



**PROJETO GLOBAL**  
APLICAÇÃO WEB EM ASP.NET

LICENCIATURA EM INFORMÁTICA  
2015/2016

FERNANDO RICARDO GONCALVES  
ALUNO Nº 1926

Coordenador: Prof. Doutor Pedro Brandão

Lisboa, março 2017



**PROJETO GLOBAL**  
APLICAÇÃO WEB EM ASP.NET

LICENCIATURA EM INFORMÁTICA  
2015/2016

FERNANDO RICARDO GONCALVES  
ALUNO Nº 1926

Coordenador: Prof. Doutor Pedro Brandão

Lisboa, março 2017

(Página intencionalmente deixada em branco)

*À minha mulher que me apoiou na decisão de voltar a estudar  
e ao seu constante incentivo para ultrapassar esta etapa e  
todas as contrariedades que surgem na nossa vida.*

*Aos meus filhos pelo seu amor, é neles que vou buscar forças  
todos os dias.*

*Aos meus pais que não mediram esforços para me  
proporcionar a melhor educação possível.*

## Agradecimentos

Antes de mais, tenho de agradecer à minha família. Evidentemente, tenho de agradecer à minha mulher pelo seu apoio incondicional neste percurso académico. Aos meus pais que são um exemplo para mim. Sem o amor familiar não conseguia ultrapassar esta etapa depois das contrariedades ultrapassadas nestes últimos anos.

Um agradecimento a toda a estrutura do Instituto Superior de Tecnologias Avançadas que me proporcionou condições para realizar este meu objetivo. Aos professores que contribuíram para enriquecimento das minhas aptidões.

Agradeço igualmente aos colegas da Licenciatura com quem partilhei todas as experiências académicas e pessoais, de certo, a nossa amizade irá perdurar.

A todos, muito obrigado!

*“A felicidade está em usufruir e não apenas em possuir.”*

Montaigne

(Página intencionalmente deixada em branco)

## Resumo

Como finalista da Licenciatura em Informática do Instituto Superior de Tecnologias Avançadas de Lisboa é necessária a criação de um trabalho para a Unidade Curricular Projeto Global. O projeto tem o intuito de provar capacidades sobre os conhecimentos adquiridos em toda a licenciatura.

Devido ao aparecimento de tecnologias microeletrónicas e da transmissão de informações sobre a automatização e robotização dos processos, o panorama produtivo mundial alterou-se.

Este documento tem o objetivo explicar o desenvolvimento de uma Aplicação Web desde o seu esboço até à sua implementação. O tópico dos sistemas de bases de dados, mais propriamente de *Structured Query Language*, é igualmente referido pois é integrado e disponibilizado através de ASP.NET.

A solução exposta é de um sistema de registos clínicos e de desenvolvimento no qual pretende melhorar alguns dos já existentes onde tive oportunidade de constatar as suas falhas.

**Palavras chave:** ASP.NET, C#, HTTP, MVC, Registos, SQL, SGBD

## **Abstract**

As a finalist of the Degree in Computer Engineering of the Advanced Technology Institute of Lisbon, it is necessary to create a work for the Global Project Curricular Unit. The project aims to prove skills on the knowledge acquired throughout the degree.

Due to the emergence of microelectronic technologies and the transmission of information on automation and robotization of processes, the world production panorama has changed.

This document aims to explain the development of a Web Application from its sketch to its implementation. The topic of database systems, rather than *Structured Query Language*, is also referred to because it is integrated and made available through ASP.NET.

The solution exposed is a system of clinical and development records in which it intends to improve some of the existing ones where I had the opportunity to verify their failures.

**Key words:** ASP.NET, C#, HTTP, MVC, Records, SQL, DBMS

## Abreviaturas e Símbolos

|       |   |
|-------|---|
| ANSI  | American National Standards Institute         |
| API   | Application Programming Interface             |
| ASP   | Active Server Pages                           |
| BD    | Base de Dados                                 |
| HTTP  | Hypertext Transfer Protocol                   |
| IIS   | Internet Information Services                 |
| MVC   | Model-View-Controller                         |
| NIF   | Número de Identificação Fiscal                |
| SGBD  | Sistema de Gestão de Base de Dados            |
| SGBDR | Sistema de Gestão de Base de Dados Relacional |
| SQL   | Structured Query Language                     |
| WCF   | Windows Communication Foundation              |
| XML   | Extendible Markup Language                    |

(Página intencionalmente deixada em branco)

# Índice

|   |      |
|---|------|
| Dedicatória .....   | ii   |
| Agradecimentos .....  | iii  |
| Resumo .....  | v    |
| Abstract (+ key words) .....                                    | vi   |
| Abreviaturas e Símbolos .....                                   | vii  |
| Índice.....   | ix   |
| Índice de Figuras.....  | xi   |
| Índice de Tabelas .....   | xii  |
| Índice de Código .....  | xiii |
| Introdução .....  | 1    |
| 1. Estado da Arte .....   | 2    |
| 1.1. A organização e o seu enquadramento .....                  | 2    |
| 1.2. Aplicação Web e o seu contexto .....                       | 3    |
| 1.3. Desenvolvimento e evolução da plataforma ASP.NET .....     | 3    |
| 1.4. Migração.....  | 4    |
| 1.5. Dependência.....   | 5    |
| 1.6. Structured Query Language .....                            | 5    |
| 1.7. Tecnologias e Ferramentas para o Desenvolvimento Web ..... | 8    |
| 1.7.1. WCF .....  | 8    |
| 1.7.2. Entity Framework .....                                   | 8    |
| 1.7.3. LINQ.....  | 8    |
| 1.7.4. IIS.....   | 8    |
| 2. Contextualização .....                                       | 9    |
| 2.1. Motivação e Relevância .....                               | 9    |
| 2.2. Objeto de estudo.....                                      | 9    |

|                                       |    |
|---------------------------------------|----|
| 2.3. Objetivos .....                  | 9  |
| 3. Desenvolvimento .....              | 10 |
| 3.1. Esboço .....                     | 11 |
| 3.2. Modelagem.....                   | 11 |
| 3.2.1. Base de Dados.....             | 11 |
| 3.2.2. Desenvolvimento Front-End..... | 13 |
| 3.2.3. Desenvolvimento Back-End ..... | 20 |
| 4. Conclusão .....                    | 33 |
| Bibliografia .....                    | 34 |
| Anexos .....                          | 36 |

## Índice de Figuras

|  |    |
|--|----|
| Figura 1 - Fluxograma do Projeto Global, Fonte Própria .....                 | 10 |
| Figura 2 - Diagrama da base de dados, Fonte Própria .....                    | 11 |
| Figura 3 - alguns dos ficheiros .aspx da minha aplicação, Fonte Própria..... | 13 |
| Figura 4 - Visão Geral do Sistema de Gestão HOMsys, Fonte Própria.....       | 14 |
| Figura 5 - Página "Login", Fonte Própria .....                               | 14 |
| Figura 6 - Página "Painel Principal", Fonte Própria .....                    | 15 |
| Figura 7 - Página "Gestão de Clientes", Fonte Própria.....                   | 16 |
| Figura 8 - Página "Gestão de Registos", Fonte Própria .....                  | 17 |
| Figura 9 - Página "Gestão de Utilizadores", Fonte Própria .....              | 18 |
| Figura 10 - Página "Editar Perfil", Fonte Própria .....                      | 19 |

## Índice de Tabelas

|   |    |
|---|----|
| Tabela 1 - Ciclo de vida da página, Fonte Microsoft Developer Network ..... | 20 |
|---|----|

## Índice de Código

|   |    |
|---|----|
| Código 1 - Sintaxe para criação da tabela "Clientes", Fonte Própria .....   | 12 |
| Código 2 - Sintaxe para criação da tabela "Staff", Fonte Própria.....   | 12 |
| Código 3 - Sintaxe para criação da tabela "Registos", Fonte Própria.....  | 12 |
| Código 4 - Classe Principal do ficheiro "Login.aspx.cs", Fonte Própria.....   | 21 |
| Código 5 - Codebehind da Página de Login, Fonte Própria .....   | 22 |
| Código 6 - Método "Page_Load" do ficheiro "11GestaoClientes.aspx.cs", Fonte Própria .....   | 23 |
| Código 7 - Evento "btnSave_Click" do ficheiro "11GestaoClientes.aspx.cs", Fonte Própria.  | 24 |
| Código 8 – Evento "btnUpdate_Click" do ficheiro "11GestaoClientes.aspx.cs", Fonte Própria<br>.....  | 25 |
| Código 9 - Evento "btnDelete_Click" do ficheiro "11GestaoClientes.aspx.cs", Fonte Própria<br>.....  | 26 |
| Código 10 - Método "txtBoxSearch_TextChanged" do ficheiro "11GestaoClientes.aspx.cs",<br>Fonte Própria.....                               | 27 |
| Código 11 - Evento "btnUpload_Click" do ficheiro "11GestaoClientes.aspx.cs", Fonte<br>Própria.....  | 28 |
| Código 12 - Método "devolveTabela" do ficheiro "21GestaoRegistos.aspx.cs", Fonte Própria<br>.....   | 29 |
| Código 13 - Evento "txtBoxSearch_TextChanged" e Método "mostraNomeCliente" do<br>ficheiro "21GestaoRegistos.aspx.cs", Fonte Própria ..... | 30 |
| Código 14 - Evento "DropDownList1_SelectedIndexChanged" do ficheiro<br>"31GestaoUsers.aspx.cs", Fonte Própria .....                       | 31 |
| Código 15 - Evento " btnUpdate_Click " do ficheiro " 41EditarPerfil.aspx.cs ", Fonte Própria<br>.....                                     | 32 |

*“Quanto mais se sabe, menos se precisa.”*

Provérbio Australiano

## Introdução

Com a necessidade de evolução constante, as empresas e instituições têm necessidade de criar mecanismos para uma melhor organização e uma melhor gestão de recursos. Com isso, houve a necessidade do uso de novas tecnologias e, naturalmente, um surgimento de inúmeros *softwares* e aplicações aperfeiçoando os processos de trabalho dessas mesmas empresas.

Com este projeto pretende-se apresentar uma sugestão aplicacional para simplificar processos e métodos de trabalho. A aplicação proposta e implementada tem o intuito de acompanhar, com maior acuidade, o estágio clínico dos clientes. Pretende-se igualmente demonstrar que, com a centralização da informação, se otimiza a troca de informações entre técnicos de diversas valências.

A aplicação permite que um técnico de uma determinada especialidade faça o registo de cada sessão de acompanhamento, mantendo o historial clínico atualizado. Assim, a informação fica disponível de imediato para consulta por profissionais da mesma ou de outra especialidade.

Esta dissertação abrange vários capítulos e está organizada da seguinte forma:

- No Estado da Arte são apresentados o conceito de organização e o seu enquadramento, é consignado o contexto das aplicações *web* bem como as tecnologias e ferramentas para o seu desenvolvimento;
- Na Contextualização é apresentado o que motivou a escolha do tema, objetos de estudo, elaboração do projeto e objetivos pretendidos com o seu desenvolvimento;
- No Desenvolvimento são descritas, detalhadamente, todas as fases de criação da aplicação desde o seu esboço até à sua implementação;
- No último capítulo são apresentadas as conclusões e potenciação da aplicação;

# 1. Estado da Arte

## 1.1. A organização e o seu enquadramento

A terceira Revolução Industrial teve início na década de 1970 (Magnoli, 1997), na qual o panorama produtivo mundial se alterou, devido ao aparecimento de tecnologias microeletrónicas e da transmissão de informações sobre a automatização e robotização dos processos produtivos. Surgiram assim novos ramos industriais até então inexistentes, como a indústria de computadores e *software*, telecomunicações, química fina, robótica e biotecnologia, que são caracterizadas principalmente pela utilização de mão-de-obra qualificada ao contrário do que acontecia na indústria até aqui.

As pessoas deparavam-se constantemente com a vivência num mundo em que as grandes organizações haviam absorvido a sociedade. A população viu o seu poder de compra crescer rapidamente devido aos recursos abundantes, o que criou um vasto mercado que, por sua vez, levou a um consumo contínuo. (Perrow, 1991)

Ainda hoje, a maioria das pessoas convidadas a identificar os pontos fortes dos computadores tendem a pensar em tarefas computacionais como a rápida multiplicação de grandes números. No entanto, os computadores não são fundamentalmente trituradores de números. São processadores de símbolos. As mesmas tecnologias básicas podem ser utilizadas para armazenar, recuperar, organizar, transmitir e transformar algoritmos em qualquer tipo de informação que possa ser digitalizada – números, textos, vídeos, músicas, discursos, programas e desenhos de engenharia, para citar alguns. (Brynjolfsson & Hitt, Outono 2000)

As atividades diárias da maioria dos gestores, profissionais e trabalhadores do conhecimento envolvem outros tipos de pensamento. Como os computadores se tornam mais baratos e mais poderosos, o seu valor é limitado a menor capacidade computacional e maior capacidade dos gestores em inventar novos processos, procedimentos e estruturas organizacionais que aproveitam esta capacidade. Com a continuação do desenvolvimento de inovações complementares, as aplicações dos computadores irão expandir-se bem além da computação num futuro previsível. (Brynjolfsson & Hitt, Outono 2000)

As empresas que utilizam a tecnologia da informação para mudar a maneira como conduzem a sua atividade comercial dizem muitas vezes que o seu investimento em complementos de tecnologia da informação provoca alterações noutros aspetos da organização. Essas complementaridades têm implicações na compreensão do valor do investimento

computacional. Para serem bem-sucedidas, as empresas normalmente precisam de adotar os computadores como parte de um sistema ou "*Cluster*" reforçando mutuamente mudanças organizacionais. (Milgrom & Roberts, 1990)

## 1.2. Aplicação Web e o seu contexto

O conceito de "*Web 2.0*" surgiu numa sessão de *brainstorming* numa conferência entre O'Reilly e *MediaLive International*. Dale Dougherty, pioneiro do desenvolvimento *web* e O'Reilly VP, notaram que a *web* era mais importante que nunca, com novas aplicações e aparecimento de *websites* com uma surpreendente regularidade. Poderia o rápido desenvolvimento das *portocom* (.com) marcar um ponto de viragem para a *web*, de modo a que a chamada "*Web 2.0*" possa fazer sentido? Concordamos que sim, e assim surgiu a "Conferência *Web 2.0*". (O'Reilly, 2007)

Após ano e meio, o termo "*Web 2.0*" tinha-se expandido com mais de 9,5 milhões de citações no Google (definição publicada na *Web* em setembro de 2005; 135 milhões de citações de fevereiro de 2007). Mas ainda existe uma enorme discórdia sobre o que significa exatamente *Web 2.0*, com algumas pessoas a descrevê-la como um cliché de marketing sem sentido, e outras a aceitá-la como a nova sabedoria convencional. (O'Reilly, 2007)

O desenvolvimento *web* tem vindo a evoluir rapidamente ao longo dos anos. As características e funcionalidades que atualmente os *web developers* desenvolvem são exponencialmente mais complexas do que eram há alguns anos. À medida que as exigências atuais do mercado e de *software* comercial crescem, também deve crescer o poder das ferramentas e *frameworks* de desenvolvimento que os *developers* usam todos os dias. (Walther, Hoffman, & Scott, 2011)

Antes de existirem *websites* baseados em ASP.NET, eram criados em ASP clássico (*Active Server Pages*), que era mais uma linguagem de *script* do que uma verdadeira plataforma orientada para objetos e construção de aplicações *web*. (Breske, Gipson, & Salazar, 2012)

## 1.3. Desenvolvimento e evolução da plataforma ASP.NET

A primeira razão para que o desenvolvimento de aplicações *web* seja uma tarefa difícil deve-se à limitação do ambiente disponível no lado do cliente – para melhor ou para pior, a

única linguagem disponível nos *web browsers* é *JavaScript*, o que não é adequado para resolver muitos dos problemas que surgem aquando do desenvolvimento de aplicações *web* complexas. (Meijer, 2007)

Desde a primeira introdução, o ASP.NET sofreu um crescimento drástico em termos de *market timing*, facilidade de uso, potência, flexibilidade, escalabilidade. Alguns dos maiores *websites* hospedados na internet têm ASP.NET e *NET Framework* para aumentar a sua velocidade, potência e escalabilidade, envolvendo *Dell*, *MySpace* e Microsoft. (Walther, Hoffman, & Scott, 2011)

Com ASP.NET 1.0 e *Web Forms*, a Microsoft abordou as desvantagens inerentes ao ASP. As tecnologias ASP requeriam uma forte ligação entre a interface do utilizador e código da página. Exigia que as páginas *web* e toda a sua lógica fossem interpretadas a cada carregamento. Isto permitia pouca ou nenhuma separação de interesses e a reutilização de código era muito difícil. Com *ASP.NET Web Forms Framework*, a lógica pode ser facilmente colocada em conjuntos separados. Isto significava que com mais facilidade, experiência e forte liderança técnica uma solução poderia ser implementada utilizando separação de interesses. No entanto, a ligação entre os controlos de interface de utilizador e o *codebehind* é inevitável. (Meijer, 2007)

“*ASP.NET MVC é um framework para criar aplicações web robustas seguindo o padrão Model View Controller.*” (Galloway, Haack, Wilson, & Allen, 2011)

De acordo com os mesmos autores (Galloway, Haack, Wilson, & Allen, 2011), *ASP.NET MVC* separa a interface do utilizador de uma aplicação em três aspetos principais:

**Model:** conjuntos de classes que descrevem os dados que se estão a trabalhar e estabelece as regras para a forma como os dados podem ser alterados e manipulados;

**View:** define como a Interface do Utilizador (IU) da aplicação será exibida;

**Controller:** um conjunto de classes que lida com a comunicação do utilizador, o fluxo geral da aplicação e a lógica específica da aplicação.

#### 1.4. Migração

“*Migração de aplicativos é o processo de mover uma aplicação de um ambiente para outro.*” (Rouse, 2012)

ASP.NET *Web API* fornece uma forma de construir ou refazer aplicações ASP.NET *Web Forms* para formato MVC, movendo o código do *codebehind* para um controlador de *Web API*. Este processo também permite às aplicações ASP.NET promover o desenvolvimento de *JavaScript Assíncrono* e XML que pode ser usado para criar uma interface de utilizador mais ágil e melhorar a escalabilidade de uma aplicação, movendo a lógica no cliente e reduzindo a comunicação com o servidor. Isso é possível porque a *Web API* utiliza o protocolo HTTP e (por meio de codificação por convenção) executa automaticamente diversas tarefas de baixo nível. (Vogel, 2013)

### 1.5. Dependência

Para se perceber o que é a dependência e como se pode aplicá-la para aplicações MVC, é necessário falar de *Software Design Pattern*. Um *Software Design Pattern* é usado para formalizar a descrição de um problema e respetiva solução. (Galloway, Haack, Wilson, & Allen, 2011)

A injeção de dependência é uma técnica usada em linguagens de programação orientada a objetos. Promove a junção de componentes, passando dependências para um objeto, em vez de ter um objeto a instanciar as suas próprias dependências. (Carr, 2010)

### 1.6. Structured Query Language

Os computadores programáveis foram concebidos pela primeira vez durante a Segunda Guerra Mundial para aplicações militares. O *UNIVAC I*, a primeira máquina com intuito comercial, foi produzida em 1951. Várias gerações de linguagens de programação têm sido desenvolvidas desde então. Cada geração tem melhorado a produtividade dos programadores, automatizando tarefas mecânicas e permitindo ao programador concentrar-se nos conceitos de nível superior relacionados com a aplicação. Os primeiros programas foram escritos em código de máquina – números correspondentes às instruções que o programador queria armazenar na memória da máquina. (Kreines, 2000)

“Os futuros utilizadores de grandes bases de dados devem ser protegidos de ter de saber como os dados são organizados na máquina (a representação interna).” (Codd, 1970)

*Structured Query Language* (SQL) é a linguagem padrão usada para se comunicar com uma base de dados relacional. O protótipo foi desenvolvido originalmente pela IBM usando um protótipo do Doutor E. F. Codd<sup>1</sup> como modelo. Em 1979, não muito tempo depois da criação do protótipo da IBM, o primeiro produto SQL, o Oracle, foi lançado pela *Relational Software, Incorporated* (que mais tarde foi renomeada *Oracle Corporation*). Hoje é um dos líderes distintos em tecnologias de base de dados relacionais. (Stephens, Jones, & Plew, 2016)

A linguagem SQL e os Sistemas de Gestão de Base de Dados (SGBD) relacionais onde se baseiam constituem uma das mais importantes tecnologias na indústria da computação. Ao longo das últimas três décadas, SQL cresceu do seu primeiro uso comercial para um produto computacional e um segmento de mercado de serviços no valor de dezenas de bilhões de dólares por ano. SQL permanece ainda hoje como a linguagem de base de dados padrão. Centenas de produtos de base de dados suporta agora SQL, executando em sistemas a partir de *mainframes* para computadores pessoais. Uma base de dados baseada em SQL pode ser incorporada, quer no telemóvel ou PDA ou ainda no painel de instruções do automóvel. (Groff, Weinberg, & Opper, 2010)

Em contraste com a pluralidade de linguagens de programação processuais, hoje há apenas uma linguagem de acesso a dados amplamente utilizada: *Structured Query Language*. SQL é uma linguagem de acesso a dados não-processuais, uma vez que deixa ao Sistema de Gestão de Base de Dados (SGBD) a responsabilidade de determinar como os dados serão processados para decifrar uma consulta. O programador da aplicação não precisa de se preocupar com o caminho de acesso e de processamento dos dados necessários para produzir o resultado desejado. Assim como era mais fácil escrever programas de aplicações em linguagens de programação de nível superior em relação ao código de máquina de baixo nível e linguagem *assembly*, a linguagem SQL facilita o acesso aos dados em aplicações ou em rede *ad hoc*, de forma interativa. SQL permite ao programador da aplicação concentrar-se na lógica, ao invés de se preocupar com as questões da utilização de índices ou navegar através das cadeias de ponteiros para recuperar ou atualizar dados. (Kreines, 2000)

