



**IPG** Politécnico  
|da|Guarda  
Escola Superior  
de Tecnologia e Gestão

# RELATÓRIO DE ESTÁGIO

Curso Técnico Superior Profissional  
em Desenvolvimento de Aplicações Informáticas

Tiago Gabriel Gonçalves Pereira

julho | 2017





Escola Superior de Tecnologia e Gestão  
Instituto Politécnico da Guarda

## Relatório de Estágio

**Tiago Gabriel Gonçalves Pereira**

Relatório para a obtenção do diploma de Técnico Superior Profissional em  
Desenvolvimento de Aplicações Informáticas

Julho de 2017

# IDENTIFICAÇÃO DOS INTERVENIENTES NO ESTÁGIO

## **Estudante**

Nome: Tiago Gabriel Gonçalves Pereira;

Nº de aluno: 1012011;

Curso: TeSP Desenvolvimento de Aplicações Informáticas;

Escola: IPG - Escola Superior de Tecnologia e Gestão.

## **Local de estágio**

Empresa: Instituto Politécnico da Guarda;

Localização: Av. Dr. Francisco Sá Carneiro, Nº50 6300 - 559 Guarda;

Telefone: 271 220 111;

Correio eletrónico: ipg@ipg.pt;

Site: www.ipg.pt;

## **Estágio curricular**

Início: 06 /03/2017;

Fim: 19/07/2017;

Supervisor: Eng.º Carlos Fonseca;

Orientador: Prof. Noel Lopes.

## **PLANO DE ESTÁGIO**

O objetivo do estágio foi desenvolver uma aplicação WEB para gerir as matrículas dos alunos dos cursos TeSP (Técnico Superior Profissional) onde pude aprofundar e aplicar os conhecimentos, teóricos e práticos, adquiridos durante as aulas, entre os quais destaco:

- Análise do problema:
  - Construção de diagramas.
  - Construção de Mockups.
- Desenvolvimento da Base de Dados:
  - Construção de uma base de dados onde irá ser guardada toda a informação sobre a candidatura do aluno.
- Desenvolvimento de uma aplicação WEB:
  - Desenvolvimento de páginas WEB onde o candidato poderá preencher a sua candidatura, e anexar os documentos pedidos.
- Testes da Aplicação:
  - Detecção de erros.
  - Correção dos mesmos.
- Instalação da aplicação;

## RESUMO

O estágio curricular do curso Técnico Superior Profissional em Desenvolvimento de Aplicações Informáticas que decorreu no Instituto Politécnico da Guarda (IPG), teve a duração de setecentas e cinquenta horas. O objetivo do mesmo, foi o de aplicar na prática os conhecimentos adquiridos durante o curso.

Durante este período foram realizadas diversas tarefas, entre as quais destaco:

O projeto foi realizado no *Visual Studio*, com o objetivo de gestão de candidaturas para os cursos Técnico Superior Profissional (TeSP), onde um candidato pode preencher a candidatura, e lhe será atribuído um login. O candidato pode também aceder à candidatura preenchida, e poderá modificar alguns dos campos preenchidos ou eliminar a candidatura se o mesmo assim o entender.

Foi também desenvolvida uma base de dados no programa *SQL Server Management Studio* onde será armazenada toda a informação sobre as diversas candidaturas.

**Palavras-Chave:** Visual Studio, ASP.NET, Candidaturas, SQL Server.

## **AGRADECIMENTOS**

Gostava de agradecer, de uma forma geral, a todos os professores e colegas que tive o prazer de conhecer e conviver durante o curso de Desenvolvimento de Aplicações Informáticas dos anos letivos 2015/2016 e 2016/2017.

Pretendo agradecer a todos aqueles que tornaram possível a realização do estágio, entre os quais destaco: o IPG e ao centro de informática por terem disponibilizado o espaço para a realização do meu estágio e por todo o apoio prestado no decorrer do mesmo.

O supervisor do estágio, Eng.º Carlos Fonseca pela disponibilidade, dedicação e acompanhamento verificados durante o estágio.

O orientador de estágio, Noel Lopes, pela paciência e disponibilidade ao ajudar-me sempre que tinha dúvidas ao longo do estágio.

## Índice

•	CAPITULO 1 – INTRODUÇÃO.....	1
•	1.1 – Enquadramento .....	1
•	1.2 – Objetivos .....	1
•	1.3 – Estrutura do Relatório .....	2
•	CAPITULO 2 – CARACTERIZAÇÃO DA EMPRESA.....	3
•	2.1 – Empresa.....	3
•	2.2 - Principais áreas de Intervenção.....	4
•	CAPITULO 3 – PROJETO DESENVOLVIDO NO ESTÁGIO .....	5
•	3.1 – Definição do Projeto a Desenvolver.....	5
•	3.2 – Objetivos previstos.....	6
•	3.3 - Cronograma de atividades.....	6
•	3.4 – Resultados esperados .....	7
•	3.5 – Análise do Sistema .....	7
•	CAPITULO 4 – DESENVOLVIMENTO DA BASE DE DADOS.....	13
•	CAPITULO 5 – DESENVOLVIMENTO DO SITE .....	16
•	5.1-Candidatura.....	16
•	5.2- Login.....	23
•	5.3- Backoffice.....	25
•	5.4- Informação da candidatura (administrador).....	26
•	5.5- Informação da candidatura (aluno) .....	30
•	CAPITULO 6 – CONCLUSÃO.....	31
•	Bibliografia .....	32

## Índice de Figuras

Figura 1 - Logotipo da empresa, fonte[1].....	3
Figura 2- IPG, fonte [1] .....	3
Figura 3- Casos de Uso .....	7
Figura 4- Diagrama de classes .....	10
Figura 5- Diagrama de Atividades .....	12
Figura 6- Tabela Alunos .....	13
Figura 7- Tabela Candidatura .....	14
Figura 8- Tabela CURSOS.....	14
Figura 9- Tabela Documentos .....	15
Figura 10- Tabela Login .....	15
Figura 11- Página Candidatura .....	16
Figura 12- Validações de campos.....	17
Figura 13- Erros nos campos .....	17
Figura 14- Insert na base de dados .....	18
Figura 15- Gerar password .....	18
Figura 16- FileUpload .....	19
Figura 17- Página Documentos .....	19
Figura 18- Criar Pasta .....	20
Figura 19- Guardar Documento .....	21
Figura 20- FileUpload .....	21
Figura 21- Mostra password .....	22
Figura 22- Página onde é mostrada password.....	22
Figura 23- Enviar Email.....	23
Figura 24- Página Login .....	23
Figura 25- Verificação Email.....	24
Figura 26- Tabela das candidaturas .....	25
Figura 27- DropDownList (Todos) .....	25
Figura 28- DropDownList (Ano Letivo).....	25
Figura 29- DropDownList (Nome) .....	26
Figura 30- DropDownList (Curso).....	26
Figura 31- Aviso documentos em falta.....	26
Figura 32- Página Informação da candidatura (admin).....	27
Figura 33- Criar ficheiro texto .....	27
Figura 34- Criar ZIP .....	28
Figura 35- Update a uma tabela.....	28
Figura 36-Botão para eliminar candidatura .....	29
Figura 37- Confirmação de eliminação .....	29
Figura 38- Código para Eliminar .....	29
Figura 39- informação candidatura (aluno) .....	30

## Índice de Tabelas

Tabela 1- Cronograma de Atividades.....	6
Tabela 2- Requisito 1 .....	8
Tabela 3- Requisito 2 .....	8
Tabela 4- Requisito 3 .....	8
Tabela 5- Requisito 4 .....	9
Tabela 6- Dicionário de dados - Identificação .....	10
Tabela 7- Dicionário de dados - Contacto.....	11
Tabela 8- Dicionário de dados - Cursos .....	11
Tabela 9- Dicionário de dados - Escolha_Curso .....	11
Tabela 10- Dicionário de dados - Situação .....	11
Tabela 11- Dicionário de dados - Candidatura.....	12

# CAPITULO 1 – INTRODUÇÃO

Neste capítulo do relatório será apresentado o enquadramento, os objetivos do projeto desenvolvido e a estrutura do relatório.

## 1.1 – Enquadramento

No âmbito do Curso Técnico Superior Profissional em Desenvolvimento de Aplicações Informáticas, realizei o respetivo estágio no Instituto Politécnico da Guarda (IPG), com a duração de setecentas e cinquenta horas, iniciado no mês de março e concluído no mês de julho.

O estágio curricular é a última etapa na finalização do curso de Desenvolvimento de Aplicações Informáticas, ocasião especial para por em prática os conhecimentos e adquirir novas competências. Assim ao longo de quatro meses tivemos oportunidade de aprofundar conhecimento ao nível de redes de computadores, em particular, numa área de franco desenvolvimento e de grande interesse como é a área de Segurança.

