

## **ESCOLA SECUNDÁRIA DE AMARES**

### **Curso Profissional Técnico de Gestão de Equipamentos Informáticos**

# **S.R.B.E.**

**(Sistema de Requisição de Bibliotecas Escolares)**



**Fábio Rafael Gomes Costa      N°5    12ºD**

Amares, 14 de junho de 2021



## **ESCOLA SECUNDÁRIA DE AMARES**

### **Curso Profissional Técnico de Gestão de Equipamentos Informáticos**

## **S.R.B.E.**

**(Sistema de Requisição de Bibliotecas Escolares)**

**Professor Acompanhante: Paulo Martins**

**Fábio Rafael Gomes Costa      N°5    12ºD**

Amares, 14 de junho de 2021



# 1. Agradecimentos

Em primeiro lugar, quero agradecer à minha turma pelo apoio prestado durante estes três anos e aos meus pais por me terem concedido a oportunidade de obter esta formação.

Em segundo lugar, quero agradecer ao meu professor orientador, Paulo Martins, ao Diretor de curso, Fernando Costa, às diretoras de turma, Albertina Afonso e Catarina Veloso, e também aos professores das componentes técnicas, Luís Bernardino, Silvia Barros, Agostinho Abreu, Edgar Soares, Conceição Cardoso e Eduardo Lopes, que foram indispensáveis no desenvolvimento e acréscimo do meu conhecimento e evolução como pessoa nestes últimos três anos.

Quero agradecer também aos restantes professores que, de uma forma direta ou indireta, me acompanharam e me apoiaram nas dificuldades e diferentes etapas desta minha formação.

Quero agradecer às empresas que se disponibilizaram para me receber como local de Formação em Contexto de Trabalho, pois deram-me uma maior abertura a nível de autonomia e visão crítica sobre o trabalho.

Por último, mas não menos importante, sou grato à Direção do Agrupamento de Escolas de Amares, por ter proporcionado a abertura deste curso, caso contrário não chegaria onde cheguei hoje.

A todos, o meu mais sincero obrigado!

## 2. Resumo

Com este relatório damos a conhecer e explicamos a nossa prova de aptidão profissional. A este projeto demos o nome de S.R.B.E. (Sistema de Requisição de Bibliotecas Escolares).

Este consiste numa aplicação destinada a bibliotecas escolares, onde um colaborador da instituição de ensino pode fazer o registo (de alunos, professores, livros, tablets, funcionários, filmes e computadores), a requisição e devolução (de livros, tablets e computadores) e os relatórios (dos registos). O utilizador também pode enviar emails, fazer backup's da base de dados e pedir apoio técnico (tanto da aplicação como de uma equipa de T.I. (da instituição onde for implementado)) sem sair da aplicação.

Como complemento ao projeto, criamos um website onde o utilizador pode sugerir, pedir ajuda, aceder aos manuais de instalação e de utilização da aplicação, fazer o download da mesma no formato executável (.exe) ou código fonte e entrar em contato com a equipa técnica através do e-mail de suporte (suporte.srbe@gmail.com) criado para o efeito.

O objetivo desta aplicação é substituir o sistema de registo arcaico e ultrapassado que é utilizado, por um sistema mais *friendly*, prático, funcional e amigo do ambiente.

### 3. Índice

1. Agradecimentos.....	2
2. Resumo.....	3
3. Índice.....	4
4. Índice de imagens.....	5
5. Introdução.....	7
6. Ferramentas Utilizadas.....	8
Base de dados: Microsoft Access .....	8
IDE: Visual Studio 2019 .....	8
Linguagem: C#.....	9
Criador Website: WIX.....	9
7. Base de dados.....	10
Criação de tabelas.....	10
Relações.....	11
Preenchimento .....	11
8. Aplicação/Formulário.....	12
Janela de Abertura .....	12
Níveis de acessos e segurança .....	13
Menu Login .....	13
Acesso Restrito .....	14
Menu Principal .....	16
9. Website.....	34
10. Conclusão .....	35
11. Web grafia .....	36
12. Anexos.....	37

## 4. Índice de imagens

Figura 1 - Access .....	8
Figura 2 - Visual Studio 2019 .....	8
Figura 3 - C#.....	9
Figura 4 - WIX .....	9
Figura 5 – Ex. Campos da tabela aluno .....	10
Figura 6 – Ex. Campos da tabela livros.....	10
Figura 7- Relações entre Tabelas.....	11
Figura 8 - Ex. Preenchimento .....	11
Figura 9 - Janela de Abertura .....	12
Figura 10 - Código janela de abertura .....	12
Figura 11 - Menu Login .....	13
Figura 12 - Código Clear Login .....	14
Figura 13 - Janela acesso restrito.....	14
Figura 14 - Código acesso restrito.....	15
Figura 15- Novo Registo Funcionário .....	15
Figura 16 - Linha de código registro funcionário.....	16
Figura 17 - Menu Principal.....	16
Figura 18 - Botões canto superior esquerdo .....	16
Figura 19 - Botões canto superior direito .....	17
Figura 20 - Registo .....	17
Figura 21 - Ex. Registo Aluno.....	17
Figura 22 - Ex. Registo Livro.....	17
Figura 23 - Preencha campos obrigatórios .....	18
Figura 24 - Guardado com sucesso.....	18
Figura 25 - Código registo.....	18
Figura 26 - Registo .....	19
Figura 27 - Ex. Pesquisa Aluno .....	19
Figura 28 - Ex. campos de pesquisa .....	19
Figura 29 - Ex. pesquisa completa .....	20
Figura 30 - Código que permite selecionar o que pesquisar.....	20
Figura 31 - Movimentos .....	21
Figura 32 - Ex menu requisição livro .....	21
Figura 33 - Linha de código guardar registo .....	22



Figura 34 - Ex menu devolução livro .....	22
Figura 35 - Linha de código devolução .....	22
Figura 36 - Ex. Menu Relatório.....	23
Figura 37 - Ex. Linha de código de relatório.....	23
Figura 38 - Menu Ferramentas .....	24
Figura 39 - Botão Backup.....	24
Figura 40 - Selecionar local para guardar.....	24
Figura 41 - Linha código Backup .....	25
Figura 42 - Botão apoio técnico .....	25
Figura 43 - formulário fornecido pela S.R.B.E. ....	25
Figura 44 - Linha de Código Apoio técnico .....	26
Figura 45 - Botão E-mail.....	26
Figura 46 - Menu Enviar Email.....	26
Figura 47 - Linha de código para enviar e-mail .....	27
Figura 48 - Definições .....	28
Figura 49 - Definições dados gerais .....	28
Figura 50 - Ex. menu tablet dados.....	29
Figura 51 - Ex. menu tablet tabelas .....	29
Figura 52 - Definições dados administrativos .....	30
Figura 53 - Ex. menu aluno dados.....	30
Figura 54 - Ex. menu aluno tabela.....	30
Figura 55 - Menu definições relatórios.....	31
Figura 56 - Menu definições ferramentas.....	31
Figura 57 - Linha de código restaurar backup.....	32
Figura 58 - Menu ajuda .....	33
Figura 59 - Linha de código direciona para o website .....	33
Figura 60 – QR Code Website.....	37
Figura 61 - Manual Como Instalar .....	37
Figura 62 - Manual Como Utilizar .....	37

## 5. Introdução

O presente projeto corresponde à última fase do Curso Profissional de Técnico de Gestão de Equipamentos Informáticos, constituindo-se como uma Prova de Aptidão Profissional, que tem como principal objetivo abranger todos os conhecimentos adquiridos ao longo dos três anos de formação.