Existem vários “dialetos” de SQL em implementações de diferentes fornecedores, mas a maior parte da linguagem SQL é comum à maioria dos Sistemas de Gestão de Base de Dados comerciais agora no mercado. Existiram outras linguagens de acesso a dados, algumas

---

<sup>1</sup> Edgar Frank Codd – Engenheiro que, enquanto trabalhador da IBM, inventou o modelo relacional para a gestão de bases de dados

desenvolvidas em universidades e outras implementadas em produtos comerciais. Mas nenhuma linguagem de acesso a dados tem sido tão bem-sucedida ou amplamente implementada como a SQL. (Kreines, 2000)

SQL também difere de outras linguagens porque descreve *o que* o utilizador quer que o computador faça, ao invés de *como* o computador deverá fazer. (Em termos mais técnicos, SQL é uma linguagem mais declarativa ou descritiva em vez de processual). SQL não contém declarações *IF* para testar condições nem declarações *GO TO*, *DO* ou *FOR* para controlo de fluxo. Em vez disso, as declarações SQL descrevem como uma recolha de dados tem de ser organizada, ou que dados são necessários recuperar ou adicionar à base de dados. A sequência dos passos para elaborar tais tarefas é determinada pelo Sistema de Gestão de Base de Dados. (Groff, Weinberg, & Opperl, 2010)

Todas as empresas modernas têm dados, o que requer algum método organizado ou mecanismo para manter e recuperar esses dados. Quando os dados são mantidos dentro de uma base de dados, este mecanismo é referido como SGBD. Os Sistemas de Gestão de Base de Dados têm sido amplamente utilizados ao longo dos anos, muitos dos quais começaram como sistemas de arquivos simples num *mainframe*. Com as tecnologias de hoje, o uso aceitável de sistemas de gestão de base de dados começou a fluir noutras direções, impulsionado pela procura de empresas em crescimento, aumento do volume de dados corporativos, e, claro, as tecnologias da Internet. (Stephens, Jones, & Plew, 2016)

A onda moderna de gestão da informação é realizada principalmente através da utilização de um Sistema de Gestão de Base de Dados Relacional (SGBDR), derivado dos SGBD tradicionais. As bases de dados modernas combinadas com tecnologias de cliente/servidor e tecnologias *web* são combinações típicas utilizadas pelas empresas atuais para gerir com sucesso os seus dados e manterem-se competitivas nos seus mercados. A tendência para muitas empresas é passar de um ambiente cliente/servidor para o *Web*, onde a localização não é uma restrição quando os utilizadores precisam de ter acesso a dados importantes. (Stephens, Jones, & Plew, 2016)

SQL emergiu como o idioma padrão para uso em bases de dados relacionais, sendo uma linguagem poderosa e de fácil aprendizagem. (Groff, Weinberg, & Opperl, 2010)

## 1.7. Tecnologias e Ferramentas para o Desenvolvimento Web

### 1.7.1. WCF

*Windows Communication Foundation* (WCF) é uma tecnologia de sistemas de próxima geração distribuída pela Microsoft. Fornece uma única *Application Programming Interface* (API) e uma única infraestrutura onde se pode construir, implantar e operar aplicações distribuídas com WCF. Tem as melhores características existentes das tecnologias distribuídas em .NET e coloca-as num único modelo de programação. (Selly, Troelsen, & Barnaby, 2006)

### 1.7.2. Entity Framework

“O seu lançamento foi em julho de 2008 fazendo parte do *Visual Studio 2008 Service Pack 1* e *.NET 3.5 Service Pack 1*, dois anos após o anúncio feito pela Microsoft na *Conferência TechEd 2006*.” (Lerman, 2010)

Um benefício fundamental do *Entity Framework* é fazer com que o programador deixe de se preocupar com a estrutura da sua base de dados. Todo o seu acesso e armazenamento de dados é feito de acordo com um modelo conceptual de dados que reflete os seus próprios objetos. (Lerman, 2010)

### 1.7.3. LINQ

LINQ só existe nas linguagens *C#* e *Visual Basic*. Com o *Entity Framework* não há outra forma de expressar consultas, que não só lhe permite utilizar outras linguagens, mas também oferece benefícios adicionais de onde pode tirar vantagem, se necessário. (Lerman, 2010)

### 1.7.4. IIS

O *Internet Information Services* (IIS) fornece os serviços essenciais para a hospedagem de servidores *Web*, aplicações *Web* e serviços *Microsoft Windows SharePoint*. Pode usar-se o IIS para publicar informações em intranets, extranets e Internet. Como os *websites* atualmente usam recursos relacionados, tais como filtros ISAPI, ASP, ASP.NET, CGI e NET Framework, o IIS junta esses recursos como parte de uma oferta abrangente. (Stanek, 2008)

## **2. Contextualização**

### **2.1. Motivação e Relevância**

Este trabalho vem no seguimento de um acontecimento importante na minha vida que foi o nascimento do meu segundo filho. O pequeno Duarte nasceu há dezoito meses com Síndrome de Down<sup>2</sup>. Neste curto espaço de tempo não nos poupámos à procura de apoios ao desenvolvimento do Duarte. Ao visitar algumas instituições notei que, apesar da organização existente, a informação encontra-se um pouco dispersa.

### **2.2. Objeto de estudo**

Cada empresa ou instituição usa o seu próprio sistema de gestão, disposição e manutenção de informação crucial, sendo necessários numerosos documentos que provem a evolução clínica dos seus clientes. A informação poderá, amiúde, estar incompleta ou conter falhas cruciais, não cumprindo as normas do sistema implementado.

Este documento apresenta um plano de desenvolvimento de uma Aplicação Web em ASP.NET, pertencendo à Área Ciências da Programação. Esta aplicação parte de pressupostos de organização e gestão centralizada da informação, simplificando a comunicação multidisciplinar.

### **2.3. Objetivos**

O acesso à aplicação é feito com controlo de sessão através da criação de utilizadores com diferentes credenciais. Dependendo das suas funções na instituição, cada utilizador tem um determinado nível de acesso e permissões.

Pretende-se, com esta aplicação desenvolver um conjunto de procedimentos que permitam simplificar o acesso à informação, gestão e acompanhamento do desenvolvimento de pessoas com distintos diagnósticos.

---

<sup>2</sup> A narração formal começou em 1866, quando um médico chamado John Langdon Down publicou um ensaio em Inglaterra em que descreveu um conjunto de crianças com características comuns que eram distintas de outras crianças com retardo mental. Down foi diretor de um lar para crianças com retardo mental em Surrey, Inglaterra, quando fez a primeira distinção entre as crianças que eram cretinos (que viria a ser diagnosticado hipotireoidismo) e que ele se referia como "mongoloides". (Leshin, 2003)

### 3. Desenvolvimento

Uma vez selecionada uma área científica, efetuado um estudo e definidos objetivos pretendidos, existem três fases para a construção da aplicação – esboço, modelagem, implementação e teste.

O esboço é o ponto de partida no desenvolvimento de qualquer projeto, contendo detalhes suficientes para permitir a sua execução até se atingir a versão final pretendida. Na atividade de programação coloca-se em prática a montagem da aplicação. Usa-se uma metodologia de trabalho que visa o desenvolvimento sistemático e gradual permitindo a resolução de problemas. A implementação e teste permitem saber se todo o plano e objetivos definidos para a aplicação estão a ser cumpridos conforme esperado.

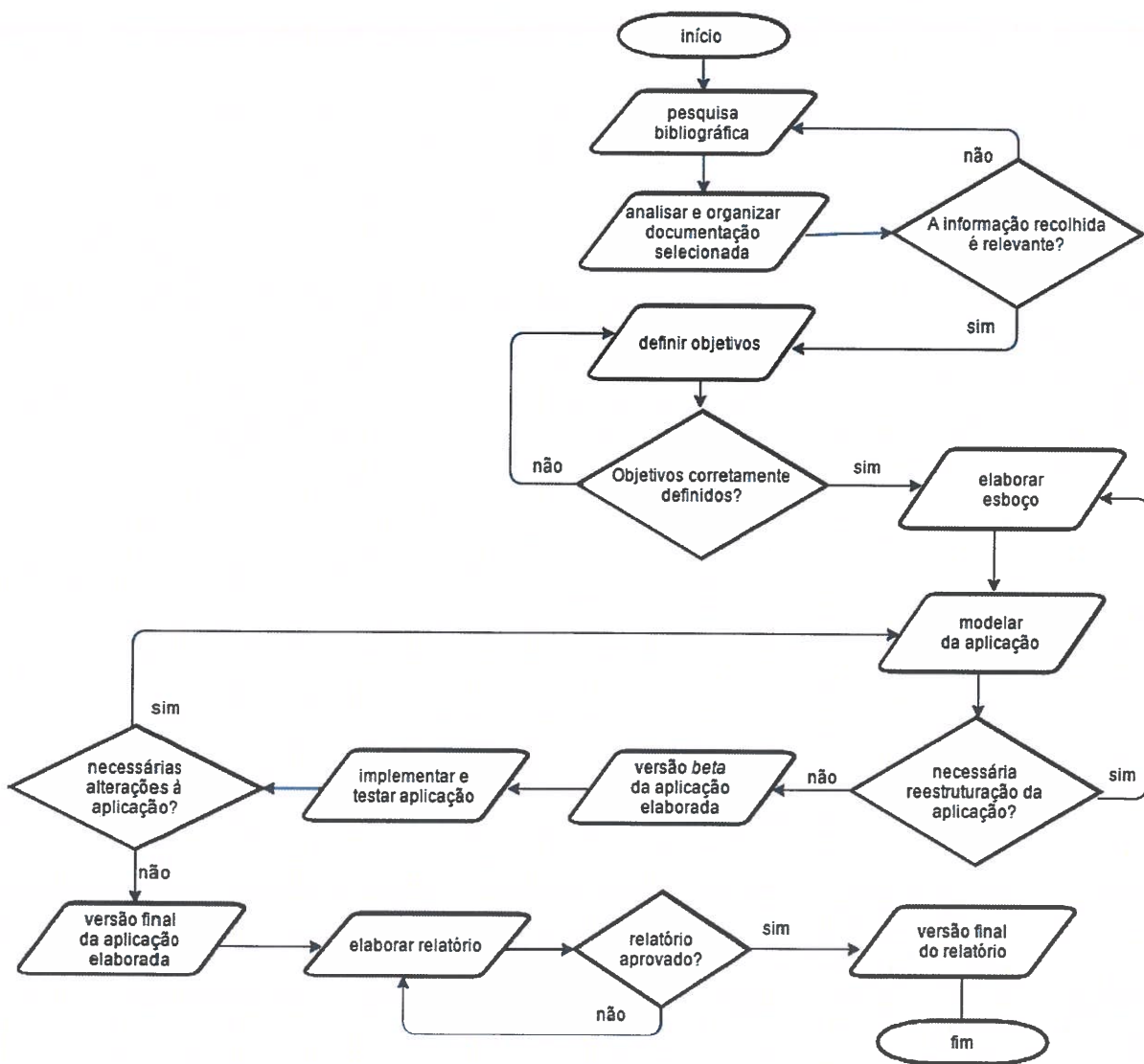


Figura 1 - Fluxograma do Projeto Global, Fonte Própria

### 3.1. Esboço

Esta fase consistiu em estruturar uma aplicação que possibilita o acesso e atualização de informação clínica centralizada, com o desígnio de simplificar processos de administração e tornar nula, ou muito baixa, a perda de informação. Para tal, estruturei uma Aplicação Web em ASP.NET usando a linguagem de programação C#, criando, com a *Framework .NET* páginas dinâmicas em que o código é executado no lado do servidor não interferindo com a interface gráfica. No que concerne ao armazenamento, recuperação e manipulação de dados em Base de Dados, a linguagem usada é SQL.

### 3.2. Modelagem

Delineados objetivos para a aplicação, é nesta fase que se efetua a elaboração e programação da aplicação.

#### 3.2.1. Base de Dados

A base de dados para esta aplicação foi estruturada com SQL Server 2014 Management Studio, sendo esquematizada da seguinte forma:

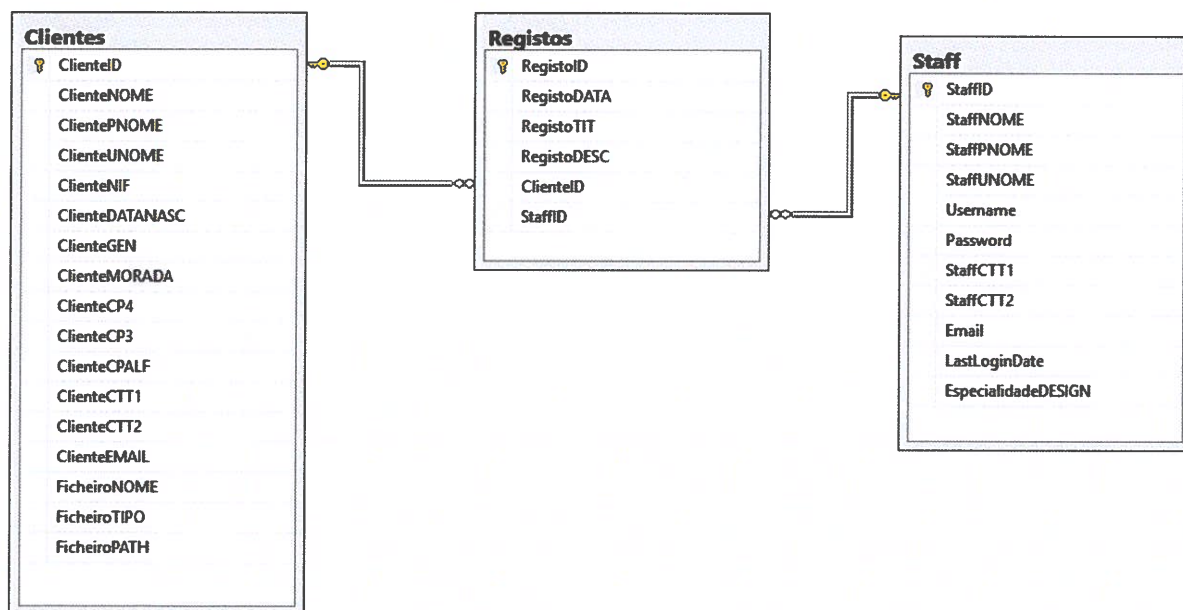


Figura 2 - Diagrama da base de dados, Fonte Própria

```

1 CREATE TABLE [dbo].[Clientes] (
2     [ClienteID] INT IDENTITY (1, 1) NOT NULL,
3     [ClienteNOME] NVARCHAR (60) NULL,
4     [ClientePNOME] NVARCHAR (20) NOT NULL,
5     [ClienteUNOME] NVARCHAR (20) NOT NULL,
6     [ClienteNIF] NVARCHAR (9) NOT NULL,
7     [ClienteDATANASC] DATE NOT NULL,
8     [ClienteGEN] NVARCHAR (20) NOT NULL,
9     [ClienteMORADA] NVARCHAR (200) NOT NULL,
10    [ClienteCP4] NVARCHAR (4) NOT NULL,
11    [ClienteCP3] NVARCHAR (3) NOT NULL,
12    [ClienteCPALF] NVARCHAR (50) NOT NULL,
13    [ClienteCTT1] INT NULL,
14    [ClienteCTT2] INT NULL,
15    [ClienteEMAIL] NVARCHAR (50) NULL,
16    [FicheiroNOME] NVARCHAR (80) NULL,
17    [FicheiroTIPO] NVARCHAR (80) NULL,
18    [FicheiroPATH] NVARCHAR (255) NULL,
19    PRIMARY KEY CLUSTERED ([ClienteID] ASC)
20 );

```

*Código 1 - Sintaxe para criação da tabela "Clientes", Fonte Própria*

```

1 CREATE TABLE [dbo].[Staff] (
2     [StaffID] INT IDENTITY (1, 1) NOT NULL,
3     [StaffNOME] NVARCHAR (60) NOT NULL,
4     [StaffPNOME] NVARCHAR (20) NOT NULL,
5     [StaffUNOME] NVARCHAR (20) NOT NULL,
6     [Username] NVARCHAR (25) NOT NULL,
7     [Password] NVARCHAR (25) NOT NULL,
8     [StaffCTT1] INT NULL,
9     [StaffCTT2] INT NULL,
10    [Email] NVARCHAR (50) NULL,
11    [LastLoginDate] DATETIME NULL,
12    [EspecialidadeDESIGN] NVARCHAR (25) NOT NULL,
13    PRIMARY KEY CLUSTERED ([StaffID] ASC)
14 );

```

*Código 2 - Sintaxe para criação da tabela "Staff", Fonte Própria*

```

1 CREATE TABLE [dbo].[Registos] (
2     [RegistoID] INT IDENTITY (1, 1) NOT NULL,
3     [RegistoDATA] NVARCHAR (20) NOT NULL,
4     [RegistoTIT] NVARCHAR (50) NULL,
5     [RegistoDESC] TEXT NULL,
6     [ClienteID] INT NOT NULL,
7     [StaffID] INT NOT NULL,
8     PRIMARY KEY CLUSTERED ([RegistoID] ASC),
9     CONSTRAINT [FK_Registos_Staff] FOREIGN KEY ([StaffID]) REFERENCES [dbo].[Staff] ([StaffID]),
10    CONSTRAINT [FK_Registos_Clientes] FOREIGN KEY ([ClienteID]) REFERENCES [dbo].[Clientes] ([ClienteID])
11 );

```

*Código 3 - Sintaxe para criação da tabela "Registos", Fonte Própria*

### 3.2.2. Desenvolvimento Front-End

Como já referido, as aplicações criadas com a tecnologia ASP.NET proporcionam um ambiente escalável do lado do servidor, melhorando a gestão de código e a interação com o ambiente do lado do cliente. Esta particularidade possibilita a separação do código em dois ficheiros diferenciados.

Para o desenvolvimento da interface gráfica são criados ficheiros do tipo *.aspx* que englobam HTML e ASP.NET:

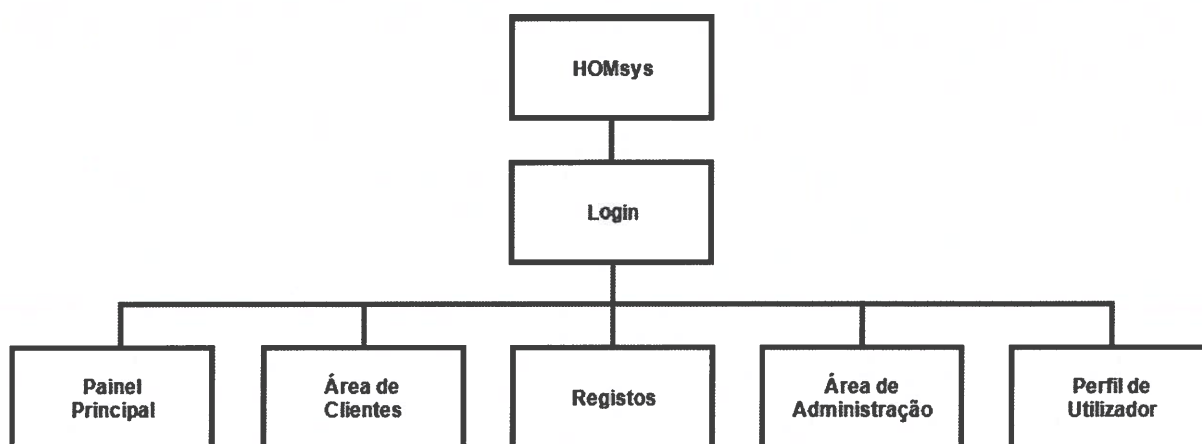


*Figura 3 - alguns dos ficheiros .aspx da minha aplicação, Fonte Própria*

Estes ficheiros invocam outros ficheiros que contêm código que controla todas as funcionalidades da interface através de diretivas específicas. No seguinte excerto de código a página “*Login.aspx*” invoca o ficheiro “*Login.aspx.cs*” que contém os métodos para o seu funcionamento:

```
1. <%@ Page Title="Login" Language="C#" AutoEventWireup="true"
   CodeFile="Login.aspx.cs" Inherits="Login" %>
```

O sistema de gestão HOMsys tem como objetivo centralizar informação com uma maior eficiência, proporcionando uma rápida pesquisa e atualização da informação com maior acuidade. Resumindo, é uma ferramenta que auxilia na gestão e organização de processos clínicos. O Administrador tem acesso a todo o conteúdo da aplicação e cada um dos outros utilizadores criados tem acesso a conteúdo disponibilizado para o perfil a que pertence.