## 1.2 – Objetivos

O projeto desenvolvido em contexto de estágio tinha como objetivo principal uma boa gestão das candidaturas dos Cursos Técnico Superior Profissional (TeSP). Para o bom funcionamento do mesmo, o projeto foi dividido em três partes:

- A primeira é onde o candidato completa a sua candidatura, preenchendo-a devidamente com a ajuda de *Validators* e por fim anexa os documentos pedidos.
- A segunda parte é onde o Aluno faz login e tem acesso à informação da sua candidatura, podendo fazer alterações, anexar os documentos caso não os tenha guardado antes e por fim o candidato poderá eliminar a candidatura se assim o entender.
- A terceira parte é, caso o login pertença a um administrador, leve o utilizador para uma página onde mostre em uma tabela todas as candidaturas registadas, e possa escolher uma. Após o utilizador escolher uma candidatura, aparece uma página semelhante à que aparece ao candidato, com as mesmas opções.

### **1.3 – Estrutura do Relatório**

Este relatório assenta nas atividades desenvolvidas durante o estágio curricular no CI do IPG, onde foram postos em prática conhecimentos adquiridos durante o curso e possibilitou adquirir novos também.

O presente relatório está organizado da seguinte forma: no capítulo 1 é feita a introdução, objetivos e a estrutura do relatório, no capítulo 2 faz-se uma apresentação da empresa onde o estágio foi realizado, no capítulo 3 faz-se uma apresentação do projeto desenvolvido, no capítulo 4 são exibidas as tabelas da base de dados, no capítulo 5 é mostrado o trabalho desenvolvido no estágio e por fim faz-se uma conclusão.

## CAPITULO 2 – CARACTERIZAÇÃO DA EMPRESA

Neste capítulo é apresentada a descrição da empresa que me aceitou no seu espaço.

### 2.1 – Empresa

O Instituto Politécnico da Guarda é uma instituição de ensino superior politécnico pública portuguesa, com sede na cidade da Guarda. Na seguinte imagem, na Figura 1, podemos ver o logotipo da empresa.



Figura 1 - Logotipo da empresa, fonte[1]

O IPG foi um dos primeiros estabelecimentos de ensino superior a ver aprovados os seus estatutos. Desta forma, ficou constituído juridicamente como pessoa coletiva de direito público, dotada de autonomia estatutária, científica, pedagógica, administrativa, financeira, disciplinar e patrimonial. Na Figura 2 podemos ver parte do espaço do IPG.



Figura 2- IPG, fonte [1]

O meu estágio foi realizado no IPG, mais propriamente no Centro de Informática onde têm a missão de gerir toda a infra-estrutura informática relacionada com o Instituto Politécnico da Guarda (IPG) e onde se quer manter de forma significativa e eficaz a maneira como se atende e informa todos os discentes, docentes e funcionários do IPG. A área de atuação compreende o suporte aos utilizadores, a manutenção de equipamentos e serviços e o desenvolvimento de projetos que visem melhorar a infraestrutura instalada.

## **2.2 - Principais áreas de Intervenção**

As principais áreas de intervenção do IPG:

- Formação dos alunos.

As principais áreas de intervenção do Centro de Informática do IPG são:

- Soluções de software específico;
- Conceção e configuração de redes e servidores;
- Manutenção de equipamentos e serviços.

## **CAPITULO 3 – PROJETO DESENVOLVIDO NO ESTÁGIO**

No início do estágio fiquei a conhecer as pessoas que fazem parte do Centro de Informática assim como as tarefas de cada elemento no local de trabalho. Como o objetivo principal do estágio seria por em prática os conhecimentos adquiridos nas aulas, então foi-me atribuído um projeto, criar um site que tem como principal objetivo a gestão de candidaturas, um site onde os alunos/candidatos podem preencher uma candidatura, para os cursos Técnico Superior Profissional (TeSP), onde toda a informação ficará guardada numa base de dados, e do lado da administração poder ter acesso a todas as candidaturas.

Embora o projeto tenha levado algum tempo a ser realizado, devido à sua complexidade, e a algumas dificuldades na realização do mesmo, foi uma boa forma de melhorar as minhas aptidões a trabalhar com o programa *Visual Studio*.

### **3.1 – Definição do Projeto a Desenvolver**

O projeto consiste no desenvolvimento de um site para onde alunos/candidatos podem preencher uma candidatura para os cursos Técnico Superior Profissional, e anexar os documentos pedidos. O site também deve ter uma página para efetuar login, onde, se o login for de um administrador vai levá-lo a uma tabela onde pode ver todas as candidaturas. Se o login não pertencer a um administrador leva o utilizador para uma página onde pode ver toda a informação sobre a sua candidatura.

### 3.2 – Objetivos previstos

Segue-se agora os principais objetivos definidos no início deste projeto.

Objetivos previstos a atingir:

- Site para preencher candidaturas
- Anexar documentos necessários
- Ter um *BackOffice*
- Gestão de Candidaturas;
- Download de ficheiro “zipado” com conteúdo da candidatura.

### 3.3 - Cronograma de atividades

Segue-se agora o cronograma das atividades (Tabela 1) realizadas no estagio relacionados com o projeto. Consoante os vários fatores estes poderão ser ou não ser implementados.

Cronograma de Atividades:

Tabela 1- Cronograma de Atividades

	Março	Abril	Maiο	Junho	Julho
Análise de problemas					
Desenvolvimento da Base de dados					
Desenvolvimento da aplicação WEB					
Instalação/testes da aplicação					

### 3.4 – Resultados esperados

Os resultados esperados com este sistema é conseguir que um candidato faça a sua candidatura de forma simples e completa, onde preenche todos os campos (Ex: Nome, morada, email, telefone, etc.), onde pode anexar os documentos necessários, e mais tarde pode fazer alterações. Mas também deve ser permitido acesso a todas as candidaturas registadas, por parte dos administradores, onde podem alterar campos ou até eliminar a candidatura.

### 3.5 – Análise do Sistema

Neste capítulo é demonstrado algumas das técnicas utilizadas para a análise do sistema, técnicas estas lecionadas em disciplinas como Engenharia de software e Levantamento de Requisitos e Análise Funcional.

### Casos de Uso

A seguinte imagem (Figura 3) é um diagrama de casos de uso em relação ao projeto desenvolvido.

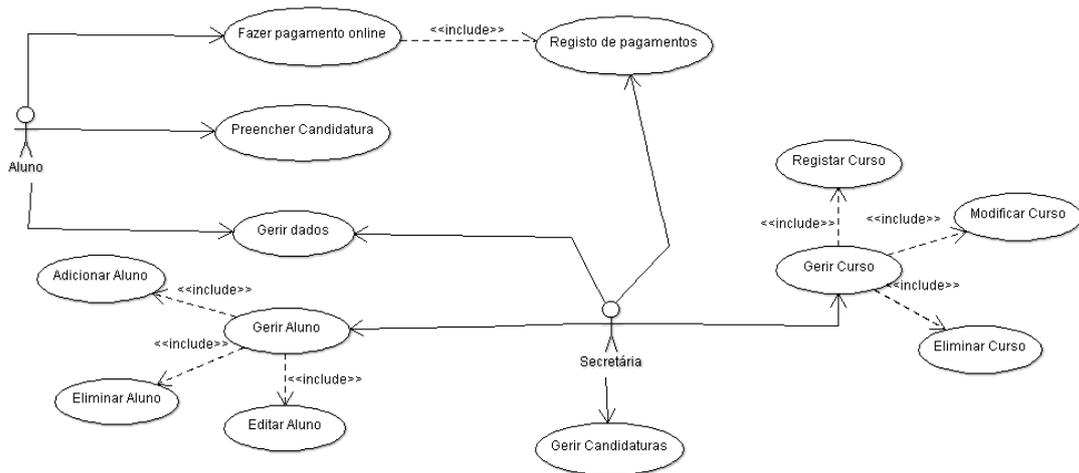


Figura 3- Casos de Uso

## Requisitos Funcionais

O requisito funcional representa o que o software vai fazer, em termos de tarefas e serviços. Desde a Tabela 2 até Tabela 5 temos exemplos de requisitos funcionais para o trabalho desenvolvido.

Tabela 2- Requisito 1

Identificação do requisito	RFO1
Nome do requisito	Candidatura
Características	Candidato
Descrição do requisito	Permite ao candidato preencher uma candidatura para os cursos TeSP e anexar documentos à mesma.

Tabela 3- Requisito 2

Identificação do requisito	RFO2
Nome do requisito	Registrar Candidatura
Características	Candidato
Descrição do requisito	Quando o candidato submeter a candidatura deve ficar registada na base de dados.

Tabela 4- Requisito 3

Identificação do requisito	RFO3
Nome do requisito	Login
Características	Candidato
Descrição do requisito	Assim que o candidato acabar a candidatura vai lhe ser mostrada a senha de acesso ao sistema, sendo o email de login o inserido na candidatura.