Para o tema, foi escolhido “Criação de um Sistema de Requisição de Bibliotecas Escolares”. Esta escolha deveu-se a uma troca de ideias entre dois alunos da turma 12ºD, Fábio Costa e Luís Freitas, (desenvolvimento da ideia e do plano prático do projeto), enquanto observamos o sistema retrógrado e arcaico, atualmente utilizado pela biblioteca. Chegamos à conclusão que seria uma mais valia para a escola abandonar o sistema utilizado em papel e passar a utilizar um sistema mais informatizado e ecológico.

Para dar corpo ao presente trabalho, foi necessário efetuar pesquisa de informação, em diversas fontes, recolha e tratamento de dados, o que possibilitou efetuar a exposição teórica que este projeto engloba.

## 6. Ferramentas Utilizadas

### Base de dados: Microsoft Access

Microsoft Access, conhecido por MSAccess, é um sistema de gestão de bases de dados da Microsoft, incluído no pacote do Microsoft Office Professional, que combina o Microsoft Jet Database Engine com uma interface gráfica do utilizador.



*Figura 1 - Access*

Desvantagens:

- Tem de se comprar uma licença do Microsoft Office
- Bastante limitado

Vantagens:

- Maior compatibilidade com outros produtos da Microsoft
- Possui mais recursos na Web
- Foi utilizado em contexto de sala de aula

### IDE: Visual Studio 2019

Microsoft Visual Studio é um ambiente de desenvolvimento integrado da Microsoft para desenvolvimento de software especialmente dedicado ao .NET Framework e às linguagens Visual Basic, C, C++, C# e F#



*Figura 2 - Visual Studio 2019*

Desvantagens:

- Número menor de soluções públicas e gratuitas, se comparado a Java

Vantagens:

- O Visual Studio é altamente produtivo
- Possui mais recursos na Web
- É gratuito na versão *Community*
- Foi utilizado em contexto de sala de aula

## Linguagem: C#

C# é uma linguagem de programação desenvolvida pela Microsoft como parte da plataforma .NET. A sua sintaxe orientada a objetos foi baseada no C++, mas inclui muitas influências de outras linguagens de programação, como Object Pascal e, principalmente, Java.



Figura 3 - C#

### Desvantagens:

- A Microsoft deixou de suportar .NET
- Fugas de memória

### Vantagens:

- Maior interatividade com os internautas
- Recurso de programação orientada a objeto
- Fácil de aprender

## Criador Website: WIX

O Wix fornece modelos de sites personalizáveis e um construtor de sites HTML5 orientado a objetos que inclui aplicações, gráficos, galerias de imagens, fontes, vetores, animações e outras opções. Os utilizadores também podem optar por criar seus sites do zero.



Figura 4 - WIX

### Desvantagens:

- Serviços limitados
- Erros e falta de suporte

### Vantagens:

- Produção gratuita
- Tem qualidade e é prático
- Hospedagem
- Foi utilizado em contexto de sala de aula

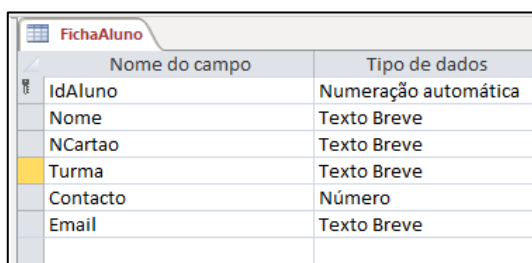
## 7. Base de dados

Para desenvolvermos a aplicação, foi necessária uma base de dados. Começamos por estudar todas as possibilidades existentes no mercado, mas por fim optamos pelo *Microsoft Access*, pois esta é de simples e fácil utilização, tem uma melhor conectividade com o *Visual Studio* e já a tínhamos utilizado em contexto de sala de aula.

### Criação de tabelas

Após decidirmos a escolha de software onde íamos contruir a nossa base de dados, tivemos de estipular os campos da mesma. Para isso realizamos diversas reuniões e pesquisas com o objetivo de encontrar os mais adequados ao projeto. Ultrapassadas essas etapas, chegamos à seguinte estrutura.

Apresentando dois exemplos a seguir:



	Nome do campo	Tipo de dados
PK	IdAluno	Numeração automática
	Nome	Texto Breve
	NCartao	Texto Breve
	Turma	Texto Breve
	Contacto	Número
	Email	Texto Breve

Figura 5 – Ex. Campos da tabela aluno



	Nome do campo	Tipo de dados
PK	IdLivro	Numeração automática
	IdEditora	Número
	Codigo ISBN	Número
	Título	Texto Breve
	Autor	Texto Breve
	Edição	Número
	IDCategoriaLivros	Número

Figura 6 – Ex. Campos da tabela livros

## Relações

Os Relacionamentos entre Tabelas fazem parte da Base de Dados que criam vínculos entre as diferentes informações. Também possibilitam executar diversas ações na Base de Dados.

As relações que nós utilizamos maioritariamente um para muitos, como se pode verificar:

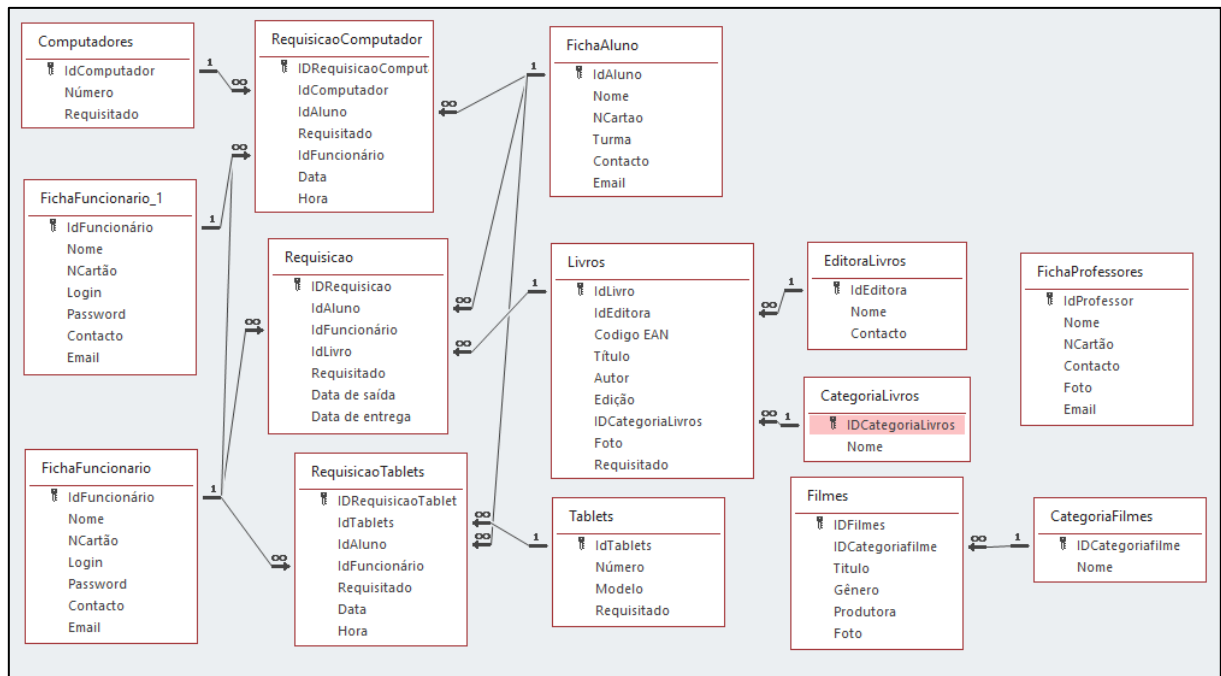


Figura 7- Relações entre Tabelas

## Preenchimento

O preenchimento da base de dados pode ser executado através da mesma, diretamente nas tabelas dos respetivos tópicos.

FichaAluno							
	IdAluno	Nome	NCartao	Turma	Contacto	Email	Clicar para Adicionar
+	1	João Pires	16245	10F	932888112	16245@aeamares.com	
+	2	Rodrigo Pais	16257	11A	916457923	16257@aeamares.com	
+	3	Lúcio Pereira	16256	12D	915231894	16256@aeamares.com	
+	4	Fábio Gomes	16249	10A	932888676	16249@aeamares.com	
+	5	Carlos Rocha	16254	11G	932888989	16249@aeamares.com	
+	6	Miguel Antunes	16265	10C	932884567	16265@aeamares.com	
+	7	Cristina Esperança	16285	12B	932888565	16285@aeamares.com	
+	8	Joana Almeida	16297	11D	916887654	16297@aeamares.com	
+	9	Tiago Sampaio	16284	12D	915657675	16284@aeamares.com	
*	(Novo)						

Figura 8 - Ex. Preenchimento

Também pode ser preenchida através da aplicação.

## 8. Aplicação/Formulário

Para criarmos a aplicação, utilizamos o Visual Studio 2019, como IDE (Ambiente de desenvolvimento integrado), e com a linguagem de programação C#. Estas escolhas deveriam-se à utilização da mesma combinação em anos anteriores no contexto de sala de aula, e tivemos a certeza de que esta opção foi a ideal, após começarmos a trabalhar, pesquisar e explorar as ferramentas e as diversas funcionalidades que oferecem.

### Janela de Abertura

Após abrir a aplicação, aparecerá a janela de abertura (*Loading Screen*) que foi criada para efeito visual, pois o tempo da barra de carregamento é programado.



Figura 9 - Janela de Abertura

Linha de código:

```
private void timer1_Tick(object sender, EventArgs e)
{
    //se a barra concluir o processo vai abrir o novo menu, senão vai fechar a aplicação
    if (progressBar1.Value < 100)
    {
        progressBar1.Value = progressBar1.Value + 2;
    }
    else
    {
        timer1.Enabled = false;
        //novo formulário
        Frm_LoginFuncionario login = new Frm_LoginFuncionario();
        login.Show();

        //fecha o formulário aberto
        this.Visible = false;
    }
}
```

Figura 10 - Código janela de abertura

## Níveis de acessos e segurança

Após estudarmos outras aplicações, chegamos à conclusão que a aplicação precisaria de níveis de acessos e segurança, uma vez que estes servem para proteção de dados (funcionários, alunos, professores e de todos os materiais que a biblioteca oferece), mantendo-a, assim, mais segura e fidedigna.

Os níveis de acessos e segurança que nós criamos são:

- Funcionário
- Administrador

O funcionário tem permissões para criar novos registros de alunos, professores, livros, filmes, tablets e computadores, pode fazer a pesquisa destes, registar as requisições e devoluções (do que a biblioteca tem a oferecer), fazer um relatório do que já foi registado (pode imprimir e exportá-los), tem acesso as ferramentas (onde pode fazer Backup, pedir apoio técnico a equipa de T.I. da instituição, e enviar e-mail) e ainda pode pedir ajuda ao suporte da aplicação.

O administrador tem um acesso total, e a capacidade de criar novos perfis de funcionário, restaurar backups e editar todos os dados.

## Menu Login

Feito o estudo dos níveis de acesso e segurança, começamos a trabalhar mais aprofundadamente nestes.

Assim surgiu o menu login, uma janela onde os funcionários podem usar as suas credenciais para aceder às funcionalidades da aplicação, sendo que, por só administrador pode criar novos funcionários.



*Figura 11 - Menu Login*



Quando criamos o menu login surgiu-nos um problema que, à primeira vista, parecia ter uma fácil resolução. Esse problema relacionava-se com as Caixas de texto (*text box*), do utilizador e da password, que iam buscar a base de dados d primeiro utilizador lá inserido. Para resolver esse problema, criamos um primeiro utilizador “fantasma” que, quando colocado ou quando o programa o vai buscar, permite que a caixa de texto se limpe.

Querenciais fantasma:

- Utilizador fantasma: ‘
- Password fantasma: ‘

Linha de código:

```
1 referência
private void textBox_login_TextChanged(object sender, EventArgs e)
{
    if (textBox_login.Text == "")
    {
        textBox_login.Clear();
    }
}
1 referência
private void textBox_password_TextChanged(object sender, EventArgs e)
{
    if (textBox_password.Text == "")
    {
        textBox_password.Clear();
    }
}
```

Figura 12 - Código Clear Login

## Acesso Restrito

O acesso restrito serve para o utilizador passar de funcionário para administrador.

Isto é: sempre que o funcionário quiser aceder às funcionalidades de administrador, aparecerá uma janela chamada “acesso restrito”, onde este deve colocar uma *password*.



Figura 13 - Janela acesso restrito

Password: admin

Linha de código:

```
1 referência
private void button1_Click(object sender, EventArgs e)
{
    if (textBox1.Text == "admin") //&& textBox1.Text != "func")
    {
        Frm_NovoRegistoFuncionario novologin = new Frm_NovoRegistoFuncionario();
        novologin.Show();
        this.Validate();
        this.Validate();
        this.Visible = false;
    }
    else
    {
        MessageBox.Show("Chave Errada!", "Acesso Negado", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}
```

Figura 14 - Código acesso restrito

### Registo Funcionário

Para o utilizador poder entrar no programa necessita de um registo funcionário. A janela de Registo Funcionário somente pode ser acedida pelo administrador (pois tem de passar pelo acesso restrito).