*Figura 4 - Visão Geral do Sistema de Gestão HOMsys, Fonte Própria*

- **Página “Login.aspx”**

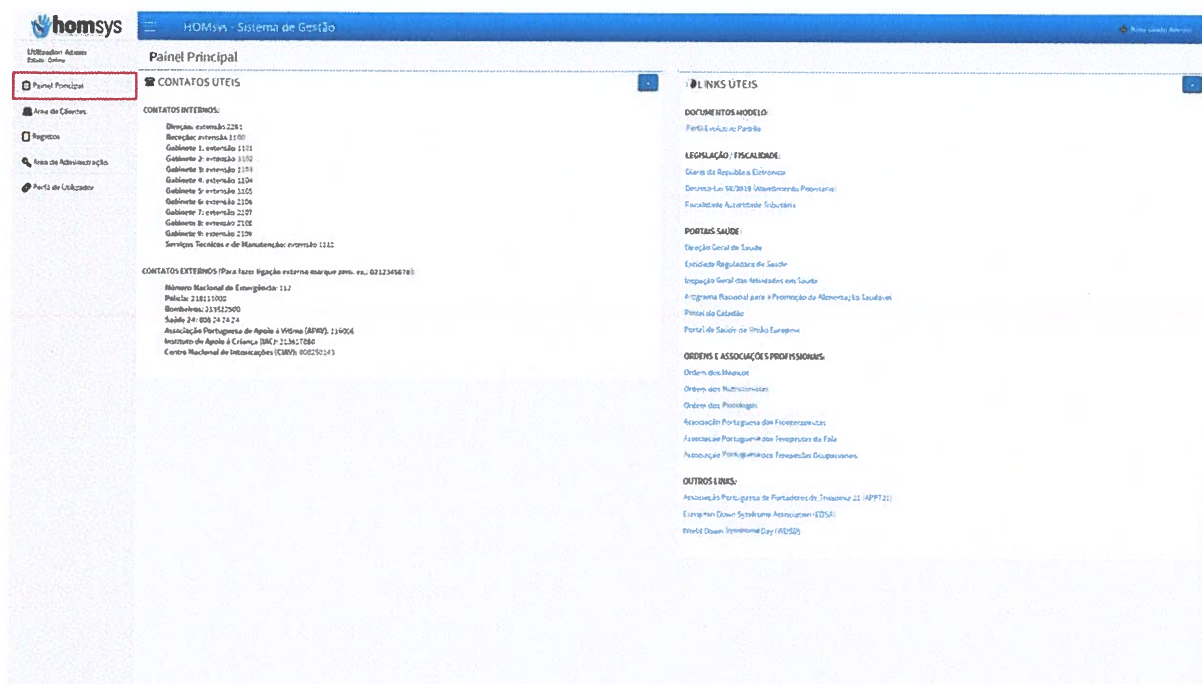
A página de autenticação do sistema surge quando o utilizador executa a aplicação. O utilizador deverá introduzir credenciais válidas para aceder ao seu conteúdo. O nome do utilizador e respetiva palavra-passe são criados pelo administrador na área de administração da aplicação.



*Figura 5 - Página "Login", Fonte Própria*

- **Página “01PainelPrincipal.aspx”**

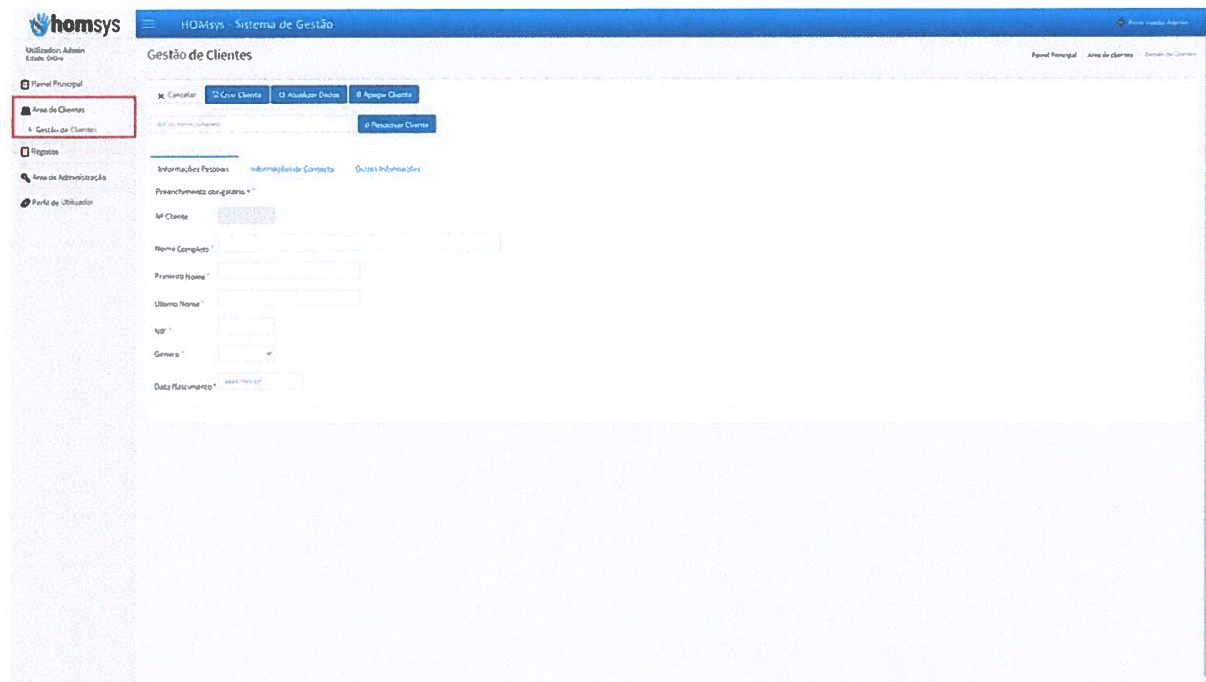
Esta é a página inicial da aplicação após validado o acesso à aplicação. Neste painel, acessível a todos os utilizadores, são apresentados contactos úteis, sejam eles internos ou externos, bem como links para páginas relevantes na internet.



**Figura 6 - Página "Painel Principal", Fonte Própria**

- **Página “11GestaoClientes.aspx”**

Esta página contém informações dos clientes, quer sejam pessoais, de contacto ou outras informações relativas ao seu perfil evolutivo. Permite ao utilizador com perfil “Recepção” fazer a criação de novos clientes e consultar as respetivas informações pessoais e de contacto, estando vedada a visualização de dados clínicos. O técnico, ou técnicos, que acompanham o cliente podem consultar os seus dados e atualizar a evolução do seu estado clínico através do *upload* do Perfil Evolutivo.



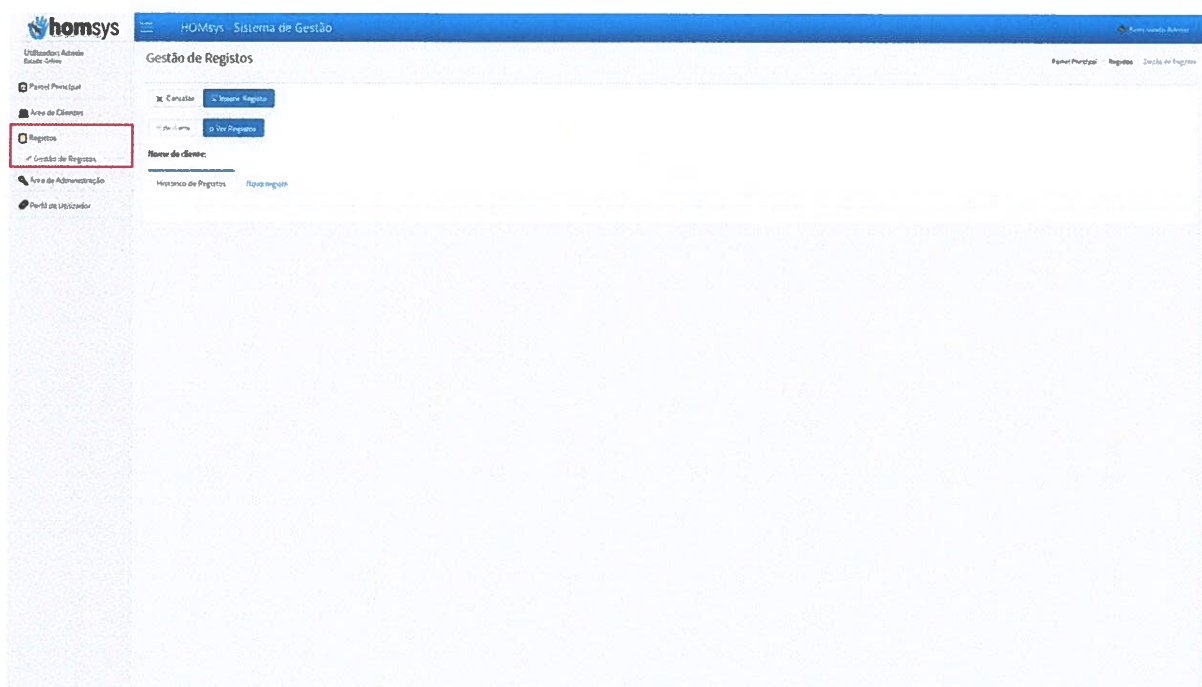
**Figura 7 - Página "Gestão de Clientes", Fonte Própria**

- **Página “21GestaoRegistos.aspx”**

Esta página serve para registo de informações clínicas respeitantes a cada cliente. É criado um registo a cada consulta onde os técnicos descrevem, no campo respetivo, todos os dados que considerem pertinentes.

O seu conteúdo é restrito apenas aos técnicos que fazem o acompanhamento dos clientes, simplificando a comunicação e transferência de informação entre os diferentes profissionais, mantendo-se assim atualizado o histórico clínico.

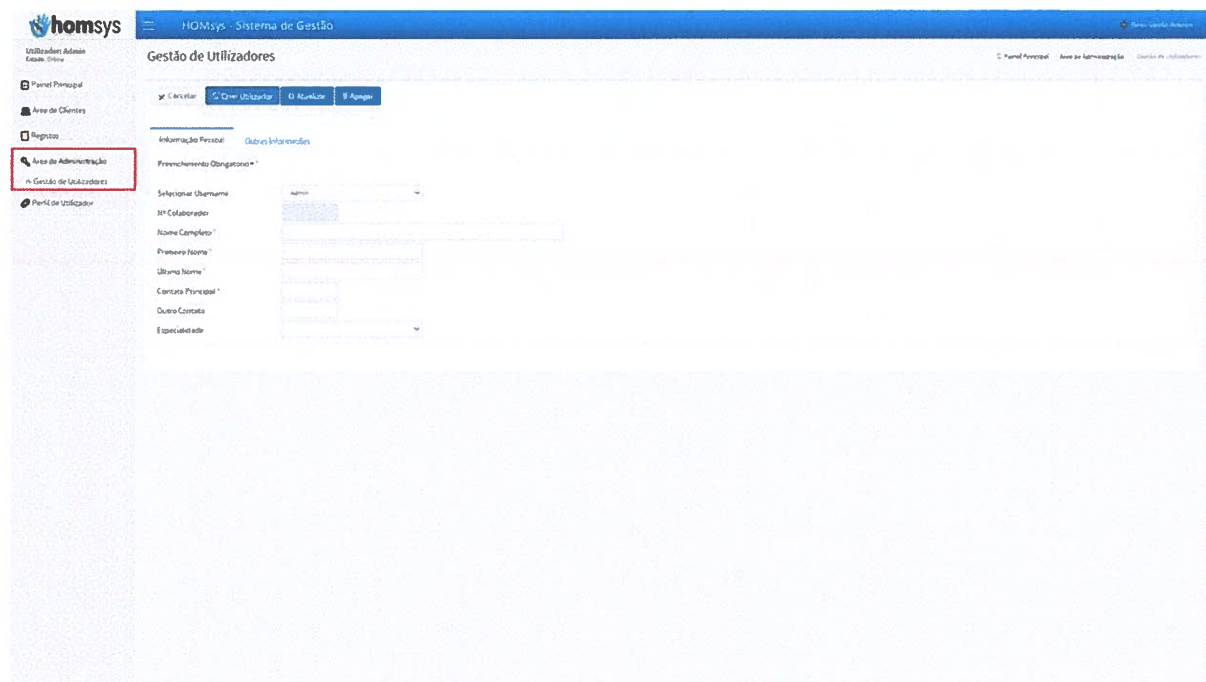
Em futuras consultas todo o histórico é passível de ser visualizado, facilitando a identificação do estágio evolutivo graças à constante atualização da informação. O histórico é apresentado numa *GridView* após pesquisa por número de cliente, apresentando-se listado cronologicamente, desde o registo mais recente ao mais antigo.



**Figura 8 - Página "Gestão de Registos", Fonte Própria**

- **Página “31GestaoUsers.aspx”**

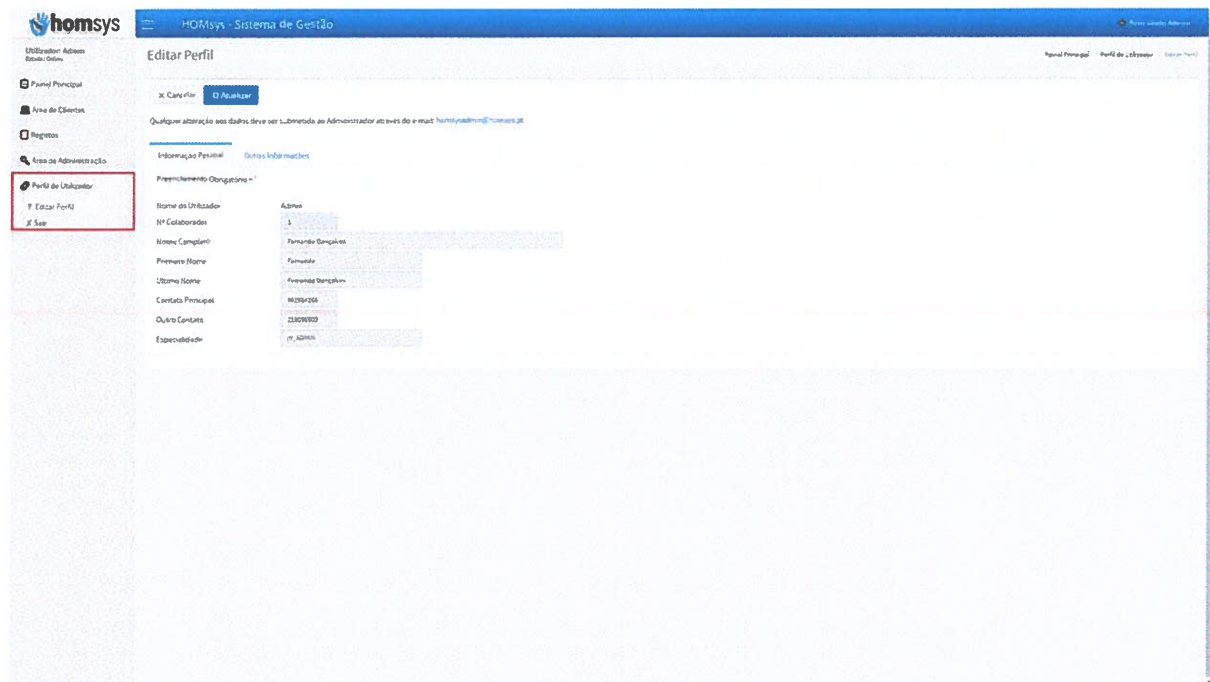
Apenas acessível ao Administrador, esta página serve para fazer gestão dos utilizadores. Existem três perfis de utilizador distintos para a aplicação: Administrador, Recepção e Técnicos. Apenas o Administrador pode aceder a esta página sendo-lhe permitido criar, apagar e alterar dados de todos os utilizadores.



**Figura 9 - Página "Gestão de Utilizadores", Fonte Própria**

- **Página “41EditarPerfil.aspx”**

Este módulo pode ser acessado por todos os utilizadores e permite a consulta dos dados pessoais, e é permitida, unicamente, a alteração da password de acesso à aplicação. Qualquer alteração de outros dados deverá ser solicitada por *e-mail* ao Administrador.



**Figura 10 - Página "Editar Perfil", Fonte Própria**

### 3.2.3. Desenvolvimento Back-End

A ordem na qual os métodos virtuais (*OnLoad*) e manipuladores de eventos (*Page\_Load*) são invocados é definida pelo chamado “ciclo de vida da página”. Esta é a única forma como o tempo de execução do ASP.NET processa uma solicitação de entrada. São exemplo disso as etapas de *Init*, *Load*, *Render*.

Tabela 1 - Ciclo de vida da página, Fonte Microsoft Developer Network

|                | Page methods                       | Page events       | Control methods and events        |
|----------------|------------------------------------|-------------------|-----------------------------------|
| Start          | Construct                          |                   |                                   |
|                | ProcessRequest                     |                   |                                   |
|                | InitializeCulture                  |                   |                                   |
|                | DeterminePostBackMode              |                   |                                   |
|                | OnPreInit                          | PreInit           | Init                              |
|                | OnInit                             | Init              |                                   |
|                | TrackViewState                     |                   |                                   |
|                | OnInitComplete                     | InitComplete      |                                   |
| Load           | LoadPageStateFromPersistenceMedium |                   | LoadViewState                     |
|                | LoadViewState                      |                   | IPostBackDataHandler.LoadPostData |
|                | ProcessPostData                    |                   |                                   |
|                | OnPreLoad                          | PreLoad           | Load                              |
|                | OnLoad                             | Load              |                                   |
| Event handling | RaisePostBackEvent                 |                   | Control-changed events            |
| Validation     | Validate                           |                   |                                   |
| PreRendering   | OnLoadComplete                     | LoadComplete      |                                   |
|                | OnPreRender                        | PreRender         | PreRender                         |
|                | OnPreRenderComplete                | PreRenderComplete | Data binding events               |
|                | SaveViewState                      |                   | SaveViewState                     |
|                | SavePageStateToPersistenceMedium   |                   |                                   |
| Rendering      | OnSaveStateComplete                | SaveStateComplete |                                   |
|                | RenderControl                      |                   | Render                            |
|                | Render                             |                   |                                   |
| Unload         | RenderChildren                     |                   |                                   |
|                | OnUnload                           |                   | Unload                            |
|                | Dispose                            |                   |                                   |

Como mencionado anteriormente, ao ser criado o ficheiro *.aspx*, é-lhe acoplado um outro ficheiro com extensão *aspx.cs*, normalmente chamado *codebehind*. Estes ficheiros têm, por norma, o mesmo nome que os primeiros, operam do lado do servidor.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Data;
using System.Data.SqlClient;
using System.Configuration;
using System.Web.Security;
using System.Web.UI.HtmlControls;
using System.Collections;

public partial class Login : System.Web.UI.Page
{

} // classe principal - Login : System.Web.UI.Page
```

**Código 4 - Classe Principal do ficheiro "Login.aspx.cs", Fonte Própria**

Através da classe não hereditária "*SqlConnection*" faz-se a ligação à Base de Dados SQL. Nela é inicializada uma nova instância contendo o caminho onde se encontra a BD, inicializa, igualmente, uma instância da classe "*SqlConnection*" usando

```
SqlConnection con = new SqlConnection("Data
Source=(LocalDB)\v11.0;AttachDbFilename=D:\\[ISTEC]\\[Disciplinas]\\_PROJETO_GLOBAL
\\homsys\\development\\HOMsys\\App_Data\\HOMsys.mdf;Integrated Security=True");
```

- Ficheiro “Login.aspx.cs”

Para fazer a validação do utilizador, no método *Page\_Load*, fiz uma declaração *if* para confirmar se o utilizador foi autenticado e em caso afirmativo reencaminha-o para a página “01PainelPrincipal.aspx”.

A autenticação é feita através do método de autenticação “*Login1\_Authenticate*” no qual é transposto o *Username* e *Password* do controlo “*Login1*” para uma função de autenticação contendo uma variável booleana “*blnresult*” que irá fazer a sua verificação na Base de Dados. Se o resultado dessa verificação for verdadeiro autentica o utilizador, caso contrário, é apresentada uma mensagem de erro.

```
protected void Page_Load(object sender, EventArgs e)
{
    if ((Session["Check"] != null) && (Convert.ToBoolean(Session["Check"]) ==
true))
    {
        if (User.Identity.IsAuthenticated)
            Response.Redirect("01PainelPrincipal.aspx");
    }
}

protected void Login1_Authenticate(object sender, AuthenticateEventArgs e)
{
    Boolean blnresult;
    blnresult = false;
    blnresult = Authentication(Login1.UserName, Login1.Password);

    if (blnresult == true)
    {
        e.Authenticated = true;
        Session["Check"] = true;
    }
    else
        e.Authenticated = false;
}
```

**Código 5 - Codebehind da Página de Login, Fonte Própria**

- Ficheiro “11GestaoClientes.aspx.cs”

Na página de gestão de clientes são informações dos clientes, sendo elas pessoais, de contacto ou outras informações clínicas.

A criação de cliente e atualização da sua ficha é feita somente pelo utilizador com perfil de receção, pelo que, apesar de acesso ao conteúdo da página a outros utilizadores, os botões para criar cliente, atualizar dados e apagar cliente não estão disponíveis.

Esta página contém, como referido, informações clínicas que apenas poderão ser acedidas por técnicos que fazem o acompanhamento dos clientes.

```
protected void Page_Load(object sender, EventArgs e)
{
    if (Page.User.Identity.Name == "Rececao" || Page.User.Identity.Name ==
"Admin")
    {
        btnSave.Visible = true;
        btnUpdate.Visible = true;
        btnDelete.Visible = true;
    }
    else
    {
        btnSave.Visible = false;
        btnUpdate.Visible = false;
        btnDelete.Visible = false;
    }

    if (Page.User.Identity.Name == "Rececao")
    {
        OItab.Visible = false;
    }
}
```

*Código 6 - Método "Page\_Load" do ficheiro "11GestaoClientes.aspx.cs", Fonte Própria*

O botão para adicionar cliente responde ao evento “*btnSave\_Click*” a ele associado. Primariamente é instanciada a *ConnectionString* criada e faz-se a abertura da mesma para se aceder à BD, procedimento que é feito para todos os botões da aplicação. De seguida cria-se o comando SQL “*INSERT INTO*” para criar novo cliente, fecha-se a conexão à Base de Dados e é apresentada uma mensagem indicando que o cliente foi criado com sucesso.

```
protected void btnSave_Click(object sender, EventArgs e)
{
    con = new
SqlConnection(System.Configuration.ConfigurationManager.ConnectionStrings["con"
].ConnectionString);
    con.Open();
    cmd = new SqlCommand("INSERT INTO Clientes(ClienteNOME, ClientePNOME,
ClienteUNOME, ClienteNIF, ClienteDATANASC, ClienteGEN, ClienteMORADA,
ClienteCP4, ClienteCP3, ClienteCPALF, ClienteCTT1, ClienteCTT2, ClienteEMAIL)
VALUES('" + txtBoxClienteNome.Text + "',
'" + txtBoxClientePNOME.Text + "',
'" + txtBoxClienteUNOME.Text + "',
'" + txtBoxNIF.Text + "',
'" + txtBoxDNasc.Text + "',
'" + ddlGenero.Text + "',
'" + txtBoxMorada.Text + "',
'" + txtCP4.Text + "',
'" + txtCP3.Text + "',
'" + txtLocalidade.Text + "',
'" + txtCttPrincipal.Text + "',
'" + txtCttOutro.Text + "',
'" + txtEmail.Text + "')", con);
    cmd.ExecuteNonQuery();
    con.Close();
    lblInfo.Text = "Cliente criado com sucesso";
    txtBoxSearch.Text = "";
}
```

*Código 7 - Evento "btnSave\_Click" do ficheiro "11GestaoClientes.aspx.cs", Fonte Própria*

Tal como acontece com o botão para adicionar cliente, o botão para atualizar dados de cliente responde a um evento com o nome “*btnUpdate\_Click*”. É feita a atualização dos dados da ficha do cliente através do comando SQL “*UPDATE*”. Sempre que os dados do cliente são atualizados é apresentada uma mensagem com tal informação.