Tabela 5- Requisito 4

Identificação do requisito	RFO4
Nome do requisito	Visualizar candidaturas
Características	Administrador/Candidato
Descrição do requisito	Permite a uma pessoa com permissões ter acesso a todas as candidaturas registadas. Se não tiver permissões de administrador é porque é um login registado como candidato que terá acesso à sua própria candidatura.

## Requisitos Não Funcionais

Requisitos não-funcionais são os requisitos relacionados ao uso da aplicação em termos de desempenho, usabilidade, confiabilidade, segurança, disponibilidade, manutenção e tecnologias envolvidas

- Interface:
  - [RNF01]: A aplicação deve ser desenvolvida na linguagem C#;
  - [RNF02]: O utilizador deve ter facilidade de uso nas tarefas a realizar.
  
- Desenvolvimento:
  - [RNF03]: O site deve permitir a gestão de candidaturas;
  - [RNF04]: O utilizador pode fazer download da informação da candidatura;
  
- Segurança:
  - [RNF05]: O site necessita de um sistema de Logins;
  - [RNF06]: Deverá haver 2 tipos de logins (Administrador/Candidato);

## Diagrama de Classes

A seguinte Figura 4 é o diagrama de classes desenvolvido para o projeto de final de curso.

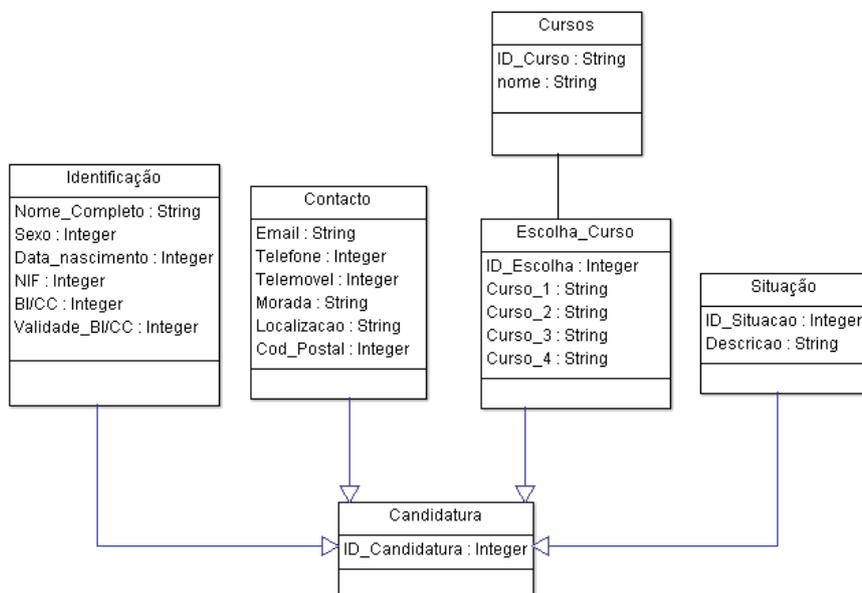


Figura 4- Diagrama de classes

Desde a Tabela 6 até à Tabela 11 temos o dicionário de dados referente à Figura 4.

Tabela 6- Dicionário de dados - Identificação

Entidade: Identificação				
ATRIBUTO	CLASSE	DOMÍNIO	TAMANHO	DESCRIÇÃO
Nome_completo	Composto	Texto	100	
Sexo	Simple	Booleano		
Data_nascimento	Simple	Data		Formato (aaaa/mm/dd)
NIF	Simple	Numérico	9	
BI/CC	Simple	Numérico	8	
Validade_BI/CC	Simple	Data		Formato (aaaa/mm/dd)

Tabela 7- Dicionário de dados - Contacto

Entidade: Contacto				
ATRIBUTO	CLASSE	DOMÍNIO	TAMANHO	DESCRIÇÃO
Email	Simples	Texto	100	
Telefone	Simples	Numérico	9	
Telemovel	Simples	Numérico	9	
Morada	Simples	Texto	100	
Localizacao	Simples	Texto	70	
Cod_Postal	Simples	Numérico	8	

Tabela 8- Dicionário de dados - Cursos

Entidade: Cursos				
ATRIBUTO	CLASSE	DOMÍNIO	TAMANHO	DESCRIÇÃO
ID_Curso	Determinante	Numérico	2	
Nome	Simples	Texto	20	

Tabela 9- Dicionário de dados - Escolha\_Curso

Entidade: Escolha_Curso				
ATRIBUTO	CLASSE	DOMÍNIO	TAMANHO	DESCRIÇÃO
ID_Escolha	Determinante	Numérico	2	
Curso_1	Simples	Numérico	2	Chave estrangeira
Curso_2	Simples	Numérico	2	Chave estrangeira
Curso_3	Simples	Numérico	2	Chave estrangeira
Curso_4	Simples	Numérico	2	Chave estrangeira

Tabela 10- Dicionário de dados - Situação

Entidade: Situação				
ATRIBUTO	CLASSE	DOMÍNIO	TAMANHO	DESCRIÇÃO
ID_Situacao	Determinante	Numérico	2	
Descricao	Simples	Texto	200	

Tabela 11- Dicionário de dados - Candidatura

Entidade: Candidatura				
ATRIBUTO	CLASSE	DOMÍNIO	TAMANHO	DESCRIÇÃO
ID_Candidatura	Determinante	Numérico	2	

### Diagrama de atividades

Um diagrama de atividade é essencialmente um fluxograma, que mostra o fluxo de controle de uma atividade, no seguinte caso (Figura 5) relativamente ao registo de uma candidatura.

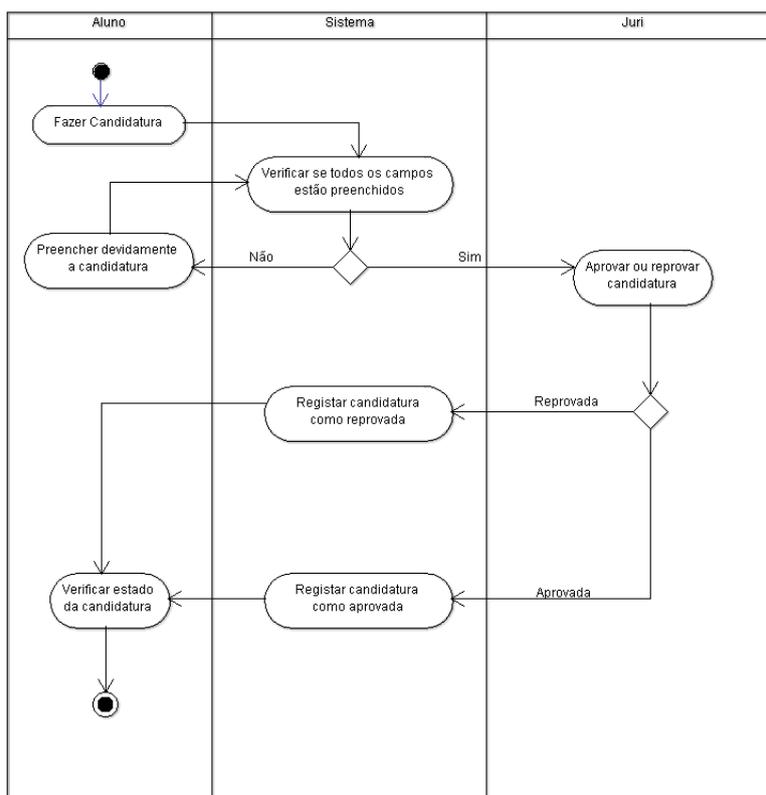


Figura 5- Diagrama de Atividades

## CAPITULO 4 – DESENVOLVIMENTO DA BASE DE DADOS

Para o bom funcionamento deste sistema é necessária uma base de dados onde possa ser armazenada toda a informação sobre a candidatura, e onde mais tarde possa ser recolhida a mesma informação.

Neste capítulo é mostrado todo o trabalho desenvolvido no SQL Server[2] e no SQL Server Management Studio[3]. Sempre que surgia alguma dúvida consultava o w3schools[4] para as esclarecer.

Na Figura 6 podemos ver o “Design” da tabela “ALUNOS” da base de dados, onde fica guardada toda a informação sobre o candidato. Na imagem que se segue podemos também verificar que a coluna da chave primária tem o “(Is identity)” em “yes”, pois sempre que for feito um insert na tabela, automaticamente vai ser inserido um valor nessa mesma coluna, que por defeito começa em um e a cada insert feito soma-se mais 1.