Dentro da janela registo funcionário o utilizador deve colocar:

- Nome
- Password
- N° Cartão

Figura 15- Novo Registo Funcionário

```

1 referência
private void btn_registar_Click(object sender, EventArgs e)
{
    if (textbox_login.Text == "" && textbox_ncartao.Text == "" && textbox_password.Text == "")
    {
        errorProvider1.SetError(textbox_login, "Preencha todos os campos");
        errorProvider2.SetError(textbox_ncartao, "Preencha todos os campos");
        errorProvider3.SetError(textbox_password, "Preencha todos os campos");
    }
    else
    {
        this.Validate();
        this.fichaFuncionarioBindingSource.EndEdit();
        this.tableAdapterManager.UpdateAll(this.bdpAPDataSet);
        this.Visible = false;
        MessageBox.Show("Registo Guardado Com Sucesso", "Guardado", MessageBoxButtons.OK, MessageBoxIcon.Information);
    }
}

```

Figura 16 - Linha de código registro funcionário

## Menu Principal

O menu principal é a janela onde o utilizador consegue aceder a todas as funcionalidades da aplicação.



Figura 17 - Menu Principal

No canto superior esquerdo, estão os botões das funções que a mesma oferece:

- Registar;
- Pesquisar;
- Movimentos;
- Relatório;
- Ferramentas;
- Definições;
- Ajuda.

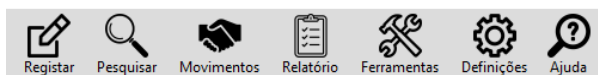


Figura 18 - Botões canto superior esquerdo

No canto superior direito encontram-se os botões:

- Terminar Sessão (ao pressionar o botão terminar sessão voltara ao menu login);
- Minimizar;
- Diminuir/Maximizar;
- Fechar.

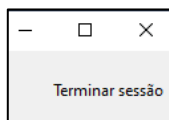


Figura 19 - Botões canto superior direito

## Registo

Ao pressionar o botão Registrar, surgem os vários elementos que podem ser registados, sendo estes:

- Aluno
- Professor
- Livro
- Tablet
- Filme
- Computador.

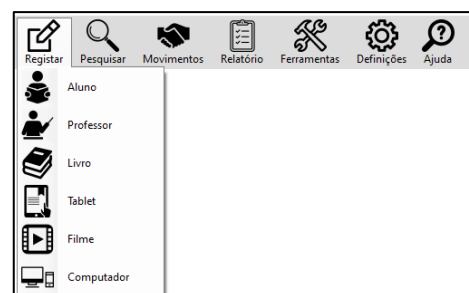


Figura 20 - Registo

Quando escolherem a que querem registar, abrir-se-á uma nova janela com o nome “registo ... (objeto/pessoa que querem registar)”. Como se verifica nos seguintes exemplos:

A imagem mostra a janela "Registo Aluno". No topo, o título é "Registo Aluno". Abaixo, há uma seção "Dados Gerais" com um ícone de três triângulos coloridos (laranja, azul, verde). Os campos de entrada são: "Nome:" (com asterisco), "NºCartão:" (com asterisco), "Turma:", "Contacto:", "Email:" e "Código:". No canto inferior direito, há três botões: "Novo" (destacado em azul), "Guardar" e "Pesquisar".

Figura 21 - Ex. Registo Aluno

A imagem mostra a janela "Registo Livro". No topo, o título é "Registo Livro". Abaixo, há uma seção "Dados Gerais" com o mesmo ícone de triângulos. Os campos de entrada são: "Título:" (com asterisco), "Editora:" (menu suspenso), "Autor:" (com asterisco), "ISBN:" (com asterisco), "Edição:" e "Categoria:" (menu suspenso). No canto inferior direito, há três botões: "Novo", "Guardar" e "Pesquisar".

Figura 22 - Ex. Registo Livro

Deve preencher todos os campos obrigatórios (esta marcada com “\*”) para conseguir executar o registro, caso não o faça receberá a seguinte mensagem.

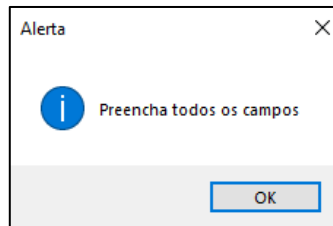


Figura 23 - Preencha campos obrigatórios

Caso o utilizador preencha tudo como deve ser, receberá esta mensagem:

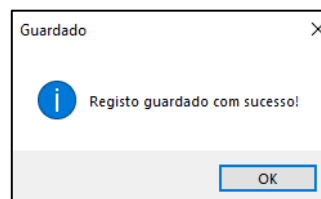


Figura 24 - Guardado com sucesso

Linha de código:

```
1 referência
private void button1_Click(object sender, EventArgs e)
{
    //Campos Obrigatorios
    if (nomeTextBox.Text == "" && nCartaoTextBox.Text == "")
    {
        errorProvider1.SetError(nomeTextBox, "Preencha os campos obrigatorio");
        errorProvider2.SetError(nCartaoTextBox, "Preencha os campos obrigatorio");
        MessageBox.Show("Preencha todos os campos", "Alerta", MessageBoxButtons.OK, MessageBoxIcon.Warning);
    }
    else
    {
        this.Validate();
        this.fichaAlunoBindingSource.EndEdit();
        this.tableAdapterManager.UpdateAll(this.bDPAPDataSet);
        MessageBox.Show("Registo guardado com sucesso!", "Guardado", MessageBoxButtons.OK, MessageBoxIcon.Information);
    }
}
```

Figura 25 - Código registo

## Pesquisa

O botão Pesquisa surgiu com a necessidade que o utilizador (com acessos de funcionário) tinha de aceder às várias informações sobre o repertório de documentos existentes ou outros dados (alunos, livros, etc....)