```
protected void btnUpdate_Click(object sender, EventArgs e)
{
    con = new
SqlConnection(System.Configuration.ConfigurationManager.ConnectionStrings["con"
].ConnectionString);
con.Open();
cmd = new SqlCommand("UPDATE Clientes SET
    ClienteNOME='" + txtBoxClienteNome.Text + "',
    ClientePNOME='" + txtBoxClientePNOME.Text + "',
    ClienteUNOME='" + txtBoxClienteUNOME.Text + "',
    ClienteNIF='" + txtBoxNIF.Text + "',
    ClienteDATANASC='" + txtBoxDNasc.Text + "',
    ClienteGEN='" + ddlGenero.Text + "',
    ClienteMORADA='" + txtBoxMorada.Text + "',
    ClienteCP4='" + txtCP4.Text + "',
    ClienteCP3='" + txtCP3.Text + "',
    ClienteCPALF='" + txtLocalidade.Text + "',
    ClienteCTT1='" + txtCttPrincipal.Text + "',
    ClienteCTT2='" + txtCttOutro.Text + "',
    ClienteEMAIL='" + txtEmail.Text + "'
    WHERE ClienteID='" + txtBoxClienteID.Text + "'", con);
cmd.ExecuteNonQuery();
con.Close();
lblInfo.Text = "Dados do Cliente atualizados com sucesso ";
txtBoxSearch.Text = "";
}
```

*Código 8 – Evento "btnUpdate\_Click" do ficheiro "11GestaoClientes.aspx.cs", Fonte Própria*

A construção do evento do botão para apagar cliente é muito semelhante à dos botões adicionar e atualizar cliente. No entanto, usa-se o comando SQL usado “*DELETE*”. É despoletada igualmente uma mensagem que confirma a eliminação do cliente pretendido.

```
protected void btnDelete_Click(object sender, EventArgs e)
{
    con = new
SqlConnection(System.Configuration.ConfigurationManager.ConnectionStrings["con"
].ConnectionString); // instância da ConnectionString
    con.Open();
    cmd = new SqlCommand("DELETE FROM Clientes WHERE ClienteID='" +
        txtBoxClienteID.Text + "'", con);
    cmd.ExecuteNonQuery();
    con.Close();

    lblInfo.Text = "Cliente eliminado com sucesso";
    txtBoxSearch.Text = "";
    txtBoxClienteID.Text = "";
    txtBoxClienteNome.Text = "";
    txtBoxClientePNOME.Text = "";
    txtBoxClienteUNOME.Text = "";
    txtBoxNIF.Text = "";
    ddlGenero.Text = "";
    txtBoxDNasc.Text = "";
    txtBoxMorada.Text = "";
    txtCP4.Text = "";
    txtCP3.Text = "";
    txtLocalidade.Text = "";
    txtCttPrincipal.Text = "";
    txtCttOutro.Text = "";
    txtEmail.Text = "";
    fich.Text = "";
}
```

*Código 9 - Evento "btnDelete\_Click" do ficheiro "11GestaoClientes.aspx.cs", Fonte Própria*

A caixa de texto onde se insere o NIF ou nome de cliente pretendido está associada ao evento “*txtBoxSearch\_TextChanged*” na qual são apresentados os dados do cliente através do comando SQL “*SELECT*”. Através da declaração *if, else* é feita a verificação se o cliente foi encontrado na BD mostrando os seus dados..

```
protected void txtBoxSearch_TextChanged(object sender, EventArgs e)
{
    con = new
SqlConnection(System.Configuration.ConfigurationManager.ConnectionStrings["con"
].ConnectionString);
    SqlDataAdapter da = new SqlDataAdapter("SELECT * FROM Clientes WHERE
ClienteNOME='" + txtBoxSearch.Text + "' OR ClienteNIF='" + txtBoxSearch.Text +
"", con);
    DataTable dt = new DataTable();
    da.Fill(dt);
    txtBoxSearch.Text = "";
    if (dt.Rows.Count > 0)
    {
        txtBoxClienteID.Text = dt.Rows[0][0].ToString();
        txtBoxClienteNome.Text = dt.Rows[0][1].ToString();
        txtBoxClientePNOME.Text = dt.Rows[0][2].ToString();
        txtBoxClienteUNOME.Text = dt.Rows[0][3].ToString();
        txtBoxNIF.Text = dt.Rows[0][4].ToString();
        DateTime data = DateTime.Parse(dt.Rows[0][5].ToString());
        txtBoxDNasc.Text = data.Date.ToString("dd/MM/yyyy");
        ddlGenero.Text = dt.Rows[0][6].ToString();
        txtBoxMorada.Text = dt.Rows[0][7].ToString();
        txtCP4.Text = dt.Rows[0][8].ToString();
        txtCP3.Text = dt.Rows[0][9].ToString();
        txtLocalidade.Text = dt.Rows[0][10].ToString();
        txtCttPrincipal.Text = dt.Rows[0][11].ToString();
        txtCttOutro.Text = dt.Rows[0][12].ToString();
        txtEmail.Text = dt.Rows[0][13].ToString();
        fich.Text = dt.Rows[0][14].ToString();
        fich.NavigateUrl = dt.Rows[0][16].ToString();
        lblInfo.Text = "";
    }
    else
    {
        txtBoxClienteID.Text = "";
        txtBoxClienteNome.Text = "";
        txtBoxClientePNOME.Text = "";
        txtBoxClienteUNOME.Text = "";
        txtBoxNIF.Text = "";
        txtBoxDNasc.Text = "";
        ddlGenero.Text = "";
        txtBoxMorada.Text = "";
        txtCP4.Text = "";
        txtCP3.Text = "";
        txtLocalidade.Text = "";
        txtCttPrincipal.Text = "";
        txtCttOutro.Text = "";
        txtEmail.Text = "";
        fich.Text = "";
        fich.NavigateUrl = "";
        lblInfo.Text = "Cliente não encontrado";
    }
}
}
```

**Código 10 - Método "txtBoxSearch\_TextChanged" do ficheiro "11GestaoClientes.aspx.cs", Fonte Própria**

Por fim, temos o botão para se efetuar o *Upload* e um ficheiro que fica guardado na pasta “ficheiros” da aplicação. A atualização é feita com o comando SQL “*UPDATE*”, mas neste caso, os argumentos são referenciados através de *parameters* permitindo que membros de funções, métodos, propriedades, indexadores, operadores e construtores alterar o valor dos parâmetros.

```
protected void btnUpload_Click(object sender, EventArgs e)
{
    HttpPostedFile file = FileUpload.PostedFile;
    if (file.FileName.Length > 0 && file.ContentType.Contains("msword"))
    { try
        { int id;
          if (int.TryParse(this.txtBoxClienteID.Text, out id))
          { string caminho = "~/ficheiros/" + id.ToString() +
System.IO.Path.GetExtension(file.FileName);
            con = new
SqlConnection(System.Configuration.ConfigurationManager.ConnectionStrings["con"
].ConnectionString);
            con.Open();
            cmd = new SqlCommand("update Clientes set FicheiroNOME =
@FicheiroNOME , FicheiroTipo=@FicheiroTIPO , FicheiroPATH=@FicheiroPATH where
ClienteID=@ClienteID", con);
            cmd.Parameters.Add("@FicheiroNOME", SqlDbType.VarChar, 80);
            cmd.Parameters["@FicheiroNOME"].Value = file.FileName;
            cmd.Parameters.Add("@FicheiroTIPO", SqlDbType.VarChar, 80);
            cmd.Parameters["@FicheiroTIPO"].Value = file.ContentType;
            cmd.Parameters.Add("@FicheiroPATH", SqlDbType.VarChar, 255);
            cmd.Parameters["@FicheiroPATH"].Value = caminho;
            cmd.Parameters.Add("@ClienteID", SqlDbType.Int);
            cmd.Parameters["@ClienteID"].Value = id;
            cmd.ExecuteNonQuery();
            FileUpload.PostedFile.SaveAs(Server.MapPath(caminho));
            lblInfo.Text = "Ficheiro inserido com sucesso.";
          }
        }
    } catch (SqlException x)
    {
        lit.Text=x.Message;
    }
    finally
    {
        if (con.State == ConnectionState.Open) con.Close();
    }
}
else
{
    lblInfo.Text = "ficheiro inválido, apenas ficheiros no formato '.doc'";
}
}
```

**Código 11 - Evento "btnUpload\_Click" do ficheiro "11GestaoClientes.aspx.cs", Fonte Própria**

- Ficheiro “21GestaoRegistos.aspx.cs”

Na página de gestão dos registos o método “*devolveTabela*” serve para preencher a *GridView* com os registos do cliente pretendido após pesquisa. Para isso, é utilizado o comando SQL “*SELECT*” e a ordenação dos registos é feita por data mais recente.

Na *GridView* são apresentadas as seguintes informações do registo: data de criação, o sumário e descrição, o nome e especialidade da pessoa que o inseriu.

```
public void devolveTabela()
{
    try
    {
        con = new
SqlConnection(System.Configuration.ConfigurationManager.ConnectionStrings["con"
].ConnectionString);
        comm = new SqlCommand();
        comm.CommandType = CommandType.Text;
        comm.CommandText = "SELECT R.RegistoDATA as 'DATA', R.RegistoTIT as
'SUMÁRIO', R.RegistoDESC as 'DESCRIÇÃO', S.StaffNOME as 'COLABORADOR',
EspecialidadeDESIGN as 'ESPECIALIDADE' FROM Registos R INNER JOIN Staff S on
R.StaffID=S.StaffID WHERE ClienteID='" + txtBoxSearch.Text + "'ORDER BY 'DATA'
DESC";

        comm.Connection = con;
        con.Open()
        reader = comm.ExecuteReader();
        dt = new DataTable();
        dt.Load(reader);
        GridRegistos.DataSource = dt;
        GridRegistos.DataBind();
        lit.Text = con.State.ToString();
    }
    catch (Exception ex)
    {
        lit.Text = ex.Message;
    }
    finally
    {
        if (con != null) con.Close();
    }
}
```

**Código 12 - Método "devolveTabela" do ficheiro "21GestaoRegistos.aspx.cs", Fonte Própria**

O evento “*txtBoxSearch\_TextChanged*” neste ficheiro está associado à pesquisa de registos associados ao cliente procurado. Com este evento são despoletados os métodos “*devolveTabela*” e “*mostrarNomeCliente*”. Neste último método faz-se a procura do cliente na BD verificando, através de declarações *if, else*, a sua existência e se o número de cliente foi inserido corretamente no campo de pesquisa.

```
protected void txtBoxSearch_TextChanged(object sender, EventArgs e)
{
    devolveTabela();
    txtBoxClienteID.Text = txtBoxSearch.Text;
    lblInfo.Text = "";
    mostraNomeCliente();
}

protected void mostraNomeCliente()
{
    con = new
SqlConnection(System.Configuration.ConfigurationManager.ConnectionStrings["con"
].ConnectionString);
    int id;
    if (int.TryParse( txtBoxSearch.Text,out id))
    {
        SqlDataAdapter da = new SqlDataAdapter("SELECT * FROM Clientes
WHERE ClienteID='" + id + "'", con);
        DataTable dt = new DataTable();
        da.Fill(dt);

        if (dt.Rows.Count > 0)
        {
            btnSave.Visible = true;
            lblNomeClt.Text = dt.Rows[0][1].ToString();
        }
        else
        {
            lblInfo.Text = "Cliente não encontrado";
            txtBoxClienteID.Text = "";
            btnSave.Visible = false;
            lblNomeClt.Text = "";
        }
    }
    else
    {
        lblInfo.Text = "Número de cliente incorreto";
    }
}
}
```

**Código 13 - Evento "txtBoxSearch\_TextChanged" e Método "mostrarNomeCliente" do ficheiro "21GestaoRegistos.aspx.cs", Fonte Própria**

- Ficheiro “31GestaoUsers.aspx.cs”

A área de gestão de utilizadores é apenas acessível ao administrador e nela é possível fazer qualquer alteração aos dados dos utilizadores. Para seleccionar qual o utilizador ao qual pretende fazer alteração foi criada uma *drop-down list* com o nome “ddlUsername” e está associada ao evento “DropDownList1\_SelectedIndexChanged” que permite seleccionar da BD os dados do utilizador pretendido.

```
protected void DropDownList1_SelectedIndexChanged(object sender, EventArgs e)
{
    con = new
    SqlConnection(System.Configuration.ConfigurationManager.ConnectionStrings["con"
    ].ConnectionString);

    SqlDataAdapter da = new SqlDataAdapter("SELECT * FROM Staff WHERE
    Username='" + ddlUsername.Text + "'", con);
    DataTable dt = new DataTable();
    da.Fill(dt);
    txtBoxStaffID.Text = dt.Rows[0][0].ToString();
    txtBoxStaffNome.Text = dt.Rows[0][1].ToString();
    txtBoxStaffPNOME.Text = dt.Rows[0][2].ToString();
    txtBoxStaffUNOME.Text = dt.Rows[0][3].ToString();
    txtBoxStaffCTT1.Text = dt.Rows[0][6].ToString();
    txtBoxStaffCTT2.Text = dt.Rows[0][7].ToString();
    ddlEspecialidade.Text = dt.Rows[0][10].ToString();
    txtBoxEmail.Text = dt.Rows[0][8].ToString();
    txtBoxUsername.Text = dt.Rows[0][4].ToString();
    txtBoxPassword.Text = dt.Rows[0][5].ToString();
}
```

*Código 14 - Evento "DropDownList1\_SelectedIndexChanged" do ficheiro "31GestaoUsers.aspx.cs", Fonte Própria*

- Ficheiro “41EditarPerfil.aspx.cs”

A área de edição de perfil está acessível ao utilizador que pretenda visualizar e editar os seus dados, sendo que qualquer alteração de dados tem de ser submetida ao administrador. No entanto, é possível fazer alteração da sua palavra-passe, e para isso foi criado o evento “*btnUpdate\_Click*” associado ao botão para atualizar dados.

Este evento contém um primeiro comando SQL para selecionar da base de dados a palavra-passe atual e através de uma declaração *if, else* é verificado se foi inserida corretamente. Seguidamente é feita, através de outra declaração *if, else*, a comparação da nova palavra-passe. Se este processo for cumprido é apresentada uma mensagem de alteração de palavra-passe com sucesso.

```
protected void btnUpdate_Click(object sender, EventArgs e)
{
    con = new
SqlConnection(System.Configuration.ConfigurationManager.ConnectionStrings["con"
].ConnectionString);
    SqlDataAdapter da = new SqlDataAdapter("SELECT Password FROM Staff
WHERE Password = '" + oldPassword.Text + "'", con);
    DataTable dt = new DataTable();
    da.Fill(dt);
    if(dt.Rows.Count.ToString() == "1")
    {
        if(newPassword.Text == confPassword.Text &&
confPassword.Text.Length > 0)
        {
            con.Open();
            SqlCommand cmd = new SqlCommand("UPDATE Staff SET Password ='
+ confPassword.Text + "' WHERE Password = '" + oldPassword.Text + "'", con);
            cmd.ExecuteNonQuery();
            con.Close();
            lblInfo.Text = "Password atualizada com sucesso";
        }
        else
        {
            lblInfo.Text = "'Nova Password' e 'Confirmação Password' não
coincidem";
        }
    }
    else
    {
        lblInfo.Text = "'Password Atual' incorreta ou campo não
preenchido.";
    }
}
```

*Código 15 - Evento " btnUpdate\_Click " do ficheiro " 41EditarPerfil.aspx.cs ", Fonte Própria*

## 4. Conclusão

Com a elaboração do trabalho procurei dar resposta a falhas organizacionais que encontrei em experiências pessoais e familiares. Para dar resposta a essas lacunas, criei uma aplicação web em ASP.NET. O uso de aplicações deste tipo e de outros *softwares* similares visa, nos dias que correm, a simplificação de processos permitindo assim a obtenção de objetivos tangíveis pretendidos pelas empresas ou instituições.

Esta aplicação minimiza o uso do suporte em papel e a utilização de mecanismos diferenciados para arquivo de informação. Por conseguinte facilita a consulta de informação e simplifica a troca da mesma entre os diversos agentes.

Apesar das contrariedades, trabalhei para poder apresentar um sistema de registos mais vantajoso do que aqueles que encontrei implementados. Mesmo assim, após a sua implementação, a aplicação pode ser alvo de melhorias. Primariamente, será a implementação de um sistema de agendamentos no qual as marcações serão feitas através da aplicação criada ao invés de cada profissional marcar as suas próprias consultas mediante a sua própria agenda. Outro melhoramento será a interligação desta aplicação com o sistema de faturação ou a implementação deste diretamente na aplicação criada.

Surgiram também sugestões de melhoria após apresentação da aplicação a profissionais da área da saúde da reabilitação/desenvolvimento. Este conjunto de propostas, em certa forma, são tangivelmente implementados, destacando-se a permissão para edição dos registos dentro da mesma consulta, mesmo depois de ter sido submetido o registo, a adição de outros profissionais nomeadamente enfermeiro, nutricionista ou assistente social, a criação de um campo de diagnóstico na área de clientes e a inclusão de um módulo de prescrição eletrónica.

Esta aplicação apresenta um sistema gráfico apelativo para os utilizadores, sendo-lhes atribuído um acesso apropriado às suas permissões e que facilita e otimiza a consulta e atualização de informação.

A confidencialidade e a segurança dos dados é outro aspeto crucial da aplicação, uma vez que somente alguns dos utilizadores, credenciados para tal, podem aceder à informação confidencial dos clientes.

Esta aplicação tem margem suficiente para alterações ou atualizações futuras, se necessário.

## Bibliografia

- Breske, M., Gipson, R., & Salazar, G. (Maio de 2012). *Can You Afford Not to Use ASP.NET MVC?*
- Brynjolfsson, E., & Hitt, L. M. (Outono 2000). Beyond Computation: Information Technology, Organizational Transformation and Business Performance. *Journal of Economic Perspectives*, pp. 23-48.
- Carr, R. (12 de Setembro de 2010). *Dependency Injection*. Obtido de <http://www.blackwasp.co.uk/DependencyInjection.aspx>
- Codd, E. F. (Junho de 1970). A relational model of data for large shared data banks. *Communications of the ACM, Volume 13 Issue 6*, pp. 377-387.
- Forrest, B. (25 de Maio de 2006). *Controversy about our "Web 2.0" service mark*. Obtido de <http://radar.oreilly.com/2006/05/controversy-about-our-web-20-s.html>
- Galloway, J., Haack, P., Wilson, B., & Allen, K. S. (2011). *Professional ASP.NET MVC 3*. Indianapolis, Indiana: John Wiley & Sons, Inc.
- Groff, J. R., Weinberg, P. N., & Opper, A. J. (2010). *SQL The Complete Reference, 3rd Edition*. The McGraw-Hill Companies.
- Kreines, D. C. (2000). *Oracle SQL: The Essential Reference*. O'Reilly & Associates, Inc.
- Lerman, J. (2010). *Programming Entity Framework, 2nd Edition*. Gravenstein Highway North, Sebastopol: O'Reilly Media, Inc.
- Leshin, L. (2003). *Trisomy 21: The Story of Down Syndrome*. Obtido de <http://www.ds-health.com/trisomy.htm>
- Magnoli, D. (1997). *Globalização: Estado Nacional e Espaço Mundial*. São Paulo: Moderna.
- Meijer, E. (2007). Project Volta. Microsoft. Obtido de <http://channel9.msdn.com/ShowPost.aspx?PostID=223865>
- Milgrom, P., & Roberts, J. (1990). The Economics of Modern Manufacturing: Technology, Strategy, and Organization. Em *The American Economic Review* (pp. 511-528). American Economic Association.
- O'Reilly, T. (2007). *COMMUNICATIONS & STRATEGIES*, pp. 17-37.
- Perrow, C. (1991). A society of organizations. Em *Theory and Society* (Vol. 20, pp. 725-762). Springer.
- Rouse, M. (Agosto de 2012). *application migration*. Obtido de TechTarget: <http://searchcloudapplications.techtarget.com/definition/application-migration>
- Selly, D., Troelsen, A., & Barnaby, T. (2006). *Windows Communication Foundation. Expert ASP. NET 2.0: Advanced Application Design*. Apress.

- Stanek, W. R. (2008). *Internet Information Services (IIS) 7.0 Administrator's Pocket Consultant*. Microsoft Press.
- Stephens, R., Jones, A. D., & Plew, R. (2016). *Sams Teach Yourself SQL in 24 Hours, 6ª Edição*. Sams.
- Vogel, P. (Março de 2013). ASP.NET - Migrating ASP.NET Web Forms to the MVC Pattern with the ASP.NET Web API. *MSDN Magazine*, 28, nº 3.
- Walther, S., Hoffman, K. S., & Scott, N. (2011). *ASP.NET 4 - UNLEASHED*. Indianapolis, USA: SAMS.

# Anexos

## Anexo I

### Logótipo da aplicação



## Anexo II

### Código completo da aplicação

#### Login.aspx