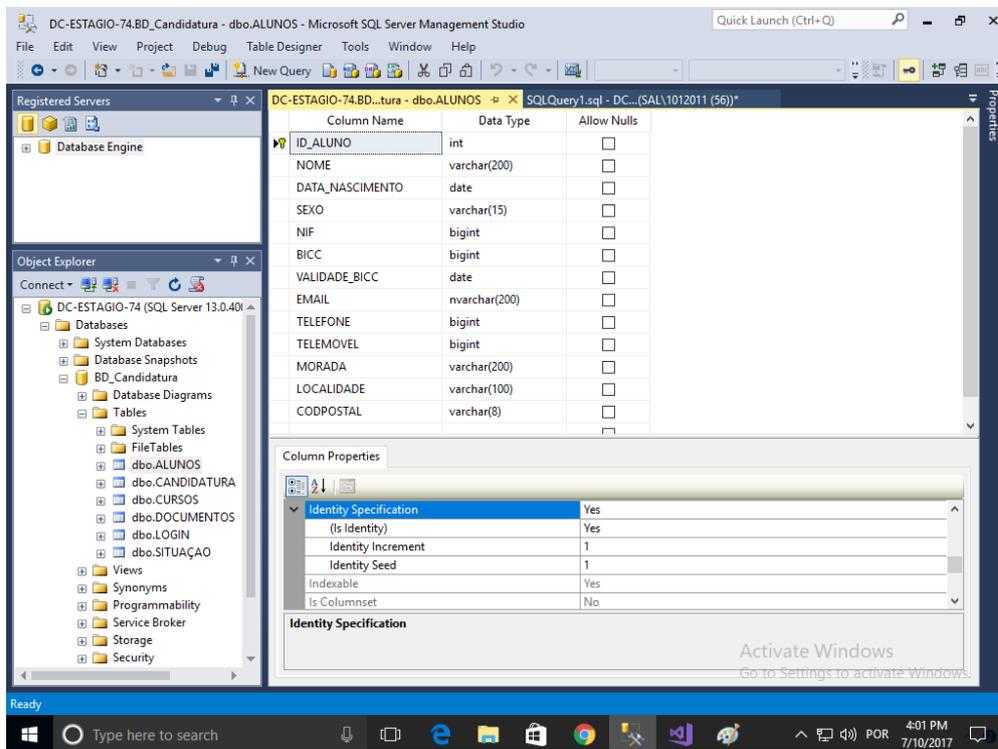


Figura 6- Tabela Alunos

Na Figura 7 podemos ver a tabela “CANDIDATURA” responsável por guardar informação acerca da candidatura como por exemplo a escolha de cursos.

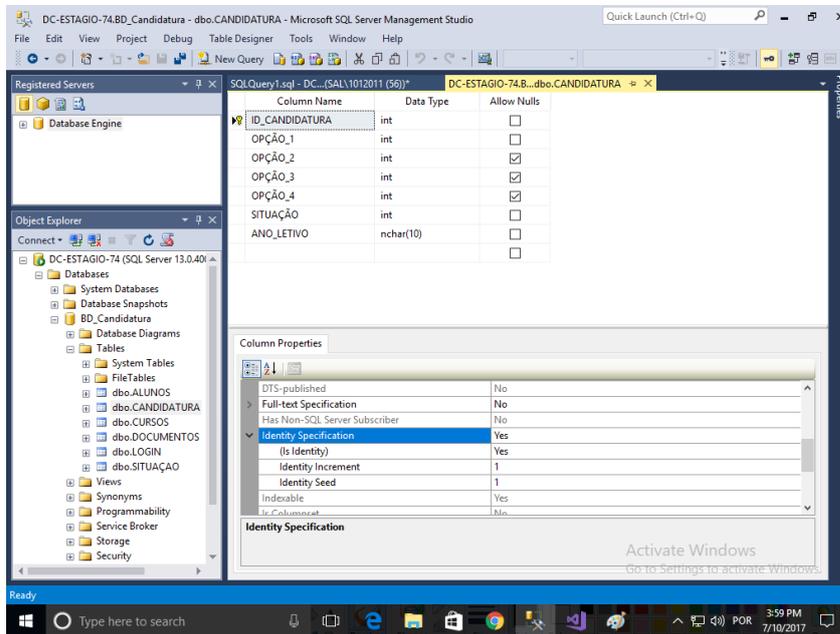


Figura 7- Tabela Candidatura

Na próxima imagem (Figura 8) podemos ver a tabela responsável por armazenar os cursos disponíveis.

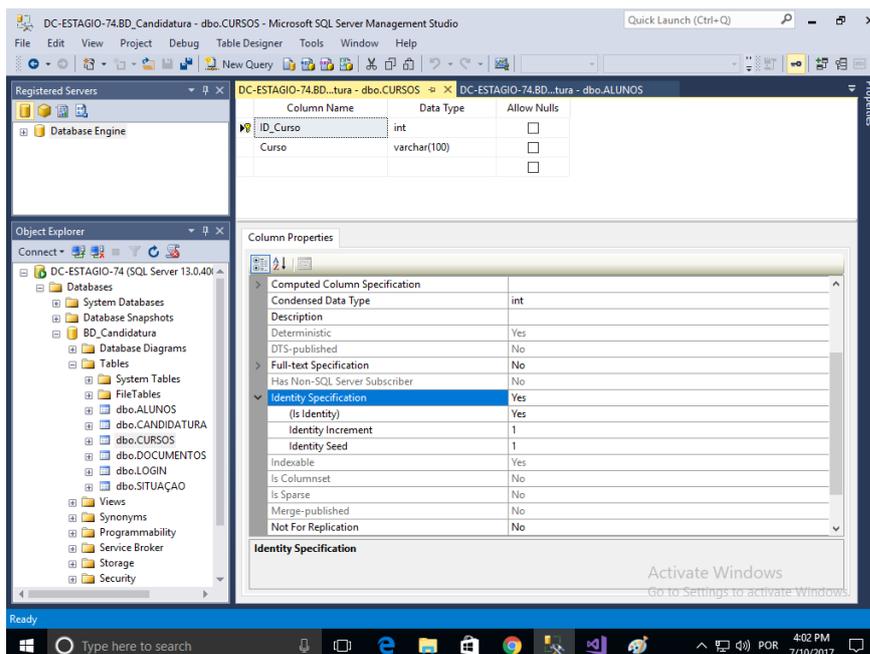


Figura 8- Tabela CURSOS

A tabela responsável por armazenar a informação sobre os documentos anexados está representada na Figura 9.

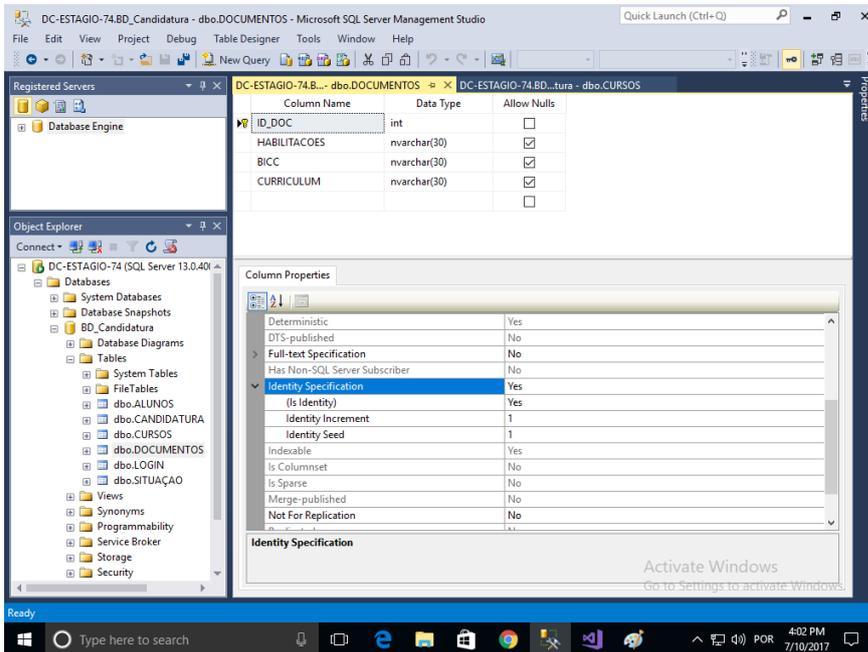


Figura 9- Tabela Documentos

Para o utilizador conseguir fazer um login, a password e o email ficam guardados na tabela “LOGIN” (Figura 10).