Ao pressionar o botão Pesquisar, surgem os vários elementos que podem ser pesquisados:

- Alunos
- Professores
- Livros
- Tablets
- Filmes

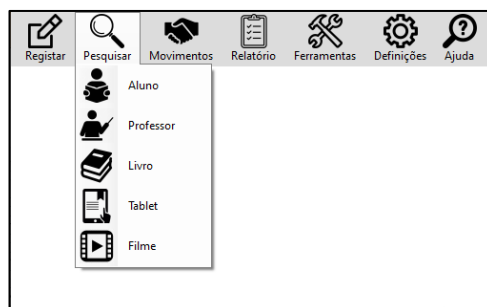


Figura 26 - Registo

Após seleccionar o elemento que se quer encontrar, abrir-se-á uma janela de pesquisa específica, como se segue no seguinte exemplo:

Código	Nome	NºCartão	Turma	Contacto	Email
1	ru filipe	12345	12h	987654321	ae@aeamares.com
2	fafa soso	87654	12c	87654320	aa@aeamares.com
3	gabi gomes	12365	7G	887875664	gab@aeamares.com
*					

Figura 27 - Ex. Pesquisa Aluno

Para pesquisar, o utilizador deve escrever na caixa de texto e seleccionar o tópico sobre o qual quer ter informação, tal como exemplificamos na seguinte imagem:

Figura 28 - Ex. campos de pesquisa

Seguidamente, poderá ver todos os dados do elemento seleccionado de forma limpa e clara, tal como se exemplifica na imagem seguinte:

The screenshot shows a window titled "Pesquisar Aluno". It has a "Geral" tab with a search bar containing "rui" and two radio buttons: "Nome" (selected) and "NºCartão". Below this is a table with columns: Código, Nome, NºCartão, Turma, Contacto, and Email. The first row is selected, showing data for "rui filipe". Below the table is a large grey rectangular area. At the bottom, there is an "Informações" section with input fields for each field from the table, displaying the data for the selected student.

	Código	Nome	NºCartão	Turma	Contacto	Email
▶	1	rui filipe	12345	12h	987654321	ae@aeamares.com
*						

Informações

Código: 1 Turma: 12h

Nome: rui filipe Contacto: 987654321

NºCartao: 12345 Email: ae@aeamares.com

Figura 29 - Ex. pesquisa completa

O seguinte código é o que nos permite seleccionar o que pretendemos pesquisar:

```
1 referência
private void textBox1_TextChanged(object sender, EventArgs e)
{
    //codigo para pesquisar por nome
    if (radiobtn_nome.Checked)
    {
        this.fichaAlunoTableAdapter.FillByPesquisaAlunoNome(bDPAPDataSet.FichaAluno, textBox1.Text);
    }

    //codigo para pesquisar por numero de cartão
    if (radiobtn_ncartao.Checked)
    {
        this.fichaAlunoTableAdapter.FillByPesquisaAluno(bDPAPDataSet.FichaAluno, textBox1.Text);
    }
}
```

Figura 30 - Código que permite seleccionar o que pesquisar

## Movimentos

O menu “movimentos”, com o seu desenvolvimento, apresentou-se como o mais complexo e, sem dúvida, um dos mais importantes, especialmente pelo facto de as funções deste menu serem um dos pilares da aplicação.

Este menu serve para o utilizador registar os livros e computadores que são requisitados, e os que são devolvidos.

De momento, a aplicação só permite aos alunos requisitarem livros e computadores, mas no futuro pretendemos adicionar a requisição de filmes e tablets, e a sua devolução. Também pretendemos permitir aos professores requisitar e devolver filmes, tablets, livros e computadores.

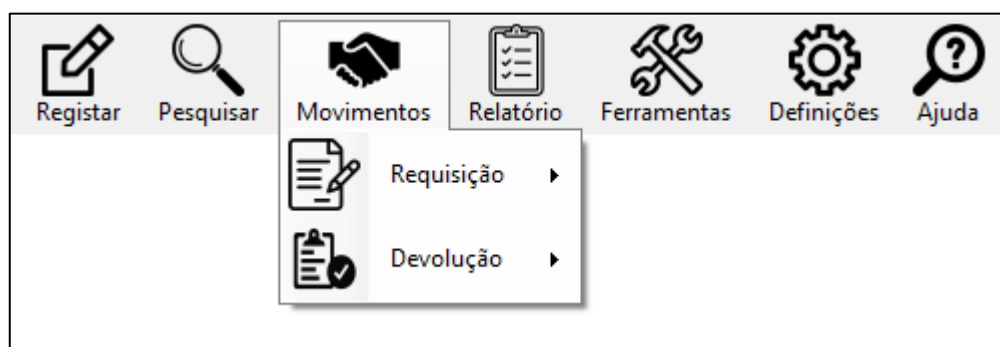


Figura 31 - Movimentos

## Requisição

A requisição é o ato de um aluno poder levar um livro, filme e tablet da biblioteca. Para a instituição ficar com um registo desta ação e para não haver perdas de materiais de propriedade da instituição, criamos o menu Requisição.

Neste menu também damos a oportunidade de registar a utilização dos computadores na biblioteca.

A seguir segue-se um exemplo deste menu:

Figura 32 - Ex menu requisição livro



Linha de código que permite guardar as informações dos registos na base de dados:

```
1 referência
private void btn_guardar_Click(object sender, EventArgs e)
{
    if (this.Validate())
    {
        this.Validate();
        this.requisicaoBindingSource.EndEdit();
        this.tableAdapterManager.UpdateAll(this.bDPAPDataSet);
        MessageBox.Show("Registo Guardado com Sucesso!", "Guardado", MessageBoxButtons.OK, MessageBoxIcon.Information);
    }
    else
    {
        MessageBox.Show("Ocorreu um erro!", "Erro", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}
```

Figura 33 - Linha de código guardar registo

## Entrega

Obrigatoriamente ao criar o menu requisição surgiu o menu devolução. Aqui neste local o utilizador pode concluir os movimentos ao fazer a devolução do item/objeto que já tinha sido requisitado anteriormente.

Código	Aluno	Funcionário	Livro	Requistado	Data de saída	Data de entrega
1	Fábio Costa	SRBE	aaaa	<input type="checkbox"/>	14/06/2021	29/06/2021
2	Luis Freitas	SRBE	aaaa	<input checked="" type="checkbox"/>		
3	Rodrigo Pais	SRBE	tete	<input checked="" type="checkbox"/>		

Figura 34 - Ex menu devolução livro

Linha de código que permite altera o estado do livro para não requisitado:

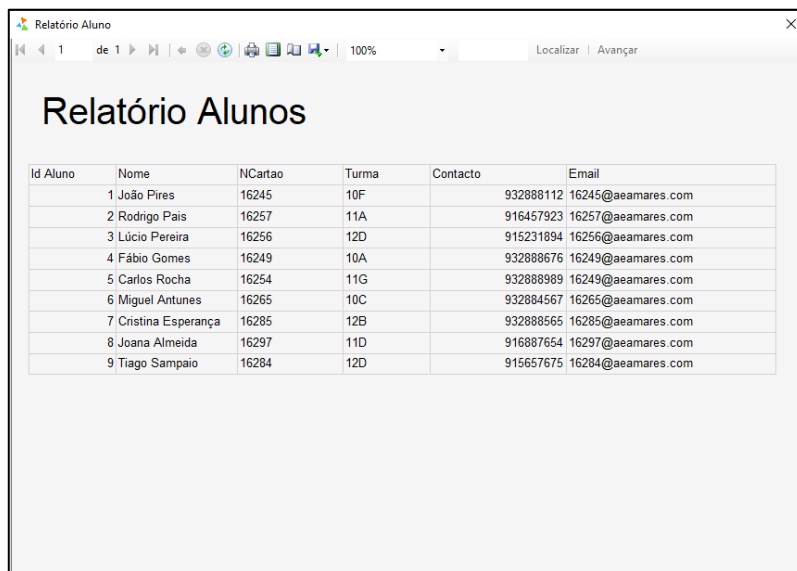
```
1 referência
private void button1_Click_1(object sender, EventArgs e)
{
    this.Validate();
    this.requisicaoBindingSource.EndEdit();
    this.tableAdapterManager.UpdateAll(this.bDPAPDataSet);
    MessageBox.Show("Registo guardado com sucesso", "Guardado", MessageBoxButtons.OK, MessageBoxIcon.Information);
}
```