```
1  <%@ Page Title="Login" Language="C#" AutoEventWireup="true"
CodeFile="Login.aspx.cs" Inherits="Login" %>
2
3  <!DOCTYPE html>
4  <html lang="en">
5
6  <head>
7      <title>Login - HOMsys</title>
8      <meta charset="utf-8">
9      <meta name="viewport" content="width=device-width, initial-scale=1.0, maximum-
scale=1.0, user-scalable=no">
10     <meta name="apple-mobile-web-app-capable" content="yes">
11
12     <!--Ficheiros CSS-->
13     <link href="css/bootstrap.min.css" rel="stylesheet" type="text/css" />
14     <link href="css/bootstrap-responsive.min.css" rel="stylesheet" type="text/css"
/>
15     <link href="css/font-awesome.css" rel="stylesheet" />
16     <link href="css/style.css" rel="stylesheet" type="text/css" />
17     <link href="css/signin.css" rel="stylesheet" type="text/css" />
18
19     <!-- Fontes Google-->
20     <link
href="http://fonts.googleapis.com/css?family=Open+Sans:400italic,600italic,400,600"
rel="stylesheet" />
21 </head>
22
23 <body bgcolor="#FFFFFF">
24     <script language="javascript">
25         setTimeout(function timeru() { $(''.alert').fadeOut(1000) }, 1500);
26     </script>
27     <script src="js/jquery-3.1.1.min.js"></script>
28     <script language="javascript">
29         $(document).ready(function () {
30             });
31     </script>
32
33     <!-- Login Background -->
34     <div style="background: #FFFFFF url('images/homsys_login_bg.png'); background-
position: center; background-size: cover;">
35
36         <!-- Logotipo HOMsys -->
37         <div style="background: url('images/homsys_logo_nb_small.png') no-repeat
center; height: 111px; margin-bottom: 50px; padding-top: 120px;"></div>
38
39         <!-- container de acessos -->
40         <div class="account-container">
41
42             <div class="content clearfix">
43
44
```

```

45         <form method="post" id="frmLogin" name="formLogin" runat="server">
46             <asp:Login ID="Login1" runat="server"
OnAuthenticate="ValidateUser">
47                 <LayoutTemplate>
48                     <table cellpadding="1" cellspacing="0" style="border-
collapse: collapse; width: 326px; height: 154px;">
49                         <tr>
50                             <td>
51                                 <table cellpadding="0">
52                                     <tr>
53                                         <td align="center"
colspan="2">Introduza as suas credenciais</td>
54                                     </tr>
55                                     <tr>
56                                         <td align="right">
57                                             <asp:Label ID="UserNameLabel"
runat="server" AssociatedControlID="UserName">Username:</asp:Label>
58                                         </td>
59                                         <td>
60                                             <asp:TextBox ID="UserName"
runat="server"></asp:TextBox>
61                                             <asp:RequiredFieldValidator
ID="UserNameRequired" runat="server" ControlToValidate="UserName" ErrorMessage="Campo
obrigatório" ToolTip="Necessário introduzir Username" ForeColor="Red"
ValidationGroup="Login1" Font-
62 Size="X-Small"></asp:RequiredFieldValidator>
63                                         </td>
64                                     </tr>
65                                     <tr>
66                                         <td align="right">
67                                             <asp:Label ID="PasswordLabel"
runat="server" AssociatedControlID="Password">Password:</asp:Label>
68                                         </td>
69                                         <td>
70                                             <asp:TextBox ID="Password"
runat="server" TextMode="Password"></asp:TextBox>
71                                             <asp:RequiredFieldValidator
ID="PasswordRequired" runat="server" ControlToValidate="Password" ErrorMessage="Campo
obrigatório" ToolTip="Necessário introduzir Password" ForeColor="Red"
ValidationGroup="Login1" Font-
72 Size="X-Small"></asp:RequiredFieldValidator>
73                                         </td>
74                                     </tr>
75                                     <tr>
76                                         <td>&nbsp;</td>
77                                         <td colspan="2">
78                                             <asp:CheckBox ID="RememberMe"
runat="server" Text="Memorizar dados" />
79                                         </td>
80                                     </tr>
81                                     <tr>
82                                         <td align="center" colspan="2"
style="color: Red;">
83                                             <asp:Literal ID="FailureText"
runat="server" EnableViewState="False"></asp:Literal>
84                                         </td>
85                                     </tr>
86                                     <tr>
87                                         <td align="right" colspan="2">
88                                             <asp:Button ID="LoginButton"
runat="server" CommandName="Login" Text="Entrar" ValidationGroup="Login1" />

```

```
89         </td>
90     </tr>
91 </table>
92 </td>
93 </tr>
94 </table>
95 </LayoutTemplate>
96 </asp:Login>
97 </form>
98 </div>
99 <!-- /content -->
100 </div>
101 <!-- /account-container -->
102 </div>
103 <script src="js/bootstrap.js"></script>
104 <script src="js/signin.js"></script>
105 </body> </html>
```

## Login.aspx.cs

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.UI;
6  using System.Web.UI.WebControls;
7  using System.Web.UI.WebControls.WebParts;
8  using System.Data;
9  using System.Data.SqlClient;
10 using System.Configuration;
11 using System.Web.Security;
12 using System.Web.UI.HtmlControls;
13 using System.Collections;
14
15 public partial class Login : System.Web.UI.Page
16 {
17     SqlConnection con = new SqlConnection("Data
Source=(LocalDB)\v11.0;AttachDbFilename=D:\\[ISTEC]\\[Disciplinas]\\_PROJETO_GLOBAL\\ho
msys\\development\\HOMsys\\App_Data\\HOMsys.mdf;Integrated Security=True"); // criar a
conexão aqui
18
19     protected void Page_Load(object sender, EventArgs e)
20     {
21         // Verifica se o utilizador já está autenticado ou não
22         if ((Session["Check"] != null) && (Convert.ToBoolean(Session["Check"]) ==
true))
23         {
24             // Se o utilizador está autenticado é reencaminhado para a página de
Painel Principal
25             if (User.Identity.IsAuthenticated)
26                 Response.Redirect("01PainelPrincipal.aspx");
27         }
28     }
29
30     protected void Login1_Authenticate(object sender, AuthenticateEventArgs e)
31     {
32         Boolean blnresult;
33         blnresult = false;
34
35         // Transpõe 'UserName' e 'Password' do controlo 'login1' para uma função de
autenticação que irá verificar o username e password do sql server.
36         // Daí vai retornar um valor verdadeiro ou falso para a variável
'blnresult'
37         blnresult = Authentication(Login1.UserName, Login1.Password);
38
39         // Se variável 'blnresult' tem um valor verdadeiro autentica o utilizador
40         if (blnresult == true)
41         {
42             // Instrução real que irá autenticar o utilizador
43             e.Authenticated = true;
44             // Armazena o modo de autenticação na variável de sessão
45             Session["Check"] = true;
46         }
47         else
48             // Então variável 'blnresult' tem um valor falso e não autentica o
utilizador
49             e.Authenticated = false;
50     }
}
```

```

51
52 // Função 'Authentication' que irá verificar o Username e Password da Base de
Dados SQL, retornando verdadeiro ou falso
53 protected static Boolean Authentication(string username, string password)
54 {
55     string sqlstring;
56     sqlstring = "Select Username, Password from [Staff] where Username='" +
username + "' and Password ='" + password + "'";
57
58     // Criar uma conexão com a Base de Dados SQL
59     System.Data.SqlClient.SqlConnection con = new
System.Data.SqlClient.SqlConnection(
60         "Data
Source=(LocalDB)\v11.0;AttachDbFilename=|DataDirectory|HOMsys.mdf;Integrated
Security=True");
61
62     // Cria um comando SQL que serão 'user connection string' e a instrução
SELECT
63     System.Data.SqlClient.SqlCommand comm = new
System.Data.SqlClient.SqlCommand(sqlstring, con);
64
65     // Cria SqlDataReader que irá executar o comando acima para obter os
valores da Base de Dados SQL
66     System.Data.SqlClient.SqlDataReader reader;
67
68     // Abre a Ligação com a Base de Dados
69     con.Open();
70
71     // Executa o Comando SQL e armazena os valores obtidos no 'reader'
72     reader = comm.ExecuteReader();
73
74     // Verifica se o 'reader' tem algum valor e retorna verdadeiro, caso
contrário retorna false
75     if (reader.Read())
76         return true;
77     else
78         return false;
79 }
80
81 // VALIDAÇÃO DE USER
82 protected void ValidateUser(object sender, AuthenticateEventArgs e)
83 {
84     int userId = 0;
85     string constr =
ConfigurationManager.ConnectionStrings["con"].ConnectionString;
86     using (SqlConnection con = new SqlConnection(constr))
87     {
88         using (SqlCommand cmd = new SqlCommand("Validate_User"))
89         {
90             cmd.CommandType = CommandType.StoredProcedure;
91             cmd.Parameters.AddWithValue("@Username", Login1.UserName);
92             cmd.Parameters.AddWithValue("@Password", Login1.Password);
93             cmd.Connection = con;
94             con.Open();
95             userId = Convert.ToInt32(cmd.ExecuteScalar());
96             con.Close();
97         }

```

```
98         switch (userId)
99         {
100             case -1:
101                 Login1.FailureText = "Username e/ou Password incorretos";
102                 break;
103             case -2:
104                 Login1.FailureText = "Conta não ativa";
105                 break;
106             default:
107                 FormsAuthentication.RedirectFromLoginPage(Login1.UserName,
Login1.RememberMeSet);
108                 Response.Redirect("01PainelPrincipal.aspx");
109                 break;
110         }
111     }
112 }
113
114 }
```

## 01PainelPrincipal.aspx

```
1  <%@ Page Title="Painel Principal" Language="C#"
MasterPageFile="~/MasterPage.master" AutoEventWireup="true"
CodeFile="01PainelPrincipal.aspx.cs" Inherits="_01PainelPrincipal" %>
2
3
4  <asp:Content ID="Content1" ContentPlaceHolderID="head" Runat="Server">
5      <!-- HEADER ContentPlaceHolder -->
6  </asp:Content>
7
8
9  <asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1"
Runat="Server">
10     <!-- CORPO ContentPlaceHolder (AQUILO QUE É CARREGADO NO "ContentPlaceHolder1"
DA MASTERPAGE) -->
11
12     <!DOCTYPE html>
13     <html style="min-height: 1008.11px;">
14     <head>
15         <meta charset="UTF-8"/>
16         <meta content="width=device-width, initial-scale=1, maximum-scale=1, user-
scalable=no" name="viewport"/>
17         <link href="css/bootstrap.min.css" rel="stylesheet" type="text/css" />
18         <link href="css/font-awesome.min.css" rel="stylesheet" type="text/css" />
19         <link href="css/ionicons.min.css" rel="stylesheet" type="text/css" />
20         <link href="css/AdminLTE.css" rel="stylesheet" type="text/css" />
21         <link href="css/dataTables.bootstrap.css" rel="stylesheet" type="text/css" />
22
23         <script type="text/javascript" src="chrome-
extension://ccbdklfbpcifppcfahmmpmbkfdjjcm/js/injected.js"></script>
24         <style type="text/css" id="igtranslator-color"></style>
25     </head>
26
27     <body class="skin-blue pace-done" style="min-height: 1008.11px;">
28
29         <!-- Content Header (Page header) -->
30         <section class="content-header">
31             <h1>Painel Principal</h1>
32         </section>
33
34         <!-- Main content -->
35
36         <div class="row">
37             <section class="col-lg-6 connectedSortable">
38
39                 <!--Start of New Patient-->
40                 <div class="box box-primary" id="loading-example">
41                     <div class="box-header">
42                         <div class="pull-right box-tools">
43                             <button class="btn btn-primary btn-sm" data-
widget='collapse' data-toggle="tooltip" title="Collapse">&#x23F7;</button>
44
45                         </div>
46                         <h3 class="box-title">&#x1f57f; CONTATOS ÚTEIS</h3>
47                     </div>
48                     <div class="box-body no-padding">
49                         <div class="table-responsive" style="padding:10px;">
50                             <p><b>CONTATOS INTERNOS:</b></p>
51                             <ul style="list-style-type:none">
```

```

52         <li><b>Direção:</b> extensão 2261</li>
53         <li><b>Receção:</b> extensão 1100</li>
54         <li><b>Gabinete 1:</b> extensão 1101</li>
55         <li><b>Gabinete 2:</b> extensão 1102</li>
56         <li><b>Gabinete 3:</b> extensão 1103</li>
57         <li><b>Gabinete 4:</b> extensão 1104</li>
58         <li><b>Gabinete 5:</b> extensão 1105</li>
59         <li><b>Gabinete 6:</b> extensão 2106</li>
60         <li><b>Gabinete 7:</b> extensão 2107</li>
61         <li><b>Gabinete 8:</b> extensão 2108</li>
62         <li><b>Gabinete 9:</b> extensão 2109</li>
63         <li><b>Serviços Técnicos e de Manutenção:</b>
extensão 1112</li>
64     </ul>
65     <br />
66     <p><b>CONTATOS EXTERNOS (Para fazer ligação externa
marque zero. ex.: 0212345678):</b></p>
67     <ul style="list-style-type:none">
68         <li><b>Número Nacional de Emergência:</b>
112</li>
69         <li><b>Polícia:</b> 218111000</li>
70         <li><b>Bombeiros:</b> 213512500</li>
71         <li><b>Saúde 24:</b> 808 24 24 24</li>
72         <li><b>Associação Portuguesa de Apoio à Vítima
(APAV):</b> 116006</li>
73         <li><b>Instituto de Apoio à Criança (IAC):</b>
213617880</li>
74         <li><b>Centro Nacional de Intoxicações
(CIAV):</b> 808250143</li>
75     </ul>
76 </div>
77 </div>
78 <div class="box-footer">
79 </div>
80 </div>
81 <!-- FIM DIV COM LINKS -->
82
83 </section>
84
85 <section class="col-lg-6 connectedSortable">
86
87     <!--Start of Patient Visited-->
88     <div class="box box-primary" id="loading-example">
89         <div class="box-header">
90             <div class="pull-right box-tools">
91                 <button class="btn btn-primary btn-sm" data-
widget='collapse' data-toggle="tooltip" title="Collapse">&#x23F7;</button>
92             </div>
93         </div>
94
95         <h3 class="box-title">&#x1f5b0; LINKS ÚTEIS</h3>
96     </div>
97     <div class="box-body no-padding">
98         <div class="table-responsive" style="padding:10px;">
99             <p><b>DOCUMENTOS MODELO:</b></p>
100            <p><a
href="documentos/Perfil_Evolutivo_Padrão.doc">Perfil Evolutivo Padrão</a></p>
101            <br />

```

```
102
103         <p><b>LEGISLAÇÃO / FISCALIDADE:</b></p>
104         <p><a href="https://dre.pt/home"
target="_blank">Diário da República Eletrónico</a></p>
105         <p><a href="documentos/DL_58_2016.pdf"
target="_blank">Decreto-Lei 58/2016 (Atendimento Prioritário) </a></p>
106         <p><a href="documentos/Folheto_AT.pdf"
target="_blank">Fiscalidade Autoridade Tributária</a></p>
107         <br />
108
109         <p><b>PORTAIS SAÚDE:</b></p>
110         <p><a href="http://www.dgs.pt/"
target="_blank">Direção Geral de Saúde</a></p>
111         <p><a href="https://www.ers.pt/"
target="_blank">Entidade Reguladora da Saúde</a></p>
112         <p><a href="http://www.igas.min-saude.pt/"
target="_blank">Inspeção Geral das Atividades em Saúde </a></p>
113         <p><a href="http://www.alimentacaosaudavel.dgs.pt/"
target="_blank">Programa Nacional para a Promoção da Alimentação Saudável</a></p>
114         <p><a href="https://www.portaldocidadao.pt/home"
target="_blank">Portal do Cidadão</a></p>
115         <p><a href="http://ec.europa.eu/health/home_pt"
target="_blank">Portal de Saúde da União Europeia</a></p>
116         <br />
117
118         <p><b>ORDENS E ASSOCIAÇÕES PROFISSIONAIS:</b></p>
119         <p><a href="https://www.ordemosmedicos.pt/"
target="_blank">Ordem dos Médicos</a></p>
120         <p><a href="http://www.ordemosnutricionistas.pt"
target="_blank">Ordem dos Nutricionistas</a></p>
121         <p><a href="https://www.ordemospsicologos.pt/pt"
target="_blank">Ordem dos Psicólogos</a></p>
122         <p><a href="http://www.apfisio.pt"
target="_blank">Associação Portuguesa dos Fisioterapeutas</a></p>
123         <p><a href="http://www.aptf.org"
target="_blank">Associação Portuguesa dos Terapeutas da Fala</a></p>
124         <p><a href="http://www.ap-to.pt"
target="_blank">Associação Portuguesa dos Terapeutas Ocupacionais</a></p>
125         <br />
126
127         <p><b>OUTROS LINKS:</b></p>
128         <p><a href="http://www.appt21.org.pt/"
target="_blank">Associação Portuguesa de Portadores de Trissomia 21 (APPT21)</a></p>
129         <p><a href="http://www.edsa.eu/"
target="_blank">European Down Syndrome Association (EDSA)</a></p>
130         <p><a href="https://worlddownsyndromeday.org/"
target="_blank">World Down Syndrome Day (WDSD)</a></p>
131         </div>
132     </div>
133     <div class="box-footer">
134     </div>
135 </div>
136 <!-- FIM DIV COM LINKS -->
137
138 </section>
139 </div>
140 <!-- Main content -->
141 </body> </html> </asp:Content>
```

## 01PainelPrincipal.aspx.cs

```
1 using System;
2 using System.Configuration;
3 using System.Data;
4 using System.Data.SqlClient;
5 using System.Linq;
6 using System.Web;
7 using System.Web.Security;
8 using System.Web.UI;
9 using System.Web.UI.HtmlControls;
10 using System.Web.UI.WebControls;
11 using System.Web.UI.WebControls.WebParts;
12 using System.Xml.Linq;
13 using System.IO;
14 using System.Globalization;
15
16 public partial class _01PainelPrincipal : System.Web.UI.Page
17 {
18     protected void Page_Load(object sender, EventArgs e)
19     {
20
21     }
22 }
```

## 11GestaoClientes.aspx

```
1 <%@ Page Title="Gestão de Clientes" Language="C#"
MasterPageFile="~/MasterPage.master" AutoEventWireup="true"
CodeFile="11GestaoClientes.aspx.cs" Inherits="_11GestaoClientes" %>
2
3 <asp:Content ID="Content1" ContentPlaceHolderID="head" runat="Server">
4 <!-- HEADER ContentPlaceHolder -->
5 </asp:Content>
6
7
8 <asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1"
runat="Server">
9 <!-- CORPO ContentPlaceHolder (AQUILO QUE É CARREGADO NO "ContentPlaceHolder1"
DA MASTERPAGE) -->
10 <!DOCTYPE html>
11 <html>
12 <head>
13 <meta charset="UTF-8" />
14 <meta content="width=device-width, initial-scale=1, maximum-scale=1, user-
scalable=no" name="viewport" />
15 <link href="css/bootstrap.min.css" rel="stylesheet" type="text/css" />
16 <link href="css/font-awesome.min.css" rel="stylesheet" type="text/css" />
17 <link href="css/ionicons.min.css" rel="stylesheet" type="text/css" />
18 <link href="css/AdminLTE.css" rel="stylesheet" type="text/css" />
19 <link href="css/dataTables.bootstrap.css" rel="stylesheet" type="text/css"
/>
20 <script type="text/javascript" src="chrome-
extension://ccbok1fbpcifppcfahmpmbkfdjjccm/js/injected.js"></script>
21 <style type="text/css" id="igtranslator-color"></style>
22 </head>
23
24 <body class="skin-blue pace-done">
25
26 <!-- Content Header (Page header) -->
27 <section class="content-header">
28 <h1>Gestão de Clientes</h1>
29 <ol class="breadcrumb">
30 <li><a href="01PainelPrincipal.aspx">Painel Principal</a></li>
31 <li><a href="#">Área de clientes</a></li>
32 <li class="active">Gestão de Clientes</li>
33 </ol>
34 </section>
35
36 <!-- Main content -->
37 <section class="content">
38 <script language="javascript">
39 function validate() {
40 if (onclick('Tem certeza de que pretende guardar?')) {
41 return true;
42 } else {
43 return false;
44 }
45 }
46 </script>
47
48 <div class="row">
49 <div class="col-md-12">
50 <!-- FORMULÁRIO NOVO CLIENTE -->
51 <form role="form" method="post" runat="server">
```

```

52         <div class="box">
53
54             <!-- Botões CANCELAR/GUARDAR/ATUALIZAR/APAGAR -->
55             <div class="box-footer clearfix">
56                 <a href="#01PainelPrincipal.aspx" class="btn btn-
default" >#x1F5D9; Cancelar</a>
57                 <asp:Button runat="server" CssClass="btn btn-
primary" name="Save" ID="btnSave" Text="#x1F5AB; Criar Cliente"
OnClick="btnSave_Click"/>
58                 <asp:Button runat="server" CssClass="btn btn-
primary" name="Update" ID="btnUpdate" Text="#x1F5D8; Atualizar Dados"
OnClick="btnUpdate_Click" />
59                 <asp:Button runat="server" CssClass="btn btn-
primary" name="Delete" ID="btnDelete" Text="#x1F5D1; Apagar Cliente"
OnClick="btnDelete_Click" />
60                 <br />
61                 <asp:Label ID="lblInfo" runat="server"
ForeColor="Red" Style="font-weight: 700"></asp:Label>
62                 <br />
63                 <table>
64                     <tr>
65                         <td><asp:TextBox ID="txtBoxSearch"
runat="server" AutoPostBack="true" class="form-control input-sm" placeholder="NIF ou
nome completo" Style="width: 365px;" TextMode="Search"
66 OnTextChanged="txtBoxSearch_TextChanged"></asp:TextBox></td>
67                         <td><asp:Button ID="btnSearch"
runat="server" CssClass="btn btn-primary" name="btnDelete" Text="#x2315; Pesquisar
Cliente" /></td>
68                     </tr>
69                 </table>
70                 <br />
71             </div>
72
73             <!-- TABULAÇÕES/SEPARADORES DAS INFORMAÇÕES -->
74             <div class="box-body table-responsive">
75                 <div class="nav-tabs-custom">
76                     <ul class="nav nav-tabs" >
77                         <li class="active" ><a href="#tab_1" data-
toggle="tab" >Informações Pessoais</a></li>
78                         <li><a href="#tab_2" data-
toggle="tab">Informações de Contacto</a></li>
79                         <li runat="server" id="0Itab"><a
href="#tab_3" data-toggle="tab">Outras Informações</a></li>
80                     </ul>
81
82                     <div class="tab-content">
83                         <div class="tab-pane active" id="tab_1">
84                             <table>
85                                 <tr>
86                                     <td colspan="2">Preenchimento
obrigatório = <font color="#FF0000" >*</font></td>
87                                 </tr>
88                                 <tr>
89                                     <td colspan="2">&nbsp;</td>
90                                 </tr>
91