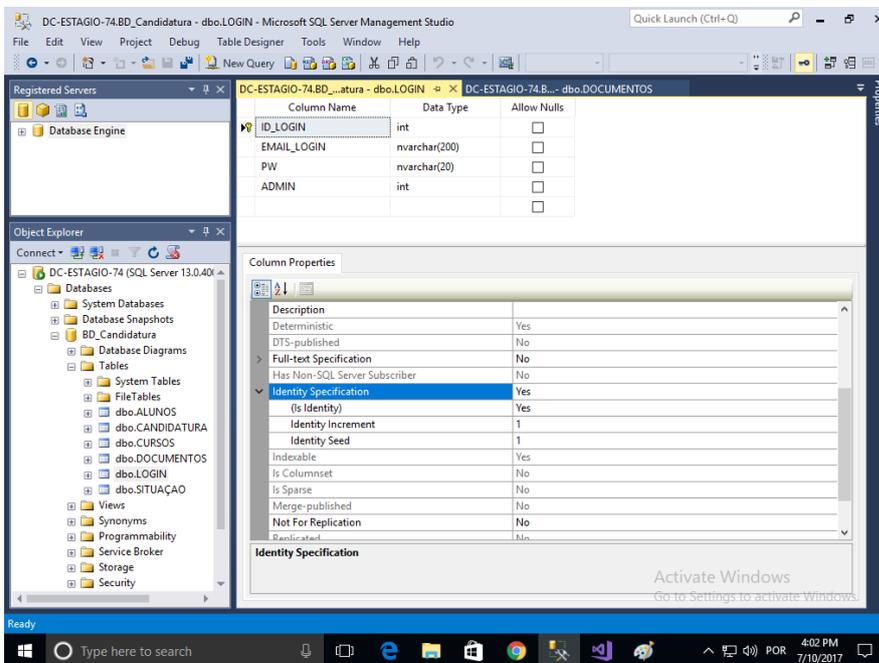


Figura 10- Tabela Login

## CAPITULO 5 – DESENVOLVIMENTO DO SITE

Neste capítulo é apresentado todo o projeto, desde o código necessário e o respetivo interface.

Este trabalho foi desenvolvido recorrendo ao Visual Studio[5]. Mas sempre que tive dificuldades sites de tutoriais como asp.net-tutorials[6], msdn.microsoft[7], e tutorialspoint[8],mas também foram usados fóruns como por exemplo MSDN[9] e stackoverflow[10].

### 5.1-Candidatura

A Figura 11 mostra a página principal do projeto, a página da candidatura onde o candidato preenche todos os campos.

Candidatura

1 IDENTIFICAÇÃO DO(A) CANDIDATO(A)

Nome:

Sexo:  
 Masculino  
 Feminino

Data de Nascimento:  (aaaa/mm/dd)

NIF:

BI/CC:  Válido até:  (aaaa/mm/dd)

2 CONTACTOS

Email:

Activate Windows  
Go to Settings to activate Windows.

Figura 11- Página Candidatura

A seguir criamos as validações dos campos. Na seguinte imagem (Figura 12) podemos ver a validação do campo do nome (txbNome), onde podemos escolher no “ControlToValidate”. O RequiredFieldValidator verifica se o campo está preenchido, enquanto que o RegularExpressionValidator valida o que é inserido, neste caso só podem ser inseridas letras e caracteres especiais como “ ç ” e “ ‘ ”, e pode ter no máximo 200 caracteres.

```
<asp:Label ID="lblNome" runat="server" Text="Nome: "></asp:Label><asp:TextBox ID="txbNome"
runat="server" MaxLength="200" Width="295px"></asp:TextBox>
<asp:RequiredFieldValidator ID="RequiredFieldValidator1" runat="server" ErrorMessage="* é necessário preencher"
ControlToValidate="txbNome" ForeColor="#FF3300" SetFocusOnError="True" ></asp:RequiredFieldValidator>
<asp:RegularExpressionValidator ID="regName" runat="server"
ControlToValidate="txbNome"
ValidationExpression="^[a-zA-Z'.\s'ç]{1,200}"
Text="* Nome inválido" ForeColor="#FF3300" SetFocusOnError="True" />
```

Figura 12- Validações de campos

Na seguinte imagem (Figura 13) podemos ver um exemplo dos erros por validação dos campos.

1. IDENTIFICAÇÃO DO(A) CANDIDATO(A)

Nome:  \* é necessário preencher

Sexo:  
 Masculino  
 Feminino

Data de Nascimento:  (aaaa/mm/dd) \* inserir data válida

NIF:  \* é necessário preencher

BI/CC:  Válido até:  (aaaa/mm/dd) \* é necessário preencher \* é necessário preencher

Figura 13- Erros nos campos

Na Figura 14 podemos ver o código responsável por registar a informação na base de dados. Onde a “string sql” é responsável por o código SQL a ser executado, neste caso para inserir vários parâmetros na tabela ALUNOS, e a “SqlConnection conn” responsável por fazer a ligação à base de dados. Depois temos a parte do “SqlParameter” que cria o parâmetro visto na “string sql” onde vai buscar o seu conteúdo à “txbNome” (TextBox). Por fim executamos o SqlCommand “Inserir\_aluno” com um “ExecuteNonQuery()”.

```
SqlConnection conn = new SqlConnection(System.Configuration.ConfigurationManager
    .ConnectionStrings["BD_CandidaturaConnectionString"].ConnectionString);

string sql = "INSERT INTO ALUNOS (NOME,DATA_NASCIMENTO,SEXO,NIF,BICC,VALIDADE_BICC,EMAIL" +
    ",TELEFONE,TELEMOVEL,MORADA,LOCALIDADE,CODPOSTAL) " +
    "VALUES (@NOME,@DATANASC,@SEXO,@NIF,@BICC,@VALIDADE,@EMAIL,@TELEFONE,@TELEMOVEL,@MORADA,@LOCALIDADE,@CODPOSTAL)";

SqlCommand Inserir_aluno = new SqlCommand(sql, conn);

SqlParameter paramNome = new SqlParameter();
paramNome.ParameterName = "@NOME";
paramNome.Value = txbNome.Text;
Inserir_aluno.Parameters.Add(paramNome);

Inserir_aluno.ExecuteNonQuery();
```

Figura 14- Insert na base de dados

Para o aluno ter acesso à sua candidatura depois de a preencher é necessário um login, onde o email é o inserido anteriormente na candidatura, e a password é gerada aleatoriamente como podemos ver na Figura 15. A string “strPwdchar” serve para definir os possíveis caracteres da password. Este método vai escolher de forma aleatória um caracter até  $i > 7$ , então a password vai ter 7 caracteres.

```
private string GeneratePassword()
{
    string strPwdchar = "abcdefghijklmnopqrstuvwxyz0123456789#+@&$ABCDEFGHIJKLMNPOQRSTUVWXYZ";
    string strPwd = "";
    Random rnd = new Random();
    for (int i = 0; i <= 7; i++)
    {
        int iRandom = rnd.Next(0, strPwdchar.Length - 1);
        strPwd += strPwdchar.Substring(iRandom, 1);
    }
    return strPwd;
}
```

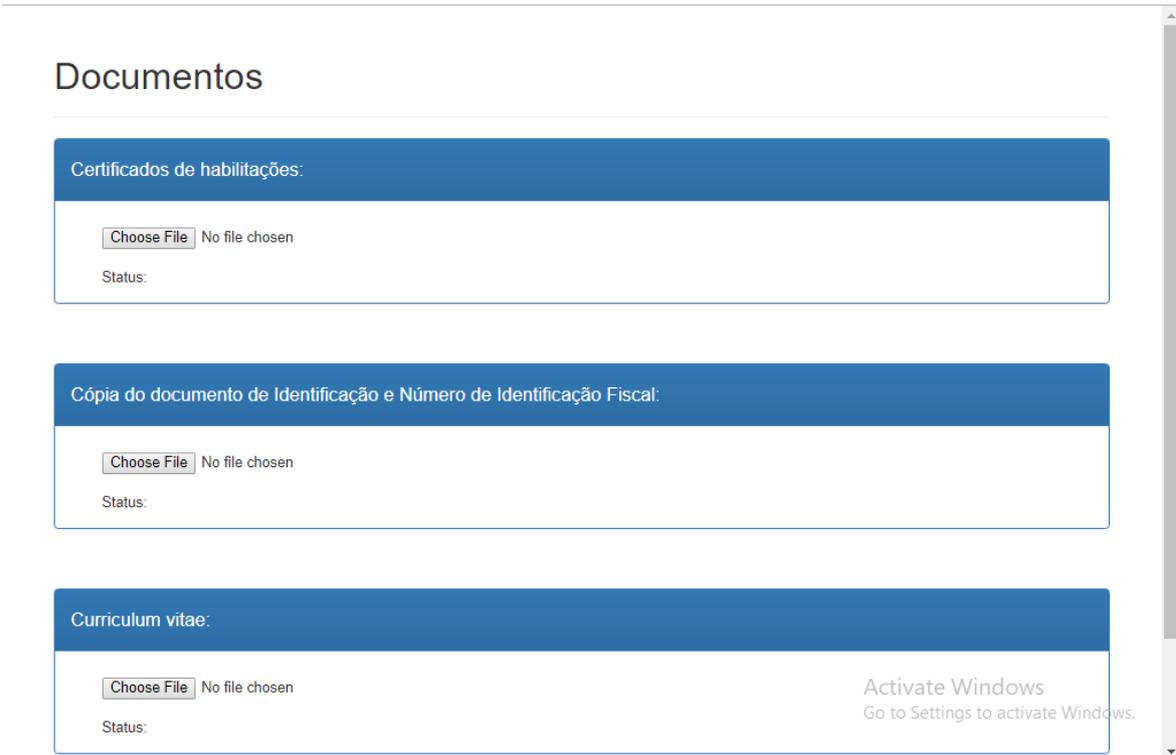
Figura 15- Gerar password

Para o aluno poder anexar os documentos pedidos, tive-mos que adicionar um FileUpload para cada documento pedido. Na Figura 16 podemos ver o exemplo do FileUpload responsável por guardar o certificado de habilitações.

```
<div id="fileuploadhabilitações">  
  <asp:FileUpload ID="FileUploadHabilitações" runat="server" />  
  <br />  
  <asp:Label ID="StatusUploadHabilitações" runat="server" Text="Status:"></asp:Label>  
</div>
```

Figura 16- FileUpload

Na Figura 17 podemos ver em termos de “*design*” a página em que o utilizador pode inserir os documentos.