Figura 35 - Linha de código devolução

## Relatório

O Menu Relatório possibilita ao utilizador com permissões de funcionário aceder, exportar (em formato .pdf, .docx e .xlsx) ou imprimir as informações dos alunos, livros, professores, filmes, tablets, computadores e movimentos.

Esta função é importante, porque permite a concretização de um registro de tudo o que existe e que passa pela aplicação o que, por sua vez, possibilita um maior controlo sobre o que existe e o que é emprestado.



The screenshot shows a window titled 'Relatório Alunos' with a table containing student information. The table has columns for Id Aluno, Nome, NCartao, Turma, Contacto, and Email. There are 9 rows of data.

Id Aluno	Nome	NCartao	Turma	Contacto	Email
1	João Pires	16245	10F	932888112	16245@aeamares.com
2	Rodrigo Pais	16257	11A	916457923	16257@aeamares.com
3	Lúcio Pereira	16256	12D	915231894	16256@aeamares.com
4	Fábio Gomes	16249	10A	932888676	16249@aeamares.com
5	Carlos Rocha	16254	11G	932888989	16249@aeamares.com
6	Miguel Antunes	16265	10C	932884567	16265@aeamares.com
7	Cristina Esperança	16285	12B	932888565	16285@aeamares.com
8	Joana Almeida	16297	11D	916887654	16297@aeamares.com
9	Tiago Sampaio	16284	12D	915657675	16284@aeamares.com

Figura 36 - Ex. Menu Relatório

Linha de código de relatório (Ex. Aluno):

```
7 referências
public partial class Frm_RelatorioAluno : Form
{
    2 referências
    public Frm_RelatorioAluno()
    {
        InitializeComponent();
    }

    1 referência
    private void Frm_RelatorioAluno_Load(object sender, EventArgs e)
    {
        this.FichaAlunoTableAdapter.Fill(this.BDPAPDataSet.FichaAluno);
        this.reportViewer1.RefreshReport();
    }

    1 referência
    private void reportViewer1_Load(object sender, EventArgs e)
    {
    }
}
```

Figura 37 - Ex. Linha de código de relatório

## Ferramenta

As ferramentas foram criadas com o princípio base de o utilizador não precisar de sair da aplicação para executar pequenas funções.

Essas funções podem ser:

- Fazer *backups*
- Chamar apoio técnico
- Enviar *email*

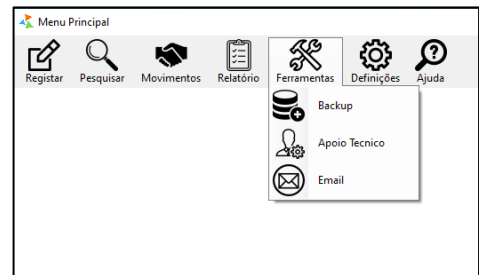


Figura 38 - Menu Ferramentas

## Backup

A ferramenta *Backup* serve para fazer uma cópia de segurança dos seus dados. Assim, mais tarde, podem ser recuperados as traves da ferramenta restaurar que se encontra nas definições.

Para fazer o *Backup*, deve clicar no botão *backup*.

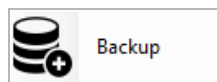


Figura 39 - Botão Backup

Vai abrir uma janela para seleccionar o local onde quer guardar o *backup* e dar o nome ao arquivo. Este ficheiro será guardado no formato BKSrBE (.BkSrbe).

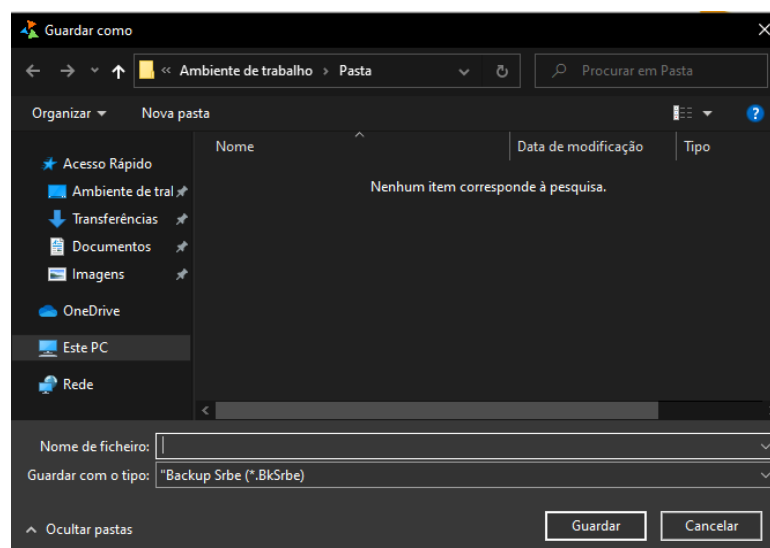


Figura 40 - Selecionar local para guardar

### Linha código *Backup*:

```
private void backupToolStripMenuItem_Click(object sender, EventArgs e)
{
    if (saveFileDialog1.ShowDialog() == DialogResult.OK)
    {
        System.IO.File.Copy(Application.StartupPath.ToString() + "\\BDPAP.mdb", saveFileDialog1.FileName);
        MessageBox.Show("Backup concluído com sucesso!", "Backup", MessageBoxButtons.OK, MessageBoxIcon.Information);
    }
    else
    {
        MessageBox.Show("Backup Cancelado!", "Backup", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}
```

Figura 41 - Linha código Backup

### Apoio Técnico

Esta função da aplicação serve para o utilizador reportar ou encomendar serviços técnicos informáticos. Simplificando, é para quando um equipamento informático avaria e necessita de conserto ou de substituição, servindo como uma linha de contato entre o utilizador e o técnico.



Figura 42 - Botão apoio técnico

Ao pressionar o botão, abrir-se-á um formulário em *google forms* onde o utilizador deve descrever os problemas detetados nos equipamentos. Este formulário é fornecido pela equipa de suporte da S.R.B.E.