```

```

92                                     <tr>
93                                     <td width="12%">Nº Cliente</td>
94                                     <td width="88%"><asp:TextBox
ID="txtBoxClienteID" runat="server" class="form-control input-sm" Style="width: 100px;"
ReadOnly="true" BackColor="#e8e8e8" ></asp:TextBox></td>
95                                     <tr><td>&nbsp;</td></tr>
96                                     </tr>
97                                     <tr>
98                                     <td width="12%">Nome Completo
<font color="#FF0000">*</font></td>
99                                     <td width="88%"><asp:TextBox
ID="txtBoxClienteNome" runat="server" class="form-control input-sm" Style="width:
500px;" required="true"></asp:TextBox>
100                                     <asp:RegularExpressionValidator
ID="RegularExpressionValidator1" runat="server" ErrorMessage="Campo contém caractere(s)
inválido(s)" ControlToValidate="txtBoxClienteNome" ValidationExpression="^[a-zA-
101 ZÁÁáÃãÄéÉéÊêËëÍíÎîÏóÓóÔôÕõÚúÛÛÇçÄäËëÏïÿñ\s]*$"
ForeColor="Red"></asp:RegularExpressionValidator></td>
102                                     </tr>
103                                     <tr>
104                                     <td width="12%">Primeiro Nome
<font color="#FF0000">*</font></td>
105                                     <td width="88%"><asp:TextBox
ID="txtBoxClientePNOME" runat="server" class="form-control input-sm" Style="width:
250px;" required="true"></asp:TextBox>
106                                     <asp:RegularExpressionValidator
ID="RegularExpressionValidator2" runat="server" ErrorMessage="Campo contém caractere(s)
inválido(s)" ControlToValidate="txtBoxClientePNOME" ValidationExpression="^[a-zA-
107 ZÁÁáÃãÄéÉéÊêËëÍíÎîÏóÓóÔôÕõÚúÛÛÇçÄäËëÏïÿñ\s]*$"
ForeColor="Red"></asp:RegularExpressionValidator></td>
108                                     </tr>
109                                     <tr>
110                                     <td width="12%">Último Nome
<font color="#FF0000">*</font></td>
111                                     <td width="88%"><asp:TextBox
ID="txtBoxClienteUNOME" runat="server" class="form-control input-sm" Style="width:
250px;" required="true"></asp:TextBox>
112                                     <asp:RegularExpressionValidator
ID="RegularExpressionValidator3" runat="server" ErrorMessage="Campo contém caractere(s)
inválido(s)" ControlToValidate="txtBoxClienteUNOME" ValidationExpression="^[a-zA-
113 ZÁÁáÃãÄéÉéÊêËëÍíÎîÏóÓóÔôÕõÚúÛÛÇçÄäËëÏïÿñ\s]*$"
ForeColor="Red"></asp:RegularExpressionValidator></td>
114                                     </tr>
115                                     <tr>
116                                     <td width="12%">NIF <font
color="#FF0000">*</font></td>
117                                     <td width="88%"><asp:TextBox
ID="txtBoxNIF" runat="server" class="form-control input-sm" Style="width: 100px;"
required="true" MaxLength="9" pattern="^[12][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]"></asp:TextBox>
118 <asp:RegularExpressionValidator ID="RegularExpressionValidator4" runat="server"
ErrorMessage="NIF inválido" ControlToValidate="txtBoxNIF"
ValidationExpression="^[12][0-9][0-9][0-9][0-9][0-9][0-9][0-9]"
119 ForeColor="Red"></asp:RegularExpressionValidator></td>
120                                     </tr>

```



```

159                                     <td
width="44%"><asp:TextBox runat="server" type="text" name="CP4input" value=""
ID="txtCP4" class="form-control input-sm" placeholder="xxxx" MaxLength="4"
title="4digCodpostal" pattern="^[1-9][0-9][0-
160 9][0-9]$" Style="width: 50px;" required="true"></asp:TextBox></td>
161                                     </td> - </td>
162                                     <td
width="44%"><asp:TextBox runat="server" type="text" name="CP3input" value=""
ID="txtCP3" class="form-control input-sm" placeholder="xxx" MaxLength="3"
title="3digCodpostal" pattern="^[0-9][0-9][0-
163 9]$" Style="width: 50px;" required="true"></asp:TextBox></td>
164                                     </tr>
165                                 </table>
166                             </td>
167                         </tr>
168                     <tr><td>&nbsp;</td></tr>
169                     <tr>
170                         <td width="12%">Localidade
<font color="#FF0000">*</font></td>
171                         <td width="88%"><asp:TextBox
runat="server" type="text" name="txtLocalidade" value="" ID="txtLocalidade"
class="form-control input-sm" placeholder="Localidade" Style="width: 200px;"
required="true"></asp:TextBox>
172                             <asp:RegularExpressionValidator
ID="RegularExpressionValidator6" runat="server" ErrorMessage="Campo contém caractere(s)
inválido(s)" ControlToValidate="txtLocalidade" ValidationExpression="^[a-zA-
173 ZÁÁáÃãÄéÉéÊêËëÍíÎîÏóÓóÔôÕúÚúÇçÄäËëÏöÿñ\s]*$"
ForeColor="Red"></asp:RegularExpressionValidator></td>
174                             </tr>
175                         <tr>
176                             <td width="12%">Contacto 1
<font color="#FF0000">*</font></td>
177                             <td width="88%"><asp:TextBox
runat="server" type="text" name="cttPrincipal" value="" ID="txtCttPrincipal"
class="form-control input-sm" placeholder="Principal" MaxLength="9" title="ctt1"
pattern="^[293][0-9][0-9][0-
178 9][0-9][0-9][0-9][0-9][0-9]$" Style="width: 100px;" required="true"></asp:TextBox>
179                             <asp:RegularExpressionValidator
ID="RegularExpressionValidator7" runat="server" ErrorMessage="Contacto inválido"
ControlToValidate="txtCttPrincipal" ValidationExpression="^[293][0-9][0-9][0-9][0-
180 9][0-9][0-9][0-
181 9]$" ForeColor="Red"></asp:RegularExpressionValidator></td>
181                             </tr>
182                         <tr>
183                             <td width="12%">Contacto 2</td>
184                             <td width="88%"><asp:TextBox
runat="server" type="text" name="cttOutro" value="" ID="txtCttOutro" class="form-
control input-sm" placeholder="Outro" MaxLength="9" title="ctt2" Style="width:
185 100px;"></asp:TextBox></td>
186                             <tr><td>&nbsp;</td></tr>
187                         </tr>
188                     <tr>
189                         <td width="12%">E-mail</td>
190                         <td width="88%"><asp:TextBox
runat="server" type="text" name="email" value="" ID="txtEmail" class="form-control
input-sm" placeholder="Endereço e-mail" title="EndEmail" pattern="[a-z0-9._-]+@[a-z0-
191 z]{2,3}]" Style="width: 250px;"></asp:TextBox>

```

```

192 <asp:RegularExpressionValidator ID="RegularExpressionValidator9" runat="server"
ErrorMessage="e-mail inválido" ControlToValidate="txtEmail" ValidationExpression="[a-
z0-9._-]+@[a-z0-9.-]+\.[a-z]{2,3}$"
193 ForeColor="Red"></asp:RegularExpressionValidator></td>
194 </tr>
195 </table>
196 </div> <!-- FIM TABELA 2 DO FORMULÁRIO -->
197
198 <div class="tab-pane" id="tab_3">
199 <table>
200 <tr>
201 <td colspan="2"></td>
202 </tr>
203 <tr>
204 <td width="12%"><asp:FileUpload
ID="FileUpload" runat="server" /></td>
205 <td><asp:Button runat="server"
CssClass="btn btn-primary" name="Upload" ID="btnUpload" Text="&#x2191; Upload"
OnClick="btnUpload_Click"/></td>
206 </tr>
207 </table>
208 <br />
209 <table id="tbl1">
210 <tr>
211 <td width="12%"
id="path">Ficheiro: </td>
212 <td width="88%"> <asp:HyperLink
ID="fich" runat="server" >HyperLink</asp:HyperLink> </td>
213 </tr>
214 </table>
215 </div> <!-- FIM TABELA 3 DO FORMULÁRIO
(UPLOAD FILE) -->
216 </div> <!-- fim 'div' comum das tres TABELAS --
>
217 </div>
218 </div>
219
220 </div>
221 </form>
222 </div>
223
224 </div>
225 </section>
226 </body>
227 </html>
228 </asp:Content>

```

## 11GestaoClientes.aspx.cs

```
1 using System;
2 using System.Configuration;
3 using System.Data;
4 using System.Data.SqlClient;
5 using System.Linq;
6 using System.Web;
7 using System.Web.Security;
8 using System.Web.UI;
9 using System.Web.UI.HtmlControls;
10 using System.Web.UI.WebControls;
11 using System.Web.UI.WebControls.WebParts;
12 using System.Xml.Linq;
13 using System.IO;
14 using System.Globalization;
15
16 public partial class _11GestaoClientes : System.Web.UI.Page
17 {
18     SqlConnection con = new SqlConnection("Data
19 Source=(LocalDB)\v11.0;AttachDbFilename=D:\\[ISTEC]\\[Disciplinas]\\_PROJETO_GLOBAL\\ho
20 msys\\development\\HOMsys\\App_Data\\HOMsys.mdf;Integrated Security=True"); // criar a
21 conexão aqui
22
23     protected SqlCommand cmd;
24     protected SqlDataReader reader;
25     protected DataTable dt;
26
27     // Page_Load
28     protected void Page_Load(object sender, EventArgs e)
29     {
30         // esconde botões caso não seja RECECIONISTA OU IT_ADMIN
31         if (Page.User.Identity.Name == "Rececao" || Page.User.Identity.Name ==
32 "Admin")
33         {
34             btnSave.Visible = true;
35             btnUpdate.Visible = true;
36             btnDelete.Visible = true;
37         }
38         else
39         {
40             btnSave.Visible = false;
41             btnUpdate.Visible = false;
42             btnDelete.Visible = false;
43         }
44
45         // esconde outras informações clínicas do cliente caso User = Rececao
46         if (Page.User.Identity.Name == "Rececao")
47         {
48             OItab.Visible = false;
49         }
50     }
51
52     // Botão CRIAR CLIENTE
53     protected void btnSave_Click(object sender, EventArgs e)
54     {
55         con = new
56 SqlConnection(System.Configuration.ConfigurationManager.ConnectionStrings["con"].Conne
57 ctionString); // instância da ConnectionString
58         con.Open();
```

```

52     cmd = new SqlCommand("INSERT INTO Clientes(ClienteNOME, ClientePNOME,
ClienteUNOME, ClienteNIF, ClienteDATANASC, ClienteGEN, ClienteMORADA, ClienteCP4,
ClienteCP3, ClienteCPALF, ClienteCTT1, ClienteCTT2, ClienteEMAIL) VALUES('" +
txtBoxClienteNome.Text + "',
53     '" + txtBoxClientePNOME.Text + "', '" + txtBoxClienteUNOME.Text + "', '" +
txtBoxNIF.Text + "', '" + txtBoxDNasc.Text + "', '" + ddlGenero.Text + "', '" +
txtBoxMorada.Text + "', '" + txtCP4.Text + "', '" + txtCP3.Text + "', '" +
txtLocalidade.Text + "', '" +
54     txtCttPrincipal.Text + "', '" + txtCttOutro.Text + "', '" + txtEmail.Text + ')",
con);
55     cmd.ExecuteNonQuery();
56     con.Close();
57     lblInfo.Text = "Cliente criado com sucesso";
58     txtBoxSearch.Text = "";
59 }
60
61 // Botão ATUALIZAR DADOS
62 protected void btnUpdate_Click(object sender, EventArgs e)
63 {
64     con = new
SqlConnection(System.Configuration.ConfigurationManager.ConnectionStrings["con"].Conne
ctionString); // instância da ConnectionString
65     con.Open();
66     cmd = new SqlCommand("UPDATE Clientes SET ClienteNOME='" +
txtBoxClienteNome.Text + "', ClientePNOME='" + txtBoxClientePNOME.Text + "',
ClienteUNOME='" + txtBoxClienteUNOME.Text + "', ClienteNIF='" + txtBoxNIF.Text + "',
ClienteDATANASC='" + txtBoxDNasc.Text
67     + "', ClienteGEN='" + ddlGenero.Text + "', ClienteMORADA='" + txtBoxMorada.Text +
"', ClienteCP4='" + txtCP4.Text + "', ClienteCP3='" + txtCP3.Text + "', ClienteCPALF='"
+ txtLocalidade.Text + "', ClienteCTT1='" + txtCttPrincipal.Text + "', ClienteCTT2='" +
68     txtCttOutro.Text + "', ClienteEMAIL='" + txtEmail.Text + "' WHERE ClienteID='" +
txtBoxClienteID.Text + "'", con);
69     cmd.ExecuteNonQuery();
70     con.Close();
71     lblInfo.Text = "Dados do Cliente atualizados com sucesso ";
72     txtBoxSearch.Text = "";
73 }
74
75 // Botão APAGAR CLIENTE
76 protected void btnDelete_Click(object sender, EventArgs e)
77 {
78     con = new
SqlConnection(System.Configuration.ConfigurationManager.ConnectionStrings["con"].Conne
ctionString); // instância da ConnectionString
79     con.Open();
80     cmd = new SqlCommand("DELETE FROM Clientes WHERE ClienteID='" +
txtBoxClienteID.Text + "'", con);
81     cmd.ExecuteNonQuery();
82     con.Close();
83
84     lblInfo.Text = "Cliente eliminado com sucesso";
85     txtBoxSearch.Text = "";
86     txtBoxClienteID.Text = "";
87     txtBoxClienteNome.Text = "";
88     txtBoxClientePNOME.Text = "";
89     txtBoxClienteUNOME.Text = "";

```

```

90     txtBoxNIF.Text = "";
91     ddlGenero.Text = "";
92     txtBoxDNasc.Text = "";
93     txtBoxMorada.Text = "";
94     txtCP4.Text = "";
95     txtCP3.Text = "";
96     txtLocalidade.Text = "";
97     txtCttPrincipal.Text = "";
98     txtCttOutro.Text = "";
99     txtEmail.Text = "";
100    fich.Text = "";
101    }
102
103    // Procura Cliente :: OnTextChanged
104    protected void txtBoxSearch_TextChanged(object sender, EventArgs e)
105    {
106        con = new
107        SqlConnection(System.Configuration.ConfigurationManager.ConnectionStrings["con"].Connec
108        tionString); // instância da ConnectionString
109        // usado para ligar os dados nas 'textbox' após fazer pesquisa de cliente
110        na textbox
111        SqlDataAdapter da = new SqlDataAdapter("SELECT * FROM Clientes WHERE
112        ClienteNOME='" + txtBoxSearch.Text + "' OR ClienteNIF='" + txtBoxSearch.Text + "'",
113        con);
114        DataTable dt = new DataTable();
115        da.Fill(dt);
116        txtBoxSearch.Text = "";
117
118        if (dt.Rows.Count > 0) // se for encontrado um Cliente na Base de Dados
119        {
120            // COLUNAS DA TABELA 'Clientes'
121            txtBoxClienteID.Text = dt.Rows[0][0].ToString();
122            txtBoxClienteNome.Text = dt.Rows[0][1].ToString();
123            txtBoxClientePNOME.Text = dt.Rows[0][2].ToString();
124            txtBoxClienteUNOME.Text = dt.Rows[0][3].ToString();
125            txtBoxNIF.Text = dt.Rows[0][4].ToString();
126            DateTime data = DateTime.Parse(dt.Rows[0][5].ToString()); // Campo
127            'data nascimento' da tabela
128            txtBoxDNasc.Text = data.Date.ToString("dd/MM/yyyy"); // converte data
129            para 'ShortDateString'
130            ddlGenero.Text = dt.Rows[0][6].ToString();
131            txtBoxMorada.Text = dt.Rows[0][7].ToString();
132            txtCP4.Text = dt.Rows[0][8].ToString();
133            txtCP3.Text = dt.Rows[0][9].ToString();
134            txtLocalidade.Text = dt.Rows[0][10].ToString();
135            txtCttPrincipal.Text = dt.Rows[0][11].ToString();
136            txtCttOutro.Text = dt.Rows[0][12].ToString();
137            txtEmail.Text = dt.Rows[0][13].ToString();
138            fich.Text = dt.Rows[0][14].ToString();
139            fich.NavigateUrl = dt.Rows[0][16].ToString();
140            lblInfo.Text = "";
141        }
142    }

```

```

136     else // se NÃO for encontrado um Cliente na Base de Dados
137     {
138         txtBoxClienteID.Text = "";
139         txtBoxClienteID.Text = "";
140         txtBoxClienteNome.Text = "";
141         txtBoxClientePNOME.Text = "";
142         txtBoxClienteUNOME.Text = "";
143         txtBoxNIF.Text = "";
144         txtBoxDNasc.Text = "";
145         ddlGenero.Text = "";
146         txtBoxMorada.Text = "";
147         txtCP4.Text = "";
148         txtCP3.Text = "";
149         txtLocalidade.Text = "";
150         txtCttPrincipal.Text = "";
151         txtCttOutro.Text = "";
152         txtEmail.Text = "";
153         fich.Text = "";
154         fich.NavigateUrl = "";
155         lblInfo.Text = "Cliente não encontrado";
156     }
157 }
158
159 // Botão UPLOAD
160 protected void btnUpload_Click(object sender, EventArgs e)
161 {
162     HttpPostedFile file = FileUpload.PostedFile;
163     if (file.FileName.Length > 0 && file.ContentType.Contains("msword"))
164     {
165         try
166         {
167             int id;
168             if (int.TryParse(this.txtBoxClienteID.Text, out id))
169             {
170
171                 string caminho = "~/ficheiros/" + id.ToString() +
System.IO.Path.GetExtension(file.FileName);
172                 con = new
SqlConnection(System.Configuration.ConfigurationManager.ConnectionStrings["con"].Conne
ctionString); // instância da ConnectionString
173                 con.Open();
174                 cmd = new SqlCommand("update Clientes set FicheiroNOME =
@FicheiroNOME , FicheiroTipo=@FicheiroTIPO , FicheiroPATH=@FicheiroPATH where
ClienteID=@ClienteID", con);
175                 cmd.Parameters.Add("@FicheiroNOME", SqlDbType.VarChar, 80);
176                 cmd.Parameters["@FicheiroNOME"].Value = file.FileName;
177
178                 cmd.Parameters.Add("@FicheiroTIPO", SqlDbType.VarChar, 80);
179                 cmd.Parameters["@FicheiroTIPO"].Value = file.ContentType;
180
181                 cmd.Parameters.Add("@FicheiroPATH", SqlDbType.VarChar, 255);
182                 cmd.Parameters["@FicheiroPATH"].Value = caminho;
183
184                 cmd.Parameters.Add("@ClienteID", SqlDbType.Int);
185                 cmd.Parameters["@ClienteID"].Value = id;
186

```

```
187         cmd.ExecuteNonQuery();
188         FileUpload.PostedFile.SaveAs(Server.MapPath(caminho));
189
190         lblInfo.Text = "Ficheiro inserido com sucesso.";
191     }
192 }
193 catch (SqlException x)
194 {
195     lit.Text=x.Message;
196 }
197 finally
198 {
199     if (con.State == ConnectionState.Open) con.Close();
200 }
201 }
202 else
203 {
204     lblInfo.Text = "ficheiro inválido, apenas ficheiros no formato '.doc'";
205 }
206 }
207 }
208 }
209 }
210 } // public partial class _11AdicionarCliente : System.Web.UI.Page
```