The screenshot shows a web page titled "Documentos". It contains three distinct sections for document uploads, each with a blue header and a white body. The first section is for "Certificados de habilitações:", the second for "Cópia do documento de Identificação e Número de Identificação Fiscal:", and the third for "Currículo vitae:". Each section includes a "Choose File" button, the text "No file chosen", and a "Status:" label. A Windows watermark is visible in the bottom right corner of the page.

Figura 17- Página Documentos

No pedaço de código mostrado na Figura 18 é responsável por verificar se já existe uma pasta, neste caso com o ID do candidato como nome, e se não existir cria. Para saber qual o id do candidato faz-se um select ao “IDENT\_CURRENT(‘ALUNOS’)” que devolve o id da última linha inserida na tabela ALUNOS.

```
public void EnsureDirectoriesExist()
{
    SqlConnection conn = null;

    conn = new
        SqlConnection(System.Configuration.ConfigurationManager
            .ConnectionStrings["BD_CandidaturaConnectionString"].ConnectionString);

    conn.Open();

    SqlCommand cmd = conn.CreateCommand();
    cmd.CommandText = "SELECT IDENT_CURRENT('ALUNOS')";

    Int32 result = Convert.ToInt32(cmd.ExecuteScalar());

    // if the \pix directory doesn't exist - create it.
    if (!System.IO.Directory.Exists(Server.MapPath("~/") + result))
    {
        System.IO.Directory.CreateDirectory(Server.MapPath("~/") + result);
    }

    conn.Close();
}
```

Figura 18- Criar Pasta

Na Figura 19 podemos ver o código responsável por guardar e registar os documentos. Neste caso o documento é o certificado de habilitações. Se o FileUploadHabilitacoes tiver um ficheiro e se o ficheiro for do mesmo tipo de ficheiro pedido (neste caso PDF), este pedaço de código vai registar o ficheiro na base de dados e vai guardar o ficheiro na diretoria de onde o site está guardado.

```
if (FileUploadHabilitacoes.HasFile)
{
    try
    {
        if (FileUploadHabilitacoes.PostedFile.ContentType == "application/pdf")
        {
            EnsureDirectoriesExist();

            SqlCommand Update_Habilitacoes = new SqlCommand("UPDATE DOCUMENTOS SET " +
                "HABILITACOES = @Nome WHERE ID_DOC = @ID_DOC", conn);

            SqlParameter paramID_DOC = new SqlParameter();
            paramID_DOC.ParameterName = "@ID_DOC";
            paramID_DOC.Value = result;
            Update_Habilitacoes.Parameters.Add(paramID_DOC);

            string cert = "(cert)" + FileUploadHabilitacoes.FileName;
            SqlParameter paramNome_DOC = new SqlParameter();
            paramNome_DOC.ParameterName = "@Nome";
            paramNome_DOC.Value = cert;
            Update_Habilitacoes.Parameters.Add(paramNome_DOC);

            Update_Habilitacoes.ExecuteNonQuery();

            string caminho = ".\\" + result + "\\" + "(cert)" + FileUploadHabilitacoes.FileName;
            String filePath = Server.MapPath(caminho);
            FileUploadHabilitacoes.SaveAs(filePath);
            StatusUploadHabilitacoes.Text = "Upload status: File uploaded!";
        }
        else
        {
            StatusUploadHabilitacoes.Text = "Upload status: Only PDF files are accepted!";
        }
    }
    catch (Exception ex)
    {
        StatusUploadHabilitacoes.Text = "Upload status: The file could not be uploaded. The " +
            "following error occurred: " + ex.Message;
    }
}
```

Figura 19- Guardar Documento

Na Figura 20 podemos ver qual é a aparência do FileUpload, responsável por guardar documentos, neste caso o certificado.



Certificados de habilitações:

Choose File certificad.pdf

Status:

Figura 20- FileUpload

Depois de o candidato completar a candidatura vai-lhe ser apresentada no ecrã a password que lhe foi atribuída. Esta mesma password também vai ser enviada por email para o candidato, onde o email de destino vai ser o email registado na candidatura. No cmd.CommandText podemos ver um select ao Ident\_Current do Login, este código (Figura 21) vai buscar o último ID do login a ser inserido.

```
private void getPW()
{
    SqlConnection conn = new SqlConnection(System.Configuration.ConfigurationManager.
        ConnectionStrings["BD_CandidaturaConnectionString"].ConnectionString);

    conn.Open();

    SqlCommand cmd = conn.CreateCommand();
    cmd.CommandText = "SELECT IDENT_CURRENT('LOGIN');";

    Int32 result = Convert.ToInt32(cmd.ExecuteScalar());

    SqlCommand pw = new SqlCommand("SELECT PW FROM LOGIN WHERE ID_LOGIN = @LOGIN", conn);

    SqlParameter paramID = new SqlParameter();
    paramID.ParameterName = "@LOGIN";
    paramID.Value = result;
    pw.Parameters.Add(paramID);

    lblPW.Text = pw.ExecuteScalar().ToString();
}
```

Figura 21- Mostra password

Exemplo da página que indica a password atribuída ao utilizador no final da candidatura (Figura 22).

## Candidatura finalizada

Guarde a seguinte password, pois vai precisar dela para aceder à sua candidatura no futuro.

q#Esbv2b

\* A password vista, vai ser enviada para o email inserido na candidatura

OK

Activate Windows  
Go to Settings to activate Windows.

Centro de Informática | Instituto Politécnico da Guarda | Av. Dr. Francisco Sá Carneiro N°50 | 6300-559 Guarda | [www.ci.ipg.pt](http://www.ci.ipg.pt) | email: [ci@ipg.pt](mailto:ci@ipg.pt)

Figura 22- Página onde é mostrada password

Na Figura 23 podemos ver o código que irá enviar a password por email aos candidatos. Onde a string “body” vai ser o texto do email. Na 1ª parte da MailMessage inserimos o email que envia , na 2ª parte inserimos o email que recebe, na 3ª parte metemos o titulo do email e a 4ª parte pertence à mensagem que queremos enviar. Para conseguirmos enviar o email na parte no NetworkCredential temos que inserir o email que envia a mensagem e a password do mesmo,

```
protected void btnOK_Click1(object sender, EventArgs e)
{
    string body = "Caro(a) candidato(a) vimos por este meio informá-lo(a) que a sua password para o sistema de gestão de " +
        "candidaturas do Instituto Politécnico da Guarda é:" + "\r\n" + "\r\n" + lblPW.Text + "\r\n" + "\r\n" + "Guarda a password " +
        "pois é necessária se quiser fazer possíveis alterações na sua candidatura.";
    MailMessage o = new MailMessage("tigaspereira@hotmail.com", "tigaspereira16@gmail.com", "Password da Candidatura", body);
    NetworkCredential netCred = new NetworkCredential("email", "pass");
    SmtplibClient smtpobj = new SmtplibClient("smtp.live.com", 587);
    smtpobj.EnableSsl = true;
    smtpobj.Credentials = netCred;
    smtpobj.Send(o);
}
```

Figura 23- Enviar Email

## 5.2- Login

Agora o que se sucede é a página que distingue um aluno de um funcionário ou administrador, é a página do login (Figura 24), onde o utilizador insere o email registado e a password mostrada na ultima página da candidatura ou enviada para o email no caso do candidato.

Please sign in

---

Email:

Password:

Activate Windows  
Go to Settings to activate Windows.

