.	Andreia Fernandes
<b>US177</b>	Pesquisar método contraceptivo	Andreia Fernandes
<b>US178</b>	Registar profissões.	Andreia Fernandes
<b>US179</b>	Criar Listagem das profissões.	Andreia Fernandes
<b>US180</b>	Editar profissões.	Andreia Fernandes
<b>US181</b>	Pesquisar profissões.	Andreia Fernandes
<b>US182</b>	Recuperar <i>username</i> do utilizador.	Andreia Fernandes
<b>US183</b>	Enviar e-mail ao utilizador.	Andreia Fernandes
<b>US184</b>	Recuperar password do utilizador.	Andreia Fernandes
<b>US185</b>	Enviar e-mail ao utilizador, para recuperar a password.	Andreia Fernandes
<b>US186</b>	Login	Andreia Fernandes
<b>US187</b>	Primeiro Acesso	Andreia Fernandes
<b>US188</b>	Denições de Utilizador	Andreia Fernandes
<b>US189</b>	Denições de Utilizador	Andreia Fernandes
<b>US190</b>	Iniciar uma consulta com marcação	Andreia Fernandes
<b>US191</b>	<b>Ver consultas do próprio dia</b>	<b>Andreia Fernandes</b>

**Tabela 7 – tabela de quem desenvolveu os requisitos de Administrador e Utilizador Normal**

Designação	Descrição	Resolvido por:
<b>Erro1</b>	Desenho inicial do DER.	Andreia Fernandes
<b>Erro2</b>	Escolher a melhor plataforma gratuita para implementar a base de dados.	Andreia Fernandes
<b>Erro3</b>	Depois da funcionalidade para iniciar sessão estar implementada, a funcionalidade para registar enfermeiro deixou de funcionar.	Andreia Fernandes
<b>Erro4</b>	Encriptação da password.	Andreia Fernandes
<b>Erro5</b>	Tipo de campo contacto era <i>double</i> e causava problemas na inserção de dados no Enfermeiro.	Andreia Fernandes
<b>Erro6</b>	Qualquer enfermeiro com sessão iniciada tinha acesso a todos os utentes registados, mesmo que não tivesse sido ele a registar.	Andreia Fernandes
<b>Erro7</b>	Funcionalidade para recuperar password e <i>username</i> não funcionava com o Gmail.	Andreia Fernandes
<b>Erro8</b>	Gravar datas na base de dados.	Andreia Fernandes
<b>Erro9</b>	Janela para visualizar consultas por paciente é redundante.	Andreia Fernandes
<b>Erro10</b>	Eliminar dados de tabela, seleccionando uma linha da <i>DataGridView</i> .	Andreia Fernandes
<b>Erro11</b>	Quando a <i>dataGridView</i> estava vazia e eram adicionados dados posteriormente, ao ir à janela onde se pretendia visualizar os dados inseridos, dava erro de execução na <i>dataGridView</i> .	Andreia Fernandes
<b>Erro12</b>	Campo código-postal aparecia separado na <i>dataGridView</i> .	Andreia Fernandes
<b>Erro13</b>	Campos não obrigatórios ficam	Andreia Fernandes

	registados a <i>NULL</i> ou vazios na base de dados. Ao tentar ler estes dados, dava exceção.	
<b>Erro14</b>	Ao inserir dados que tenham o campo NIF e contacto no sistema aparece exceção.	Andreia Fernandes
<b>Erro15</b>	Campo email com restrição de <i>Constraint Unique</i> . Esta restrição faz com o que os dados sejam únicos, mas o email não é um campo obrigatório, logo quando são inseridos 2 utilizadores sem email, ficam registados com o valor <i>NULL</i> , e por este motivo dava exceção por já existir um valor igual.	Andreia Fernandes
<b>Erro16</b>	Exceções devido a chaves estrangeiras na base de dados.	Andreia Fernandes
<b>Erro17</b>	Editar dados do tipo de parto, tipo de aleitamento e métodos contraceptivos se não forem escritos de maneira exatamente igual dá erro.	Andreia Fernandes
<b>Erro18</b>	ID das encomendas estava a ser registado a 0.	Andreia Fernandes
<b>Erro19</b>	Círculos que se desenham no audiograma não ficavam no lugar onde o rato clicava.	Andreia Fernandes
<b>Erro20</b>	Impressão da imagem do audiograma ficava cortada.	Andreia Fernandes
<b>Erro21</b>	Inserção errada de dados na Linha de Encomenda.	Andreia Fernandes
<b>Erro22</b>	Editar dados de Enfermeiro, em que os campos NIF, email e número de SNS não são permitidos alterar. Mas ao guardar os outros campos alterados, aparecia uma exceção pois aqueles dados já estavam registados.	Andreia Fernandes

**Tabela 8 - tabela de quem desenvolveu**



## **15.6. Anexo F – Manual de Utilizador**

Este anexo apresenta o manual de utilizador.

Neste manual, não é dito que tipos de dados inserir em cada janela (se são números, string, etc), a pedido do administrador da clínica, por já saber o que inserir.

O manual por ser extenso, e por o cliente querer o manual, está num ficheiro separado ‘Manual de Utilização.pdf’