## 21GestaoRegistos.aspx

```
1 <%@ Page Title="Gestão de Registos" Language="C#"
MasterPageFile="~/MasterPage.master" AutoEventWireup="true"
CodeFile="21GestaoRegistos.aspx.cs" Inherits="_21GestaoRegistos" %>
2
3 <asp:Content ID="Content1" ContentPlaceHolderID="head" runat="Server">
4 <!-- HEADER ContentPlaceHolder -->
5 </asp:Content>
6
7
8 <asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1"
runat="Server">
9 <!-- CORPO ContentPlaceHolder (AQUILO QUE É CARREGADO NO "ContentPlaceHolder1"
DA MASTERPAGE) -->
10 <!DOCTYPE html>
11 <html>
12 <head>
13 <meta charset="UTF-8" />
14 <meta content="width=device-width, initial-scale=1, maximum-scale=1, user-
scalable=no" name="viewport" />
15 <link href="css/bootstrap.min.css" rel="stylesheet" />
16 <link href="css/font-awesome.min.css" rel="stylesheet" type="text/css" />
17 <link href="css/ionicons.min.css" rel="stylesheet" type="text/css" />
18 <link href="css/AdminLTE.css" rel="stylesheet" type="text/css" />
19 <link href="css/dataTables.bootstrap.css" rel="stylesheet" />
20 <style type="text/css" id="igtranslator-color"></style>
21 </head>
22
23 <body class="skin-blue pace-done">
24
25 <!-- Content Header (Page header) -->
26 <section class="content-header">
27 <h1>Gestão de Registos</h1>
28 <ol class="breadcrumb">
29 <li><a href="01PainelPrincipal.aspx">Painel Principal</a></li>
30 <li><a href="#">Registos</a></li>
31 <li class="active">Gestão de Registos</li>
32 </ol>
33 </section>
34
35 <!-- Main content -->
36 <section class="content">
37
38 <div class="row">
39 <div class="col-md-12">
40 <!-- FORMULÁRIO NOVO CLIENTE -->
41 <form role="form" method="post" runat="server"
enctype="multipart/form-data">
42 <div class="box">
43
44 <!-- Botões CANCELAR/GUARDAR/ATUALIZAR/APAGAR -->
45 <div class="box-footer clearfix">
46 <a href="01PainelPrincipal.aspx" class="btn btn-
default">&#x1F5D9; Cancelar</a>
47 <asp:Button runat="server" CssClass="btn btn-
primary" name="btnSave" ID="btnSave" Text="&#x1F5AB; Inserir Registo"
OnClick="btnSave_Click" />
```



```

84         <SortedDescendingHeaderStyle
BackColor="#000065" />
85     </asp:GridView>
86     <asp:Literal ID="lit" runat="server"
></asp:Literal>
87     <!-- *****
FIM DA GRIDVIEW ***** -->
88
89     </div> <!-- FIM TABELA 1 DO FORMULÁRIO -->
90
91     <div class="tab-pane" id="tab_2">
92         <table>
93             <tr>
94                 <td colspan="2">Preenchimento
obrigatório = <font color="#FF0000"> *</font></td>
95             </tr>
96             <tr>
97                 <td colspan="2">&nbsp;&nbsp;&nbsp;</td>
98             </tr>
99
100            <tr>
101                <td width="12%">Nº Cliente</td>
102                <td width="88%"><asp:TextBox
ID="txtBoxClienteID" runat="server" class="form-control input-sm" Style="width: 100px;"
ReadOnly="true" BackColor="#e8e8e8"></asp:TextBox></td>
103            </tr>
104            <tr>
105                <td width="12%">Data Registro
<font color="#FF0000">*</font></td>
106                <td width="88%"><asp:TextBox
ID="txtBoxDateTime" runat="server" class="form-control input-sm" Style="width: 150px;"
ReadOnly="true" Wrap="true" ></asp:TextBox></td>
107            </tr>
108            <tr>
109                <td width="12%">Sumário <font
color="#FF0000">*</font></td>
110                <td width="88%"><asp:TextBox
ID="txtBoxTitulo" runat="server" class="form-control input-sm" Style="width: 250px;"
placeholder="(máx. 50 caracteres)" MaxLength="50" required="true" ></asp:TextBox></td>
111            </tr>
112            <tr>
113                <td width="12%">Descrição <font
color="#FF0000">*</font></td>
114                <td width="88%"><asp:TextBox
ID="txtBoxDescricao" runat="server" class="form-control input-sm" Style="width: 600px;
height:200px" TextMode="MultiLine" placeholder="(máx. 250 caracteres)" MaxLength="250"
115                required="true" ></asp:TextBox></td>
116            </tr>
117            <tr>
118                <td width="12%">Colaborador
<font color="#FF0000">*</font></td>
119                <td width="88%"><asp:TextBox
ID="txtBoxStaffID" runat="server" placeholder="nº colaborador" class="form-control
input-sm" Style="width: 100px;" required="true" ></asp:TextBox></td>
120            </tr>
121        </table>

```

```
122         </div> <!-- FIM TABELA 2 DO FORMULÁRIO -->
123     </div> <!-- fim 'div' comum das duas TABELAS --
>
124         </div>
125     </div>
126
127         </div>
128     </form>
129 </div>
130 </div>
131 </section>
132 </body>
133 </html>
134 </asp:Content>
```

## 21GestaoRegistos.aspx.cs

```
1  using System;
2  using System.Configuration;
3  using System.Data;
4  using System.Data.SqlClient;
5  using System.Linq;
6  using System.Web;
7  using System.Web.Security;
8  using System.Web.UI;
9  using System.Web.UI.HtmlControls;
10 using System.Web.UI.WebControls;
11 using System.Web.UI.WebControls.WebParts;
12 using System.Xml.Linq;
13 using System.IO;
14 using System.Globalization;
15
16 public partial class _21GestaoRegistos : System.Web.UI.Page
17 {
18     SqlConnection con = new SqlConnection("Data
19 Source=(LocalDB)\v11.0;AttachDbFilename=D:\\[ISTEC]\\[Disciplinas]\\_PROJETO_GLOBAL\\homsys\\development\\HOMsys\\App_Data\\HOMsys.mdf;Integrated Security=True"); // criar a
20 conexão aqui
21     protected SqlCommand comm;
22     protected SqlDataReader reader;
23     protected DataTable dt;
24
25     // Page Load
26     protected void Page_Load(object sender, EventArgs e)
27     {
28         txtBoxDateTime.Text = DateTime.Now.ToString(); // data e hora do sistema
29     }
30
31     // método DevolveTabeta()
32     public void devolveTabela() // conecta à Base de Dados para 'alimentar' a
33 GridView com os dados
34     {
35         try
36         {
37             con = new
38 SqlConnection(System.Configuration.ConfigurationManager.ConnectionStrings["con"].Conne
39 ctionString); // instância da ConnectionString
40             comm = new SqlCommand();
41             comm.CommandType = CommandType.Text;
42             comm.CommandText = "SELECT R.RegistoDATA as 'DATA', R.RegistoTIT as
43 'SUMÁRIO', R.RegistoDESC as 'DESCRIÇÃO', S.StaffNOME as 'COLABORADOR',
44 EspecialidadeDESIGN as 'ESPECIALIDADE' FROM Registos R INNER JOIN Staff S on
45 R.StaffID=S.StaffID WHERE ClienteID='"
46 + txtBoxSearch.Text + "'ORDER BY 'DATA' DESC";
47             comm.Connection = con;
48             con.Open(); // abertura da ligação à BD
49             reader = comm.ExecuteReader();
50             dt = new DataTable();
51             dt.Load(reader);
52             GridRegistos.DataSource = dt;
53             GridRegistos.DataBind();
54         }
55         catch { }
56     }
57 }
```

```

47         lit.Text = con.State.ToString(); // comentar esta linha pois ela serve
para verificar o estado da ligação
48     }
49     catch (Exception ex)
50     {
51         lit.Text = ex.Message; // comentar esta linha pois ela serve para
verificar o estado da ligação
52     }
53     finally
54     {
55         if (con != null) con.Close(); // fecho da ligação à BD
56     }
57 }
58
59 // Botão INSERIR REGISTO
60 protected void btnSave_Click(object sender, EventArgs e)
61 {
62     con = new
SqlConnection(System.Configuration.ConfigurationManager.ConnectionStrings["con"].Connec
tionString); // instância da ConnectionString
63     con.Open();
64     SqlCommand cmd = new SqlCommand("INSERT INTO Registos(ClienteID,
RegistoDATA, RegistoTIT, RegistoDESC, StaffID) VALUES('" + txtBoxClienteID.Text + "',
'" + txtBoxDateTime.Text + "', '" + txtBoxTitulo.Text + "', '" + txtBoxDescricao.Text +
'", '" +
65     txtBoxStaffID.Text + "')", con);
66     cmd.ExecuteNonQuery();
67     con.Close();
68     lblInfo.Text = "Registo inserido com sucesso";
69     txtBoxSearch.Text = String.Empty;
70     txtBoxClienteID.Text = String.Empty;
71     txtBoxTitulo.Text = String.Empty;
72     txtBoxDescricao.Text = String.Empty;
73     txtBoxStaffID.Text = String.Empty;
74 }
75
76 // Procura Cliente :: OnTextChanged
77 protected void txtBoxSearch_TextChanged(object sender, EventArgs e)
78 {
79     devolveTabela();
80     txtBoxClienteID.Text = txtBoxSearch.Text;
81     lblInfo.Text = "";
82     mostraNomeCliente();
83 }
84
85 // método mostraNomeCliente() :: após pesquisa de registos
86 protected void mostraNomeCliente()
87 {
88     con = new
SqlConnection(System.Configuration.ConfigurationManager.ConnectionStrings["con"].Connec
tionString); // instância da ConnectionString
89     // usado para ligar os dados nas 'textbox' após fazer pesquisa de cliente
na textbox
90     int id;
91     if (int.TryParse( txtBoxSearch.Text,out id))

```

```
92     {
93         SqlDataAdapter da = new SqlDataAdapter("SELECT * FROM Clientes WHERE
ClienteID='" + id + "'", con);
94         DataTable dt = new DataTable();
95         da.Fill(dt);
96
97         if (dt.Rows.Count > 0) // se existe cliente na base de dados
98         {
99             btnSave.Visible = true;
100            lblNomeClt.Text = dt.Rows[0][1].ToString();
101        }
102        else // se não existe cliente na base de dados
103        {
104            lblInfo.Text = "Cliente não encontrado";
105            txtBoxClienteID.Text = "";
106            btnSave.Visible = false;
107            lblNomeClt.Text = "";
108        }
109    }
110    else
111    {
112        lblInfo.Text = "Número de cliente incorreto";
113    }
114
115 }
116
117 }
```

## 31GestaoUsers.aspx

```
1 <%@ Page Title="Adicionar Utilizador" Language="C#"
MasterPageFile="~/MasterPage.master" AutoEventWireup="true"
CodeFile="31GestaoUsers.aspx.cs" Inherits="_31AdicionarUser" %>
2
3 <asp:Content ID="Content1" ContentPlaceHolderID="head" Runat="Server">
4 <!-- HEADER ContentPlaceHolder -->
5 </asp:Content>
6 <asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1"
Runat="Server">
7 <!-- CORPO ContentPlaceHolder (AQUILO QUE É CARREGADO NO "ContentPlaceHolder1"
DA MASTERPAGE) -->
8
9 <!DOCTYPE html>
10 <html>
11 <head>
12 <meta charset="UTF-8"/>
13 <meta content="width=device-width, initial-scale=1, maximum-scale=1, user-
scalable=no" name="viewport"/>
14
15 <link href="css/bootstrap.min.css" rel="stylesheet" type="text/css" />
16 <link href="css/font-awesome.min.css" rel="stylesheet" type="text/css" />
17 <link href="css/ionicons.min.css" rel="stylesheet" type="text/css" />
18 <link href="css/AdminLTE.css" rel="stylesheet" type="text/css" />
19 <link href="css/dataTables.bootstrap.css" rel="stylesheet" type="text/css" />
20
21 <script type="text/javascript" src="chrome-
extension://ccbdoklfbpcifppcfahmmpmbkfdjjccm/js/injected.js"></script>
22 <style type="text/css" id="igtranslator-color"></style>
23 </head>
24
25 <body class="skin-blue pace-done">
26
27 <!-- Content Header (Page header) -->
28 <section class="content-header">
29 <h1>Gestão de Utilizadores</h1>
30 <ol class="breadcrumb">
31 <li><a href="01PainelPrincipal.aspx"> Painel
Principal</a></li>
32 <li><a href="#">Área de Administração</a></li>
33 <li class="active">Gestão de Utilizadores</li>
34 </ol>
35 </section>
36
37 <!-- Main content -->
38 <section class="content">
39
40
41 <div class="row">
42 <div class="col-md-12">
43 <form id="formUsers" name="formUsers" runat="server"
method="post" >
44 <div class="box">
45
46 <!-- Botões
CANCELAR/GUARDAR/ATUALIZAR/APAGAR -->
47 <div class="box-footer clearfix">
```

```

48         <a href="#01PainelPrincipal.aspx" class="btn btn-
default">&#x1F5D9; Cancelar</a>
49         <asp:Button runat="server" CssClass="btn btn-
primary" name="btnSave" ID="btnSave" Text="&#x1F5AB; Criar Utilizador"
OnClick="btnSave_Click" onsubmit="return confirm('Pretende inserir utilizador?');" />
50         <asp:Button runat="server" CssClass="btn btn-
primary" name="btnUpdate" ID="btnUpdate" Text="&#x1F5D8; Atualizar"
OnClick="btnUpdate_Click" onsubmit="return confirm('Pretende alterar dados?');" />
51         <asp:Button runat="server" CssClass="btn btn-
primary" name="btnDelete" ID="btnDelete" Text="&#x1F5D1; Apagar"
OnClick="btnDelete_Click" onsubmit="return confirm('Pretende apagar utilizador?');"/>
52         <br />
53         <asp:Label ID="lblInfo" runat="server"
ForeColor="Red" style="font-weight: 700"></asp:Label>
54         <br />
55     </div>
56     <div class="box-body table-responsive">
57         <div class="nav-tabs-custom">
58             <ul class="nav nav-tabs">
59                 <li class="active"><a
href="#tab_1" data-toggle="tab">Informação Pessoal</a></li>
60                 <li><a href="#tab_2" data-
toggle="tab">Outras Informações</a></li>
61             </ul>
62             <div class="tab-content">
63                 <div class="tab-pane active"
id="tab_1">
64                     <table cellpadding="3"
cellspacing="3" width="100%">
65                         <tr>
66                             <td colspan="2">Preenchimento
Obrigatório = <font color="#FF0000">*</font></td>
67                         </tr>
68                         <tr>
69                             <td colspan="2">&nbsp;</td>
70                         </tr>
71                         <tr>
72                             <td width="12%">Selecionar
Username</td>
73                             <td width="88%">
74                                 <asp:DropDownList
ID="ddlUsername" runat="server" AutoPostBack="true" class="form-control input-sm"
Width="250px" OnSelectedIndexChanged="DropDownList1_SelectedIndexChanged">
75                                     <asp:listitem
selected="True" text="" value=""></asp:listitem>
76                                 </asp:DropDownList>
77                             </td>
78                         </tr>
79                         <tr>
80                             <td width="12%">Nº Colaborador
81                         </td>
82                         <td
width="88%"><asp:TextBox ID="txtBoxStaffID" runat="server" class="form-control input-
sm" style="width: 100px;" ReadOnly="true" BackColor="#e8e8e8" ></asp:TextBox></td>
83                     </tr>

```

```

84                                     <tr>
85                                     <td width="12%">Nome Completo
<font color="#FF0000">*</font></td>
86                                     <td
width="88%"><asp:TextBox ID="txtBoxStaffNome" runat="server" class="form-control input-
sm" style="width: 500px;" ></asp:TextBox></td>
87                                     </tr>
88                                     <tr>
89                                     <td width="12%">Primeiro Nome
<font color="#FF0000">*</font></td>
90                                     <td
width="88%"><asp:TextBox ID="txtBoxStaffPNOOME" runat="server" class="form-control
input-sm" style="width: 250px;" ></asp:TextBox></td>
91                                     </tr>
92                                     <tr>
93                                     <td width="12%">Último Nome
<font color="#FF0000">*</font></td>
94                                     <td
width="88%"><asp:TextBox ID="txtBoxStaffUNOME" runat="server" class="form-control
input-sm" style="width: 250px;" ></asp:TextBox></td>
95                                     </tr>
96                                     <tr>
97                                     <td width="12%">Contato
Principal <font color="#FF0000">*</font></td>
98                                     <td
width="88%"><asp:TextBox ID="txtBoxStaffCTT1" runat="server" class="form-control input-
sm" style="width: 100px;" MaxLength="9" TextMode="Phone" ></asp:TextBox></td>
99                                     </tr>
100                                    <tr>
101                                    <td width="12%">Outro
Contato</td>
102                                    <td
width="88%"><asp:TextBox ID="txtBoxStaffCTT2" runat="server" class="form-control input-
sm" style="width: 100px;" MaxLength="9" TextMode="Phone"></asp:TextBox></td>
103                                    </tr>
104                                    <tr>
105                                    <td
width="12%">Especialidade</td>
106                                    <td width="88%">
107                                        <asp:DropDownList
ID="ddlEspecialidade" runat="server" class="form-control input-sm" style="width:
250px;" >
108                                        <asp:ListItem
Selected="True" Value=""></asp:ListItem>
109                                        <asp:ListItem
Value="IT_ADMIN"></asp:ListItem>
110                                        <asp:ListItem
Value="Secretariado"></asp:ListItem>
111                                        <asp:ListItem
Value="Pediatria"></asp:ListItem>
112                                        <asp:ListItem
Value="Psiquiatria Infantil"></asp:ListItem>
113                                        <asp:ListItem
Value="Psiquiatria"></asp:ListItem>
114                                        <asp:ListItem
Value="Genética Médica"></asp:ListItem>

```

```

115                                     <asp:ListItem
Value="Nutrição"></asp:ListItem>
116                                     <asp:ListItem
Value="Psicologia Educacional"></asp:ListItem>
117                                     <asp:ListItem
Value="Psicologia Clínica"></asp:ListItem>
118                                     <asp:ListItem
Value="Ensino Especializado"></asp:ListItem>
119                                     <asp:ListItem
Value="Reabilitação Psicomotora"></asp:ListItem>
120                                     <asp:ListItem
Value="Terapia da Fala"></asp:ListItem>
121                                     </asp:DropDownList>
122                                     </td>
123                                 </tr>
124                             </table>
125                         </div>
126                         <div class="tab-pane" id="tab_2">
127                             <table cellpadding="3"
cellspacing="3" width="100%">
128                                 <tr>
129                                     <td colspan="2"></td>
130                                 </tr>
131                                 <tr>
132                                     <td width="12%">E-mail <font
color="#FF0000">*</font></td>
133                                     <td
width="88%"><asp:TextBox ID="txtBoxEmail" runat="server" class="form-control input-sm"
style="width: 250px;" TextMode="Email"></asp:TextBox></td>
134                                 </tr>
135                                 <tr>
136                                     <td width="12%">Username <font
color="#FF0000">*</font></td>
137                                     <td
width="88%"><asp:TextBox ID="txtBoxUsername" runat="server" class="form-control input-
sm" style="width: 250px;" ></asp:TextBox></td>
138                                 </tr>
139                                 <tr>
140                                     <td width="12%">Password <font
color="#FF0000">*</font></td>
141                                     <td
width="88%"><asp:TextBox ID="txtBoxPassword" runat="server" class="form-control input-
sm" style="width: 250px;" TextMode="Password" required ></asp:TextBox></td>
142                                 </tr>
143                             </table>
144                         </div>
145                     </div>
146                 </div>
147             </div>
148             <asp:RegularExpressionValidator runat="server"
ID="regexCTT1" ControlToValidate="txtBoxStaffCTT1" ValidationExpression="^[29][0-9][0-
9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]" ErrorMessage="*** 'Contacto Principal' inválido ***"
ForeColor="Red"
149 ></asp:RegularExpressionValidator>
150             <asp:RegularExpressionValidator runat="server"

```



### 31GestaoUsers.aspx.cs

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.UI;
6  using System.Web.UI.WebControls;
7  using System.Web.UI.WebControls.WebParts;
8  using System.Data;
9  using System.Data.SqlClient;
10 using System.Configuration;
11 using System.Web.Security;
12 using System.Web.UI.HtmlControls;
13 using System.Collections;
14
15 public partial class _31AdicionarUser : System.Web.UI.Page
16 {
17     SqlConnection con = new SqlConnection("Data
Source=(LocalDB)\v11.0;AttachDbFilename=D:\\[ISTEC]\\[Disciplinas]\\_PROJETO_GLOBAL\\homsys\\development\\HOMsys\\App_Data\\HOMsys.mdf;Integrated Security=True"); // criar a
conexão aqui
18
19     // Page_Load
20     protected void Page_Load(object sender, EventArgs e)
21     {
22         if(!IsPostBack)
23         {
24             con = new
SqlConnection(System.Configuration.ConfigurationManager.ConnectionStrings["con"].Connec
tionString); // instância da ConnectionString
25             // usado para ligar os dados da Base de dados à 'drowdownlist' do
username
26
27             SqlDataAdapter da = new SqlDataAdapter("SELECT * FROM Staff", con);
28             DataTable dt = new DataTable();
29             da.Fill(dt);
30             ddlUsername.DataSource = dt;
31             ddlUsername.DataTextField = "Username";
32             DataBind();
33         }
34     }
35
36     // Botão GUARDAR
37     protected void btnSave_Click(object sender, EventArgs e)
38     {
39         con = new
SqlConnection(System.Configuration.ConfigurationManager.ConnectionStrings["con"].Connec
tionString); // instância da ConnectionString
40         con.Open();
41         SqlCommand cmd = new SqlCommand("INSERT INTO Staff(StaffNOME, StaffPNOME,
StaffUNOME, StaffCTT1, StaffCTT2, EspecialidadeDESIGN, Email, Username, Password)
VALUES('" + txtBoxStaffNome.Text + "', '" + txtBoxStaffPNOME.Text + "', '" +
txtBoxStaffUNOME.Text +
42     "', '" + txtBoxStaffCTT1.Text + "', '" + txtBoxStaffCTT2.Text + "', '" +
ddlEspecialidade.Text + "', '" + txtBoxEmail.Text + "', '" + txtBoxUsername.Text + "',
'" + txtBoxPassword.Text + "')", con);
43         cmd.ExecuteNonQuery();
```