Figura 24- Página Login

Na Figura 25 podemos ver o pedaço de código responsável por detetar uma combinação correta entre email password e uma combinação errada. O “reader” vem de um select à tabela LOGIN onde se o email do id fosse igual ao do email inserido na página verifica a palavra pass do mesmo id, e por fim, compara o login inserido com o login vindo da base de dados.

```
while (reader.Read())
{
    //passo os valores para o objeto cliente
    //que será retornado
    login.EMAIL_LOGIN = reader["EMAIL_LOGIN"].ToString();
    login.PW = reader["PW"].ToString();
    login.ADMIN = Convert.ToInt32(reader["ADMIN"].ToString());
}

if (login.EMAIL_LOGIN == txbEmail.Text)
{
    if (login.PW == txbPW.Text)
    {
        if(login.ADMIN == 1) {
            Response.Redirect("BackOffice.aspx");
        }
        else
        {
            Response.Redirect("CandidaturaAluno.aspx");
        }
    }
    else
    {
        lblPW.Visible = true;
        lblPW.Text = "* Password Errada";
    }
}
else
{
    lblEmail.Visible = true;
    lblEmail.Text = "* Email não existe";
}
```

Figura 25- Verificação Email

### 5.3- Backoffice

Se o login for de um utilizador com permissões de administrador vai ser exibida uma tabela com todas as candidaturas registadas (Figura 26), onde o utilizador vai ter uma DropDownList com as opções de pesquisa como podemos ver desde a Figura 27 até à Figura 30.

The screenshot shows a web browser window with the address bar displaying 'localhost:56911/BackOffice'. The page title is 'Backoffice'. Below the title, there is a search filter section with a dropdown menu set to 'TODOS' and a 'Procurar' button. The main content is a table with the following data:

	ID_ALUNO	NOME	OPÇÃO 1	OPÇÃO 2	OPÇÃO 3	OPÇÃO 4	ANO_LETIVO
Select	3	Goncalves Tiago	Acompanhamento de Crianças e Jovens	Energias Renováveis e Eficiência Energética	Gestão Clínica Administrativa	Design e Fabrico Digital	2017/2018
Select	4	JOAO	Acompanhamento de Crianças e Jovens	Energias Renováveis e Eficiência Energética	Gestão Clínica Administrativa	Design e Fabrico Digital	2018/2019
Select	5	Goncalves Tiago	Acompanhamento de Crianças e Jovens	Energias Renováveis e Eficiência Energética	Gestão Clínica Administrativa	Gestão Clínica Administrativa	2017/2018
Select	6	Goncalves Tiago	Reabilitação Energética e Conservação de Edifícios	Acompanhamento de Crianças e Jovens	Design e Fabrico Digital	SELECIONE O CURSO	2017/2018
Select	7	Goncalves Tiago	Reabilitação Energética e Conservação de Edifícios	Cadastro Predial	Acompanhamento de Crianças e Jovens	SELECIONE O CURSO	2017/2018
Select	8	Goncalves Tiago	Reabilitação Energética e Conservação de Edifícios	Cadastro Predial	Cibersegurança	Acompanhamento de Crianças e Jovens	2017/2018
Select	9	Goncalves Tiago	Cozinha e Produção Alimentar	SELECIONE O CURSO	SELECIONE O CURSO	SELECIONE O CURSO	2017/2018

Below the table, there is a 'Ver informação' button.

Figura 26- Tabela das candidaturas

Escolher por :

Figura 27- DropDownList (Todos)

Escolher por :

Figura 28- DropDownList (Ano Letivo)

Escolher por :

Figura 29- DropDownList (Nome)

Escolher por :

Figura 30- DropDownList (Curso)

## 5.4- Informação da candidatura (administrador)

Depois de o utilizador (administrador) escolher o aluno pretendido, e clicar no botão “Ver informação”, abre uma página com toda a informação sobre o aluno, candidatura e os documentos anexados, onde tem a escolha de poder eliminar ou editar a candidatura. Se houver algum documento em falta aparece um aviso de que os documentos estão em falta (Figura 31) e aparece um botão “Aviso” (Figura 32) que ao ser premido envia um email ao candidato com a informação de que documentos estão em falta. Podemos também fazer download de um ficheiro ZIP, com todo o conteúdo da candidatura ao clicarmos no hyperlink “download”.

Opção 1:

Opção 2:

Opção 3:

Opção 4:

- Titular de curso de ensino secundário ou habilitação legalmente equivalente.

- Obteve aprovação nas provas especialmente adequadas destinadas a avaliar a capacidade para a frequência do ensino superior dos maiores de 23 anos, realizadas, para o curso em causa, nos termos do Decreto -Lei n.º 64/2006, de 21 de março.

- Obteve aprovação em todas as disciplinas dos 10.º e 11.º anos de um curso de ensino secundário, ou habilitação legalmente equivalente, e não tendo concluído o curso de ensino secundário, seja considerado apto através de prova de avaliação de capacidade, a realizar pelo IPG.

Não existe certificado de habilitações

Não existe Cópia do BI/CC

Não existe Curriculum

Figura 31- Aviso documentos em falta

# Candidatura

Editar Eliminar Aviso

Download

Login

Email:  Password:

ID do Aluno:

Nome:

Sexo:

Masculino

Feminino

Data de Nascimento:  (aaaa/mm/dd)

NIF:

BI/CC:  Válido até:  (aaaa/mm/dd)

Activate Windows  
Go to Settings to activate Windows.

Figura 32- Página Informação da candidatura (admin)

Na Figura 33 podemos ver o código responsável por criar um ficheiro texto com toda a informação sobre a candidatura. Toda a informação guardada neste ficheiro vem da base de dados, pois se viesse diretamente da página podia ficar com alguns erros, caso o utilizador pedisse este documento enquanto estava a alterar os campos.

```
string situacao = RadioButtonListSituacao.SelectedValue;
string FilePath = Server.MapPath(@"~/\" + txbID.Text + "/" + txbID.Text + ".txt");
string FileContent = "ID: " + txbID.Text + "\r\n" + "NOME: " + candidatura.NOME + "\r\n" +
"SEXO: " + RadioButtonListSexo.SelectedValue + "\r\n" + "DATA DE NASCIMENTO: " + candidatura.DATA_NASCIMENTO + "\r\n" +
"NIF: " + candidatura.NIF + "\r\n" + "BI/CC: " + candidatura.BICC + "\r\n" + " " + "VALIDADE: " + candidatura.VALIDADE_BICC +
"\r\n" + "EMAIL: " + candidatura.EMAIL + "\r\n" + "TELEFONE: " + candidatura.TELEFONE + " " + "TELEMÓVEL: " + candidatura.TELEMOVEL + "\r\n" +
"MORADA: " + candidatura.MORADA + "\r\n" + "LOCALIDADE: " + candidatura.LOCALIDADE + " " + "CÓDIGO POSTAL: " + candidatura.CODPOSTAL +
"\r\n" + "OPÇÃO 1: " + Session["curso1"].ToString() + "\r\n" + "OPÇÃO 2: " + Session["curso2"].ToString() + "\r\n" +
"OPÇÃO 3: " + Session["curso3"].ToString() + "\r\n" + "OPÇÃO 4: " + Session["curso4"].ToString() + "\r\n" + "SITUAÇÃO: "
+ "\r\n" + situacao;
File.WriteAllText(FilePath, FileContent);
```

Figura 33- Criar ficheiro texto

Na Figura 34 temos o código responsável por criar um documento zip da pasta onde estão armazenados os documentos anexados e o ficheiro texto.

```
private void ZIP()
{
    if ((System.IO.File.Exists(Server.MapPath("~/\" + txbID.Text + ".zip"))))
    {
        System.IO.File.Delete(Server.MapPath("~/\" + txbID.Text + ".zip"));
    }

    string startPath = "C: \\Users\\1012011\\Documents\\Visual Studio 2017\\WebSites\\Matricula\\" + txbID.Text;
    string zipPath = "C: \\Users\\1012011\\Documents\\Visual Studio 2017\\WebSites\\Matricula\\" + txbID.Text + ".zip";
    ZipFile.CreateFromDirectory(startPath, zipPath);
}
```

Figura 34- Criar ZIP

Ao contrário da página da candidatura nesta página não se faz um insert à tabela pois já tem dados guardados, neste caso temos que fazer um update como na Figura 35.

```
SqlCommand Update_Habilitacoes = new SqlCommand("UPDATE DOCUMENTOS SET HABILITACOES = @Nome WHERE ID_DOC = @ID_DOC", conn);

SqlParameter paramID_DOC = new SqlParameter();
paramID_DOC.ParameterName = "@ID_DOC";
paramID_DOC.Value = txbID.Text;
Update_Habilitacoes.Parameters.Add(paramID_DOC);

SqlParameter paramNome_DOC = new SqlParameter();
paramNome_DOC.ParameterName = "@Nome";
paramNome_DOC.Value = "(cert)" + FileUploadHabilitacoes.FileName ;
Update_Habilitacoes.Parameters.Add(paramNome_DOC);

Update_Habilitacoes.ExecuteNonQuery();
```

Figura 35- Update a uma tabela

Visto que esta página tem um botão que permite ao utilizador eliminar a candidatura (Figura 36), antes de eliminar a candidatura temos que confirmar a ação do utilizador (Figura 37), caso ele confirme eliminamos a candidatura e voltamos para a página da tabela, caso não confirme não acontece nada.