A imagem mostra a interface de um formulário online. No topo, há uma barra decorativa com ícones de tecnologia e justiça. Abaixo, o título "Apoio Técnico Escolar" é seguido por uma explicação: "Formulário Fornecedor pela SRBE a Instituição! Este formulário terá como responsável a equipa de T.I da escola." Um asterisco vermelho indica que o campo é obrigatório. O primeiro campo de entrada é rotulado "Data \*" e contém o formato "dd/mm/aaaa" com um ícone de calendário. O segundo campo é rotulado "Funcionária Responsável \*".

Figura 43 - formulário fornecido pela S.R.B.E.

Linha de Código Apoio técnico:

```
private void tecnesaToolStripMenuItem_Click(object sender, EventArgs e)
{
    //encaminha para o website
    System.Diagnostics.Process.Start("https://docs.google.com/forms/d/1rVYL_rom2x-9o1YpcI902hITxy3H-KbGFrcc_JjPaY/edit");
}
```

Figura 44 - Linha de Código Apoio técnico

## E-mail

A ferramenta de *Email* tem por objetivo facilitar o trabalho do funcionário.

Deste modo, o mesmo não tem necessidade de sair da aplicação para mandar um *email*, de devolução (por exemplo).

Para enviar um *email* deve clicar no botão “e-mail”.



Figura 45 - Botão E-mail

Isso vai abrir o Menu Enviar *Email*, onde deve preencher todos os campos e pressionar o botão enviar.

Uma janela de diálogo intitulada "Enviar Email" com uma barra de título cinza e botões de minimizar, maximizar e fechar. O conteúdo da janela tem um fundo cinza claro. No canto superior esquerdo, há um logotipo composto por três triângulos (laranja, azul e verde) e a palavra "EMAIL" em letras maiúsculas e negritadas. Abaixo, há três campos de entrada: "Para:" com um campo de texto curto, "Assunto:" com um campo de texto médio e "Mensagem:" com um campo de texto grande. No canto inferior direito, há um botão "Enviar" com uma borda dupla.

Figura 46 - Menu Enviar Email

## Linha de código para enviar e-mail:

```
1 referência
private void button2_Click(object sender, EventArgs e)
{
    string para, remetente, pass, messagebody;
    MailMessage mensagem = new MailMessage();
    para = textBox_para.Text;
    remetente = "suporte.srbe@gmail.com";
    pass = "srbesrbe";
    messagebody = textBox_mensagem.Text;
    mensagem.To.Add(para);
    mensagem.From = new MailAddress(remetente);
    mensagem.Body = messagebody;
    mensagem.Subject = textBox_assunto.Text;
    mensagem.IsBodyHtml = true;
    SmtpClient smtp = new SmtpClient("smtp.gmail.com");
    smtp.EnableSsl = true;
    smtp.Port = 587;
    smtp.DeliveryMethod = SmtpDeliveryMethod.Network;
    smtp.Credentials = new NetworkCredential(remetente, pass);

    try
    {
        smtp.Send(mensagem);
        DialogResult code = MessageBox.Show("Email Enviado Com Sucesso", "Email Enviado", MessageBoxButtons.OK, MessageBoxIcon.Information);

        if (code == DialogResult.OK)
        {
            textBox_assunto.Clear(); textBox_mensagem.Clear(); textBox_para.Clear();
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message);
    }
}
```

Figura 47 - Linha de código para enviar e-mail

## Definições

O menu “definições” é o local onde todos os dados podem ser apagados, editados, criados e adicionados. É aqui também onde se pode fazer o restauro dos *Backup*’s.

Para o utilizador aceder a estas funções e informações, necessita da permissão do administrador, pois para este menu se abrir, é obrigatório passar por um menu de acesso restrito.

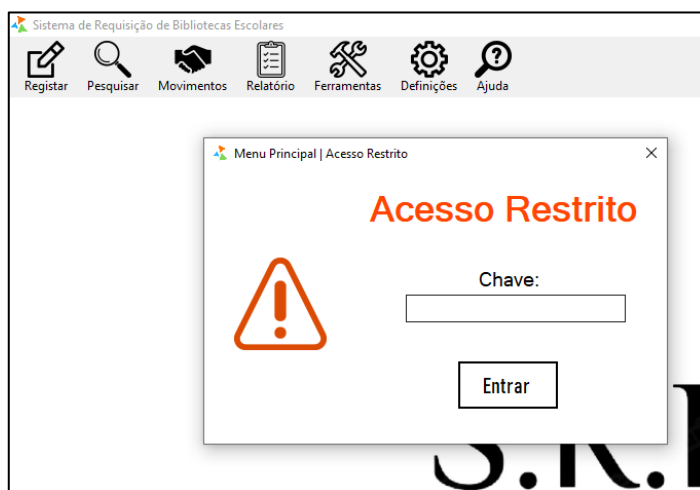


Figura 48 - Definições

## Dados gerais

Nos dados gerais o administrador pode seleccionar os seguintes menus:

- Menu livro
- Menu tablet
- Menu computador
- Menu categoria (filmes e livros)
- Menu editora



Figura 49 - Definições dados gerais

Nestes menus, o administrador, pode editar, deletar e adicionar dados, assim sendo neste menu o administrador tem acesso total sobre todos os menus acima referidos. A seguir seguem-se dois exemplos destas funções:

*Figura 50 - Ex. menu tablet dados*

	Id Tablets	Número	Modelo
▶	2	1	
	3	2	
	4	3	
	5	4	
	6	5	
	7	6	
	8	7	

*Figura 51 - Ex. menu tablet tabelas*

### **Dados Administrativos**

A semelhança dos dados gerais, os dados administrativos funcionam exatamente da mesma maneira, só que com os seguintes menus:

- Menu Funcionário
- Menu aluno
- Menu Professor





Figura 52 - Definições dados administrativos

Seguem-se os exemplos de um menu deste tipo de dados:

Figura 53 - Ex. menu aluno dados

IdAluno	Nome	NºCartao	Turma	Email
1	Luis Freitas	16250	12D	16250@aeamares.com
2	Fábio Costa	16409	11A	16409@aeamares.com
3	Rodrigo Pais	16458	10F	16458@aeamares.com
-1	Tomas D. Moreira	23456	12p	23456@aeamares.com
-2	jose costa	34567	7h	34567@aeamares.com
-3	Rita Amoreiras	45678	9f	45678@gmail.com
-4	Lara Law	12349	11k	12349@hotmail.pt
-5	法比奧	SN	intercambio	法比奧@gmail.com
-7				

Figura 54 - Ex. menu aluno tabela

## Relatórios

Os relatórios oferecem exatamente as mesmas funções que o menu relatório já anteriormente referido. Os relatórios só estão inseridos nas definições, única e exclusivamente para facilitar o manuseamento da aplicação pela parte do administrador assim não o obrigando a sair das definições para aceder ao menu relatórios.