```

44         con.Close();
45         lblInfo.Text = "Utilizador criado com sucesso";
46     }
47
48     // Botão ATUALIZAR
49     protected void btnUpdate_Click(object sender, EventArgs e)
50     {
51         con = new
SqlConnection(System.Configuration.ConfigurationManager.ConnectionStrings["con"].Conne
ctionString); // instância da ConnectionString
52         con.Open();
53         SqlCommand cmd = new SqlCommand("UPDATE Staff SET StaffNOME='" +
txtBoxStaffNome.Text + "', StaffPNOME='" + txtBoxStaffPNOME.Text + "', StaffUNOME='" +
txtBoxStaffUNOME.Text + "', StaffCTT1='" + txtBoxStaffCTT1.Text + "', StaffCTT2='" +
txtBoxStaffCTT2.Text +
54         "', EspecialidadeDESIGN='" + ddlEspecialidade.Text + "', Email='" +
txtBoxEmail.Text + "', Password='" + txtBoxPassword.Text + "' WHERE Username='" +
ddlUsername.Text + "'", con);
55         cmd.ExecuteNonQuery();
56         con.Close();
57         lblInfo.Text = "Dados do Utilizador atualizados com sucesso ";
58     }
59
60     // Botão APAGAR
61     protected void btnDelete_Click(object sender, EventArgs e)
62     {
63         con = new
SqlConnection(System.Configuration.ConfigurationManager.ConnectionStrings["con"].Conne
ctionString); // instância da ConnectionString
64         con.Open();
65         SqlCommand cmd = new SqlCommand("DELETE FROM Staff WHERE Username='" +
ddlUsername.Text + "'", con);
66         cmd.ExecuteNonQuery();
67         con.Close();
68         lblInfo.Text = "Utilizador eliminado com sucesso";
69
70         txtBoxStaffID.Text = "";
71         txtBoxStaffNome.Text = "";
72         txtBoxStaffPNOME.Text = "";
73         txtBoxStaffUNOME.Text = "";
74         txtBoxStaffCTT1.Text = "";
75         txtBoxStaffCTT2.Text = "";
76         ddlEspecialidade.Text = "";
77         txtBoxEmail.Text = "";
78         txtBoxUsername.Text = "";
79         txtBoxPassword.Text = "";
80     }
81
82     // evento DROPDOWNLIST dos utilizadores
83     protected void DropDownList1_SelectedIndexChanged(object sender, EventArgs e)
84     {
85         con = new
SqlConnection(System.Configuration.ConfigurationManager.ConnectionStrings["con"].Conne
ctionString); // instância da ConnectionString
86

```

```
87      SqlDataAdapter da = new SqlDataAdapter("SELECT * FROM Staff WHERE
Username='" + ddlUsername.Text + "'", con);
88      DataTable dt = new DataTable();
89      da.Fill(dt);
90      txtBoxStaffID.Text = dt.Rows[0][0].ToString();
91      txtBoxStaffNome.Text = dt.Rows[0][1].ToString();
92      txtBoxStaffPNOME.Text = dt.Rows[0][2].ToString();
93      txtBoxStaffUNOME.Text = dt.Rows[0][3].ToString();
94      txtBoxStaffCTT1.Text = dt.Rows[0][6].ToString();
95      txtBoxStaffCTT2.Text = dt.Rows[0][7].ToString();
96      ddlEspecialidade.Text = dt.Rows[0][10].ToString();
97      txtBoxEmail.Text = dt.Rows[0][8].ToString();
98      txtBoxUsername.Text = dt.Rows[0][4].ToString();
99      txtBoxPassword.Text = dt.Rows[0][5].ToString();
100     }
101
102 }
```

## 41EditarPerfil.aspx

```
1  <%@ Page Title="Editar Perfil" Language="C#" MasterPageFile=~\MasterPage.master"
AutoEventWireup="true" CodeFile="41EditarPerfil.aspx.cs" Inherits="_41EditarPerfil" %>
2
3  <asp:Content ID="Content1" ContentPlaceHolderID="head" Runat="Server">
4      <!-- HEADER ContentPlaceHolder -->
5  </asp:Content>
6  <asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1"
Runat="Server">
7      <!-- CORPO ContentPlaceHolder (AQUILO QUE É CARREGADO NO "ContentPlaceHolder1"
DA MASTERPAGE) -->
8
9      <!DOCTYPE html>
10     <html>
11     <head>
12         <meta charset="UTF-8"/>
13         <meta content="width=device-width, initial-scale=1, maximum-scale=1, user-
scalable=no" name="viewport"/>
14         <link href="css/bootstrap.min.css" rel="stylesheet" type="text/css" />
15         <link href="css/font-awesome.min.css" rel="stylesheet" type="text/css" />
16         <link href="css/ionicons.min.css" rel="stylesheet" type="text/css" />
17         <link href="css/AdminLTE.css" rel="stylesheet" type="text/css" />
18         <link href="css/dataTables.bootstrap.css" rel="stylesheet" type="text/css" />
19
20         <style type="text/css" id="igtranslator-color"></style>
21     </head>
22
23     <body class="skin-blue pace-done">
24
25         <!-- Content Header (Page header) -->
26         <section class="content-header">
27             <h1>Editar Perfil</h1>
28             <ol class="breadcrumb">
29                 <li><a href="01PainelPrincipal.aspx">Painel
Principal</a></li>
30                 <li><a href="#">Perfil de Utilizador</a></li>
31                 <li class="active">Editar Perfil</li>
32             </ol>
33         </section>
34
35         <!-- Main content -->
36         <section class="content">
37             <div class="row">
38                 <div class="col-md-12">
39                     <form id="formUsers" name="formUsers" runat="server"
method="post" >
40                         <div class="box">
41
42                             <!-- Botões
CANCELAR/GUARDAR/ATUALIZAR/APAGAR -->
43                             <div class="box-footer clearfix">
44                                 <a href="01PainelPrincipal.aspx" class="btn btn-
default">&#x1F5D9; Cancelar</a>
45                                 <asp:Button runat="server" CssClass="btn btn-
primary" name="btnUpdate" ID="btnUpdate" Text="&#x1F5D8; Atualizar"
OnClick="btnUpdate_Click" />

```

```

46         <!--<asp:Button runat="server" CssClass="btn btn-
primary" name="btnSave" ID="btnSave" Text="&#x1F5AB; Guardar" OnClick="btnSave_Click"
onsubmit="return confirm('Pretende inserir utilizador?');" />
47         <asp:Button runat="server" CssClass="btn btn-
primary" name="btnUpdate" ID="btnUpdate" Text="&#x1F5D8; Atualizar"
OnClick="btnUpdate_Click" onsubmit="return confirm('Pretende alterar dados?');" />
48         <asp:Button runat="server" CssClass="btn btn-
primary" name="btnDelete" ID="btnDelete" Text="&#x1F5D1; Apagar"
OnClick="btnDelete_Click" onsubmit="return confirm('Pretende apagar utilizador?');"/>
49         --%>
50         <br />
51         <asp:Label ID="lblInfo" runat="server"
ForeColor="Red" style="font-weight: 700"></asp:Label>
52         <br />
53         <p>Qualquer alteração aos dados deve ser submetida
ao Administrador através do e-mail: <a href =
"mailto:homsysadmin@homsys.pt">homsysadmin@homsys.pt</a></p>
54         </div>
55         <div class="box-body table-responsive">
56             <div class="nav-tabs-custom">
57                 <ul class="nav nav-tabs">
58                     <li class="active"><a
href="#tab_1" data-toggle="tab">Informação Pessoal</a></li>
59                     <li><a href="#tab_2" data-
toggle="tab">Outras Informações</a></li>
60                 </ul>
61                 <div class="tab-content">
62                     <div class="tab-pane active"
id="tab_1">
63                         <table cellpadding="3"
cellspacing="3" width="100%">
64                             <tr>
65                                 <td colspan="2">Preenchimento
Obrigatório = <font color="#FF0000">*</font></td>
66                             </tr>
67                             <tr>
68                                 <td colspan="2">&nbsp;</td>
69                             </tr>
70                             <tr>
71                                 <td width="12%">Nome do
Utilizador </td>
72                                 <td width="88%"><asp:Label
runat="server" ID="lblUser"></asp:Label></td>
73                             </tr>
74                             <tr>
75                                 <td width="12%">Nº Colaborador
</td>
76                                 <td
width="88%"><asp:TextBox ID="txtBoxStaffID" runat="server" class="form-control input-
sm" style="width: 100px;" ReadOnly="true" BackColor="#e8e8e8" ></asp:TextBox></td>
77                             </tr>
78                             <tr>
79                                 <td width="12%">Nome Completo
</td>
80                                 <td
width="88%"><asp:TextBox ID="txtBoxStaffNome" runat="server" class="form-control

```



```

115                                     <td width="12%">Password Atual
<font color="#FF0000">*</font></td>
116                                     <td
width="88%"><asp:TextBox ID="oldPassword" runat="server" class="form-control input-sm"
style="width: 250px;" TextMode="Password" ></asp:TextBox></td>
117                                     </tr>
118                                     <tr><td>&nbsp;</td></tr>
119                                     <tr>
120                                     <td width="12%">Nova Password
<font color="#FF0000">*</font></td>
121                                     <td
width="88%"><asp:TextBox ID="newPassword" runat="server" class="form-control input-sm"
style="width: 250px;" TextMode="Password" ></asp:TextBox>
122 <asp:RegularExpressionValidator ID="RegularExpressionValidator1" runat="server"
ErrorMessage="Mínino 8 caracteres (deve incluir 1 letra maiúscula, 1 dígito e 1
caractere especial)"
123 ControlToValidate="newPassword" ValidationExpression="^(?=.*?[A-Z])(?=.*?[a-
z])(?=.*?[0-9])(?=.*?[#?!@$%^&*-.]).{8,}$"
ForeColor="Red"></asp:RegularExpressionValidator></td>
124                                     </tr>
125                                     <tr>
126                                     <td width="12%">Confirmação
Password <font color="#FF0000">*</font></td>
127                                     <td
width="88%"><asp:TextBox ID="confPassword" runat="server" class="form-control input-sm"
style="width: 250px;" TextMode="Password" ></asp:TextBox></td>
128                                     </tr>
129                                     <tr><td>&nbsp;</td></tr>
130                                     </table>
131                                     </div>
132                                     </div>
133                                     </div>
134                                     </div>
135                                     </div>
136                                     </form>
137                                     </div>
138                                     </div>
139 </section><!-- /.content -->
140 </body>
141 </html>
142 </asp:Content>

```

## 41EditarPerfil.aspx.cs

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.UI;
6  using System.Web.UI.WebControls;
7  using System.Web.UI.WebControls.WebParts;
8  using System.Data;
9  using System.Data.SqlClient;
10 using System.Configuration;
11 using System.Web.Security;
12 using System.Web.UI.HtmlControls;
13 using System.Collections;
14
15 public partial class _41EditarPerfil : System.Web.UI.Page
16 {
17     SqlConnection con = new SqlConnection("Data
Source=(LocalDB)\v11.0;AttachDbFilename=D:\\[ISTEC]\\[Disciplinas]\\_PROJETO_GLOBAL\\homsys\\development\\HOMsys\\App_Data\\HOMsys.mdf;Integrated Security=True"); // criar a
conexão aqui
18
19     protected void Page_Load(object sender, EventArgs e)
20     {
21         if (!IsPostBack)
22         {
23             con = new
SqlConnection(System.Configuration.ConfigurationManager.ConnectionStrings["con"].Connec
tionString); // instância da ConnectionString
24             // usado para ligar os dados da Base de dados
25             lblUser.Text = Page.User.Identity.Name;
26
27             SqlDataAdapter da = new SqlDataAdapter("SELECT * FROM Staff WHERE
Username='" + lblUser.Text + "'", con);
28             DataTable dt = new DataTable();
29             da.Fill(dt);
30             txtBoxStaffID.Text = dt.Rows[0][0].ToString();
31             txtBoxStaffNome.Text = dt.Rows[0][1].ToString();
32             txtBoxStaffPNOME.Text = dt.Rows[0][2].ToString();
33             txtBoxStaffUNOME.Text = dt.Rows[0][3].ToString();
34             txtBoxStaffCTT1.Text = dt.Rows[0][6].ToString();
35             txtBoxStaffCTT2.Text = dt.Rows[0][7].ToString();
36             txtBoxEspecialidade.Text = dt.Rows[0][10].ToString();
37             txtBoxEmail.Text = dt.Rows[0][8].ToString();
38
39         }
40
41     }
42
43     protected void btnUpdate_Click(object sender, EventArgs e)
44     {
45         con = new
SqlConnection(System.Configuration.ConfigurationManager.ConnectionStrings["con"].Connec
tionString); // instância da ConnectionString
46         SqlDataAdapter da = new SqlDataAdapter("SELECT Password FROM Staff WHERE
Password = '" + oldPassword.Text + "'", con);
47         DataTable dt = new DataTable();
```

```
48     da.Fill(dt);
49
50     // verifica se existe password atual
51     if(dt.Rows.Count.ToString() == "1")
52     {
53         // critério de comparação de nova password
54         if(newPassword.Text == confPassword.Text && confPassword.Text.Length >
55         0)
56         {
57             con.Open();
58             SqlCommand cmd = new SqlCommand("UPDATE Staff SET Password ='" +
59             confPassword.Text + "' WHERE Password = '" + oldPassword.Text + "'", con);
60             cmd.ExecuteNonQuery();
61             con.Close();
62             lblInfo.Text = "Password atualizada com sucesso";
63         }
64         else
65         {
66             lblInfo.Text = "'Nova Password' e 'Confirmação Password' não
67             coincidem";
68         }
69     }
70     else
71     {
72         lblInfo.Text = "'Password Atual' incorreta ou campo não preenchido.";
73     }
74 }
```

## MasterPage.master

```
1 <%@ Master Language="C#" AutoEventWireup="true" CodeFile="MasterPage.master.cs"
Inherits="MasterPage" %>
2
3 <!DOCTYPE html>
4 <html xmlns="http://www.w3.org/1999/xhtml">
5 <head runat="server">
6
7 <meta charset="UTF-8" />
8 <title><%: Page.Title %> - HOMsys</title>
9 <meta content="width=device-width, initial-scale=1.0" name="viewport" />
10
11 <link href="css/bootstrap.min.css" rel="stylesheet" type="text/css" />
12 <link href="css/font-awesome.min.css" rel="stylesheet" type="text/css" />
13 <link href="css/AdminLTE.css" rel="stylesheet" type="text/css" />
14 <link href="css/dataTables_bootstrap.css" rel="stylesheet" type="text/css" />
15
16 <!-- Fontes Google-->
17 <link
href="http://fonts.googleapis.com/css?family=Open+Sans:400italic,600italic,400,600"
rel="stylesheet" />
18
19
20 <asp:ContentPlaceHolder ID="head" runat="server">
21 <!-- ContentPlaceHolder id=head (PROIBIDO ESCREVER AQUI DENTRO) -->
22 </asp:ContentPlaceHolder>
23 </head>
24
25 <body class="skin-blue">
26 <!-- header logo: style pode ser encontrado em header.less -->
27 <script>
28     setTimeout(function timeru() { $(''.alert').fadeOut(1000) }, 3000);
29 </script>
30 <header class="header" style="background: url('images/header_bar_bg_01.jpg')
/>repeat-x; background-size: 100% 100%; border-bottom: 1px solid #CCC">
31 <a href="#" class="logo">
32 <!-- Adiciona o ícone de classe à imagem do logotipo ou ao ícone do
logotipo para adicionar a margem [CANTO SUPERIOR ESQUERDO] -->
33 <div class="logo-pms">
34 
35 </div>
36 </a>
37 <!-- Header Navbar: style pode ser encontrado em header.less -->
38 <nav class="navbar navbar-static-top" role="navigation" style="background:
url('images/header_bar_bg_02.jpg') repeat-x; background-size: 100% 100%; border-bottom:
1px solid #CCC">
39 <!-- Sidebar toggle button [BOTÃO ESCONDER MENU] -->
40 <a href="#" class="navbar-btn sidebar-toggle" data-toggle="offcanvas"
role="button">
41 <span class="sr-only">Toggle navigation</span>
42 <span class="icon-bar"></span>
43 <span class="icon-bar"></span>
44 <span class="icon-bar"></span>
45 </a>
46 <div class="logo2">HOMsys - Sistema de Gestão</div>
47 <div class="navbar-right">
48 <ul class="nav navbar-nav">
```

```

49
50      <!-- Conta de Utilizador: style pode ser encontrado
emdropdown.less -->
51      <li class="dropdown user user-menu">
52          <a href="#" class="dropdown-toggle" data-toggle="dropdown">
53              <asp:LoginView runat="server">
54                  <LoggedInTemplate>
55                      &#x26a1; Bem-vindo
56                      <asp:Label runat="server" ID="lblLoginName">
57                          <asp:LoginName ID="LoginName1"
runat="server" />
58                      </asp:Label>
59                      &nbsp;
60                  </LoggedInTemplate>
61              </asp:LoginView>
62          </a>
63          <ul class="dropdown-menu">
64              <!-- User image -->
65              <li class="user-header bg-light-blue">
66                  <p>
67                      <asp:Label ID="lblShowName" runat="server">
68                          <asp:LoginName ID="LoginName2"
runat="server" />
69                      </asp:Label>
70                      <br />
71                      <asp:Label ID="lblShowLastLogin"
runat="server"></asp:Label>
72                  </p>
73              </li>
74              <!-- Menu Footer-->
75              <li class="user-footer">
76                  <div class="pull-left">
77                      <a href="41EditarPerfil.aspx" class="btn btn-
default btn-flat">Perfil</a>
78                  </div>
79                  <div class="pull-right">
80                      <a href="Login.aspx" class="btn btn-default
btn-flat">&#x2718; Sair</a>
81                  </div>
82              </li>
83          </ul>
84      </li>
85  </ul>
86  </div>
87  </nav>
88  </header>
89
90  <div class="wrapper row-offcanvas row-offcanvas-left">
91
92      <!-- coluna do lado esquerdo. contém o logotipo e a barra lateral -->
93      <aside class="left-side sidebar-offcanvas">
94
95          <section class="sidebar">
96
97              <!-- barra lateral, painel de utilizador -->
98

```

```

99         <div class="user-panel">
100             <div class="pull-left info">
101                 <asp:Label ID="Labelid" runat="server">Utilizador:
102                 <asp:LoginName ID="LoginName3" runat="server" />
103                 </asp:Label>
104
105                 <!--<p id=""></p> -->
106                 <br />
107                 <a href="#">Estado: Online</a>
108             </div>
109         </div>
110         <!-- /.search form -->
111         <!-- sidebar menu: : style can be found in sidebar.less -->
112         <ul class="sidebar-menu">
113
114             <li>
115                 <a href="01PainelPrincipal.aspx"><span>&#x1F4CB; Painel
Principal</span></a>
116             </li>
117
118             <!-- INÍCIO DE Gestão Clientes :: com 'renderização' para
confirmação de utitizador -->
119             <li class="treeview" id="CltArea" runat="server">
120                 <a href="#"><span>&#x1F465; Área de Clientes</span></a>
121                 <ul class="treeview-menu">
122                     <li><a href="11GestaoClientes.aspx">&#x1F588; Gestão de
Clientes</a></li>
123                 </ul>
124             </li>
125
126             <!-- INÍCIO DE Registos :: com 'renderização' para confirmação
de utitizador -->
127             <li class="treeview" id="RegArea" runat="server">
128                 <a href="#"><span>&#x1F4D4; Registos</span></a>
129                 <ul class="treeview-menu">
130                     <li><a href="21GestaoRegistos.aspx">&#x1F589; Gestão de
Registos</a></li>
131                 </ul>
132             </li>
133
134             <!-- INÍCIO DE Área de Administração :: com 'renderização' para
confirmação de utitizador -->
135             <li class="treeview" id="AdmArea" runat="server">
136                 <a href="#"><span>&#x1F511; Área de
Administração</span></a>
137                 <ul class="treeview-menu">
138                     <li><a href="31GestaoUsers.aspx">&#x1F58E; Gestão de
Utilizadores</a></li>
139                 </ul>
140             </li>
141
142             <!-- INÍCIO de Perfil de Utilizador -->
143             <li class="treeview ">
144                 <a href="#"><span>&#x1F517; Perfil de Utilizador</span></a>
145                 <ul class="treeview-menu">
146                     <li><a href="41EditarPerfil.aspx">&#x1F5BA; Editar

```

```

147         <li><a href="Login.aspx">&#x2718; Sair</a></li>
148     </ul>
149 </li>
150 </ul>
151 </section>
152 <!-- /.sidebar -->
153 </aside>
154
155 <!-- INÍCIO do <style> -->
156 <style type="text/css">
157     #footerLO {
158         background-color: #1f1c1c;
159         height: 35px;
160         color: #f8d756;
161         bottom: 0;
162         position: fixed;
163         bottom: 0;
164         width: 100%;
165         padding: 10px;
166         margin-bottom: 0px;
167         z-index: 99999;
168     }
169
170     .linkFooter {
171         color: #f8d756 !important;
172     }
173
174     .linkFooter:hover {
175         color: #f8d756 !important;
176         text-decoration: underline !important;
177     }
178 </style>
179
180 <!-- Coluna do lado direito. Contém a barra de navegação e o conteúdo da
181 página -->
182 <aside class="right-side">
183     <!-- ContentPlaceHolder id=ContentPlaceHolder1 (PROIBIDO ESCREVER AQUI
184 DENTRO) -->
185     <asp:ContentPlaceHolder ID="ContentPlaceHolder1" runat="server">
186         <!-- ** AS PÁGINAS VÃO SER CARREGADAS AQUI ** -->
187
188     </asp:ContentPlaceHolder>
189     <!-- Main content -->
190 </aside>
191 <!-- /.right-side -->
192 </div>
193 <!-- ./wrapper -->
194 <script src="js/jquery.min.js"></script>
195 <script src="js/bootstrap.min.js" type="text/javascript"></script>
196 <script src="js/app.js" type="text/javascript"></script>
197 <!-- </form> -->
198 </body>
199 </html>

```

## Web.config

```
1 <?xml version="1.0"?>
2 <!--
3   For more information on how to configure your ASP.NET application, please visit
4   http://go.microsoft.com/fwlink/?LinkId=169433
5   -->
6 <configuration>
7
8   <connectionStrings>
9     <add name="con" connectionString="Data
Source=(LocalDB)\v11.0;AttachDbFilename=D:\[ISTEC]\[Disciplinas]\_PROJETO_GLOBAL\homsys
\development\HOMsys\App_Data\HOMsys.mdf;Integrated Security=True"/>
10  </connectionStrings>
11
12  <system.web>
13    <compilation debug="true" targetFramework="4.5.2"/>
14    <httpRuntime targetFramework="4.5.2"/>
15
16    <authentication mode="Forms">
17      <forms loginUrl="login.aspx" name="login" protection="All"/>
18    </authentication>
19
20    <authorization>
21      <deny users="?" />
22    </authorization>
23  </system.web>
24
25  <appSettings>
26    <add key="ValidationSettings:UnobtrusiveValidationMode" value="None"/>
27  </appSettings>
28
29
30
31 </configuration>
```

(Página intencionalmente deixada em branco)