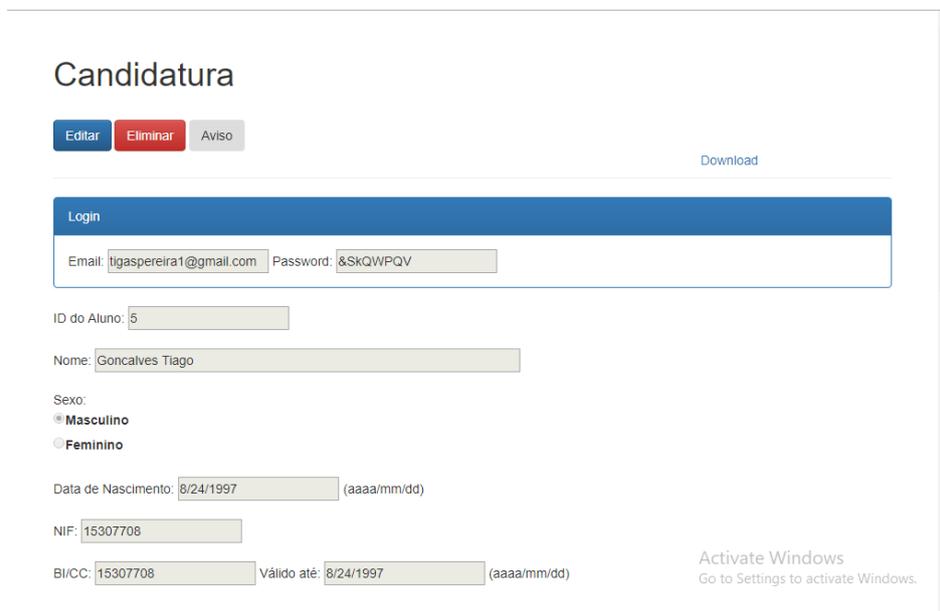


Figura 36-Botão para eliminar candidatura

O código que confirma a eliminação da candidatura está demonstrado na Figura 37.

```
<script type="text/javascript">
    function ConfirmOnDelete() {
        if (confirm("Do you really want to delete?") == true)
            return true;
        else
            return false;
    }
</script>
```

Figura 37- Confirmação de eliminação

Código responsável por eliminar o login do candidato (Figura 38):

```
SqlCommand Eliminar_LOGIN = new SqlCommand("DELETE LOGIN WHERE EMAIL_LOGIN = @EMAIL; ", conn);

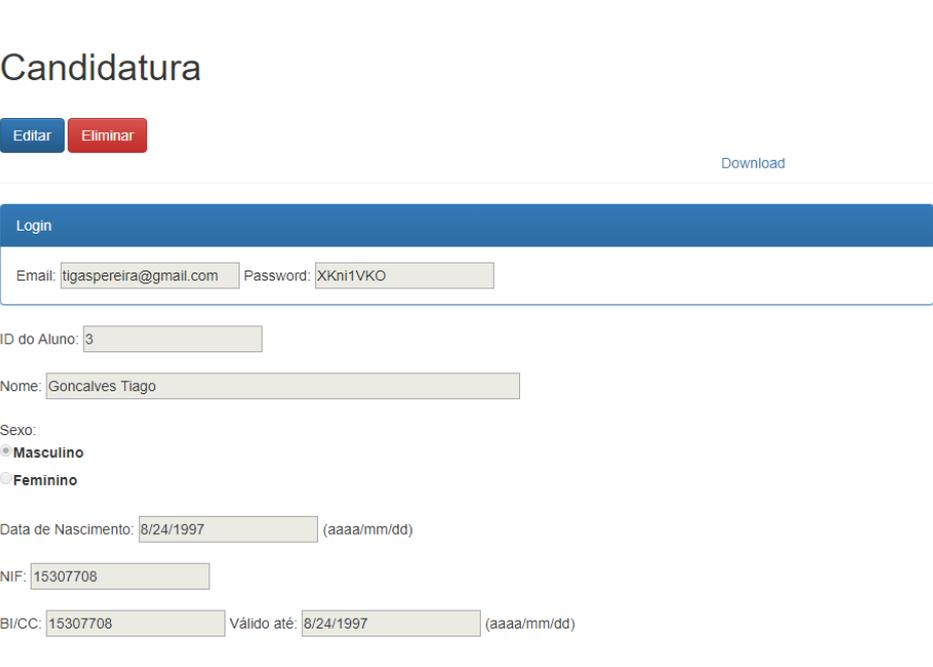
SqlParameter ParamID_LOGIN = new SqlParameter();
ParamID_LOGIN.ParameterName = "@EMAIL";
ParamID_LOGIN.Value = txbEmail.Text;
Eliminar_LOGIN.Parameters.Add(ParamID_LOGIN);

Eliminar_LOGIN.ExecuteNonQuery();
```

Figura 38- Código para Eliminar

## 5.5- Informação da candidatura (aluno)

Esta página (Figura 39), é semelhante à página onde o administrador tem acesso à informação da candidatura, apenas não tem um botão onde possa ser alertado por email dos documentos em falta.



**Candidatura**

[Editar](#) [Eliminar](#) [Download](#)

**Login**

Email:  Password:

ID do Aluno:

Nome:

Sexo:

Masculino

Feminino

Data de Nascimento:  (aaaa/mm/dd)

NIF:

BI/CC:  Válido até:  (aaaa/mm/dd)

Figura 39- informação candidatura (aluno)

## **CAPITULO 6 – CONCLUSÃO**

Estou contente com o estágio realizado, pois consegui aplicar alguns conhecimentos adquiridos durante os anos letivos, e melhorei os meus conhecimentos em linguagem de programação e ter adquirido novos conhecimentos.

Conhecimentos básicos em linguagem de programação C#, ASP.NET e no Visual Studio e estrutura visual do sistema desenvolvido foram alguns dos conhecimentos adquiridos no decorrer das aulas que foram fundamentais para pôr em prática no decorrer do estágio.

O estágio curricular é algo fundamental para todos os estudantes em fase de finalização de curso, pois é o primeiro contato com o mercado de trabalho, e é onde vamos pôr em pratica todo o conteúdo adquirido durante estes dois anos.

Graças a este estágio realizado no Centro de Informática do IPG, pude usufruir destas experiências todas e crescer como trabalhador e como pessoa, tornou-me numa pessoa mais responsável e autónoma, e para mim foi um enorme orgulho poder ter passado por estas experiências todas.

## Bibliografia

- [1] Instituto Politécnico da Guarda, “IPG - Politécnico da Guarda.” [Online]. Available: <http://www.ipg.pt/website/>. [Accessed: 21-Apr-2017].
- [2] SQL Server, “SQL Server 2016 | Microsoft.” [Online]. Available: <https://www.microsoft.com/pt-pt/sql-server/sql-server-2016>. [Accessed: 15-Apr-2017].
- [3] SSMS, “Download SQL Server Management Studio (SSMS) | Microsoft Docs.” [Online]. Available: <https://docs.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms>. [Accessed: 20-Apr-2017].
- [4] W3schools, “W3Schools Online Web Tutorials.” [Online]. Available: <https://www.w3schools.com/>. [Accessed: 21-May-2017].
- [5] Visual Sudio, “Visual Studio IDE, Code Editor, Team Services, & Mobile Center.” [Online]. Available: <https://www.visualstudio.com/>. [Accessed: 10-Mar-2017].
- [6] ASP.NET, “The complete ASP.NET Tutorial.” [Online]. Available: <http://asp.net-tutorials.com/>. [Accessed: 21-Jun-2017].
- [7] Microsoft, “Sample Code - MSDN Examples in C#, VB.NET, C++, JavaScript, F#.” [Online]. Available: <https://code.msdn.microsoft.com/>. [Accessed: 21-Apr-2017].
- [8] Tutorialspoint, “Text and Video Tutorials for UPSC, Civil Services, Banking, Aptitude, Questions, Answers, Explanation, Interview, Entrance, Exams, Solutions, Examples, Online, Quiz, Current Affairs 2017, Aptitude Test, Verbal Ability, General Knowledge, Reasoning, Mock t.” [Online]. Available: <https://www.tutorialspoint.com/>. [Accessed: 21-May-2017].
- [9] Fórum MSDN, “Fóruns do Msdn.” [Online]. Available: <https://social.msdn.microsoft.com/Forums/pt-BR/home>. [Accessed: 21-Apr-2017].
- [10] Stackoverflow, “Stack Overflow - Where Developers Learn, Share, & Build Careers.” [Online]. Available: <https://stackoverflow.com/>. [Accessed: 21-May-2017].