Figura 55 - Menu definições relatórios

## Ferramentas

As ferramentas no menu definições tem uma funcionalidade única e também bastante importante pois é onde se encontra a capacidade de o administrador restaurar um *backup* feito anteriormente. Também é possível fazer por aqui um novo *backup* exatamente com a mesma funcionalidade que tem no menu ferramentas.



Figura 56 - Menu definições ferramentas

Linha de código que permite restaurar *backup*:

```
1 referência
private void btn_menurestaurar_Click(object sender, EventArgs e)
{
    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        if (System.IO.File.Exists(Application.StartupPath.ToString() + "\\BDPAP.mdb"))
        {
            System.IO.File.Delete(Application.StartupPath.ToString() + "\\BDPAP.mdb");
        }
        System.IO.File.Copy(openFileDialog1.FileName, Application.StartupPath.ToString() + "\\BDPAP.mdb");
        MessageBox.Show("Restaurado concluído com sucesso!", "Backup", MessageBoxButtons.OK, MessageBoxIcon.Information);
    }
    else
    {
        MessageBox.Show("Restaurado Cancelado!", "Backup", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}
```

Figura 57 - Linha de código restaurar backup

## Ajuda

O menu ajuda existe para dar um maior apoio ao utilizador e para existir uma maior comunicação entre este e a equipa de suporte da aplicação.

Neste menu o utilizador pode seleccionar um dos seguintes tópicos e serra direcionado ao website de suporte:

- Website
- Contato
- Sugestões
- Ajuda



Figura 58 - Menu ajuda

A seguinte linha de código é o que permite que o utilizador seja direcionado para o website:

```
1 referência
private void websiteToolStripMenuItem_Click(object sender, EventArgs e)
{
    //direciona para o website
    System.Diagnostics.Process.Start("https://suportesrbe.wixsite.com/suportesrbe");
}
```

Figura 59 - Linha de código direciona para o website

## 9. Website

Enquanto planejávamos e desenvolvíamos a nossa aplicação sentimos que esta poderia estar á ficar levemente confusa e complexa, então para podermos elucidar o utilizador decidimos criar um website onde o nosso publico alvo pode recorrer aos manuais de utilização e instalação da aplicação, pode tirar duvidas, dar sugestões, entrar em contato connosco, saber mais sobre nos e fazer o *download* da aplicação nos formatos de instalador executável (.exe) ou código fonte.

O nosso website é composto pelas seguintes páginas:

- **Inicio** (página de rosto)  
<https://suportesrbe.wixsite.com/suportesrbe>
- **Ajuda** (página onde temos os manuais de utilização e de instalação também tem uma caixa de diálogo onde o utilizador pode mandar as suas dúvidas diretamente para o nosso e-mail de suporte)  
<https://suportesrbe.wixsite.com/suportesrbe/ajuda>
- **Sugestões** (nesta página a semelhança da página ajuda aqui também temos uma caixa de diálogo onde o utilizador deve mandar as suas sugestões)  
<https://suportesrbe.wixsite.com/suportesrbe/sugest%C3%A3o>
- **Contacto** (na página contactos nos apresentamo-nos etemos outra caixa de diálogo para quem quiser conversar diretamente sobre a aplicação)  
<https://suportesrbe.wixsite.com/suportesrbe/contato>
- **Download** (na página download fornece um download da aplicação, de forma segura sobre o formato de executável (.exe) e sobre o formato de código fonte.)  
<https://suportesrbe.wixsite.com/suportesrbe/download>

## 10. Conclusão

Com o desenvolvimento desta prova de aptidão profissional (PAP) tivemos a oportunidade de colocar à prova e enriquecer os conhecimentos adquiridos nos últimos três anos de curso, nomeadamente no desenvolvimento de bases de dados, utilização de linguagem de código (C#), criação de aplicações e criação de *Websites*.

Desde já deixamos aqui claro que este relatório não será um ponto final no projeto, pois pretendemos melhorar tanto a base de dados como a aplicação e o website de suporte. Temos diversas ideias para melhorar o funcionamento da aplicação, como por exemplo: dar autonomia ao aluno para o próprio fazer requisição sem o auxílio de um funcionário.

Esta prova demonstrou-se mais do que um projeto escolar, pois passou a ser algo pessoal que levamos para a nossa vida, por ter marcado o final do nosso percurso no ensino secundário. Este curso ofereceu-nos muito durante três anos fundamentais, tanto para o nosso desenvolvimento cognitivo e como para o desenvolvimento humano.

## 11. Web grafia

Microsoft. (s.d.). *Access*. Obtido de Microsoft: <https://www.microsoft.com/pt-pt/microsoft-365/access>

Microsoft. (s.d.). *Forums*. Obtido de Microsoft: <https://social.msdn.microsoft.com/Forums/pt-BR/home>

Microsoft. (s.d.). *Visual Studio*. Obtido de Microsoft: <https://visualstudio.microsoft.com/pt-br/vs/features/net-development/>

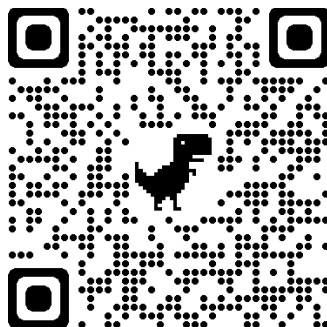
Pacievitch, Y. (s.d.). *c-sharp*. Obtido de infoescola: <https://www.infoescola.com/informatica/c-sharp/>

Pereira, I. P. (23 de 08 de 2016). *Desenvolvimento C#*. Obtido em 13 de 12 de 2020, de Youtube: <https://www.youtube.com/playlist?list=PL92DBFA5B749AA0EC>

## 12. Anexos

QR Code que direciona para Capa do Website

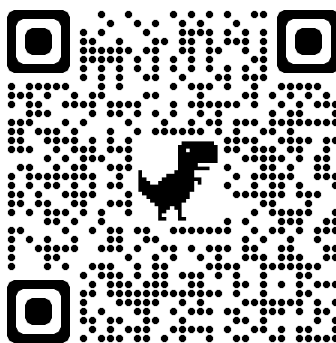
<https://suportesrbe.wixsite.com/suportesrbe>



*Figura 60 – QR Code Website*

QR Code que direciona para o manual “Como Instalar” que esta alojado no Website

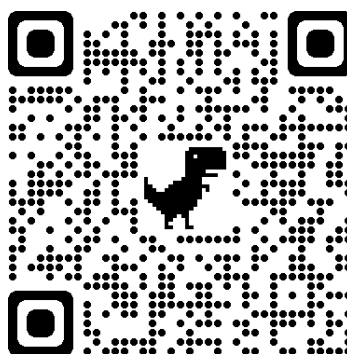
<https://suportesrbe.wixsite.com/suportesrbe/como-instalar>



*Figura 61 - Manual Como Instalar*

QR Code que direciona para o manual “Como Utilizar” que esta alojado no Website

<https://suportesrbe.wixsite.com/suportesrbe/como-utilizar>



*Figura 62 - Manual Como Utilizar*