

ESCOLA SECUNDÁRIA DE PINHEIRO E ROSA

ANO LECTIVO DE 2013/2014



CURSO PROFISSIONAL DE TÉCNICO DE GESTÃO E PROGRAMAÇÃO DE
SISTEMAS INFORMÁTICOS - 3º ANO

Prova de Aptidão Profissional (PAP)

Software de uma
Biblioteca



Cyrill Rosa Brito

Índice

Introdução	2
Cronograma de Actividades	3
Descrição do sistema	4
Conhecimentos aplicados.....	5
Base de dados.....	6
Diagrama ER.....	6
Tabelas	7
CBClass.....	8
Introdução	8
Funções Gerais.....	9
Guardar Definições	9
Carregar Definições	9
Funções MySQL.....	10
Criação de uma Pesquisa	10
Controlo Pesquisa Estrangeira.....	11
Introdução	11
Propriedades.....	12
Controlo de Pesquisa Local.....	13
Introdução	13
Propriedades.....	13
Formulários	14
Introdução	14
Pesquisar.....	15
Colunas não estrangeiras.....	15
Colunas estrangeiras.....	16
Funções.....	18
Verificação de dados.....	19
Caracterização do produto final.....	20
Meios de Divulgação.....	21
Conclusão	22

Introdução

A Prova de Aptidão Profissional, mais conhecida como PAP, consiste na apresentação, perante um Júri, de um projecto que representa os conhecimentos adquiridos durante os 3 anos de escolaridade no curso profissional. É completado de um relatório sobre o projecto que também será avaliado pelos júris.

No curso de técnico de gestão e programação de sistemas informáticos os alunos têm como costume apresentar um programa ou um site, com ligação a base de dados, ou não. Como uma grande parte dos três anos foi dedicada à programação de programas e de base de dados eu decidi então criar um programa/software ligado a uma base de dados sobre uma biblioteca.

No projecto pretendo incorporar o número máximo de conhecimentos adquiridos no curso, e muitos outros adquiridos através de pesquisas e de experiência própria.

Cronograma de Actividades

Actividades	Tarefas	Calendarização
Escolha do projecto	Seleção de ideias e de objectivos para do projecto	30 Outubro
Pesquisa	Recolha de informação sobre o funcionamento de uma biblioteca.	06 Novembro
Plano do Projecto	Preenchimento do documento disponível no Moodle, relativo ao Plano do Projecto,	Até 07 Novembro
Forms	Criação de esboços dos forms para o programa.	Até 14 Novembro
Diagrama ER	Criação dos diagramas E-R	Até 21 Novembro
Documento/relatório sobre o modelo E-R	Elaboração do documento/relatório sobre o modelo E-R do sistema a tratar, modelação da informação e criação da base de dados relacional	Até 06 Dezembro
Apresentação das bases de dados	Apresentação das bases de dados necessárias para os projectos	Até 12 e 13 Dezembro
Relembrar conhecimentos	Criação de alguns projectos de teste para relembrar matéria dada.	Até 15 Janeiro
Sistema de Login	Elaboração do sistema de login em c#	Até 22 Janeiro
Forms dos livros	Elaboração dos forms que mostra/edita os livros e os seus dados	Até 30 Janeiro
Forms dos leitores	Elaboração dos forms que mostra/edita os leitores e os seus dados	Até 12 Fevereiro
Forms para requisitar	Elaboração dos forms que mostra/requisitar livros e os seus dados	Até 19 Fevereiro
Aquisição de novos conhecimentos	Aquisição de novos conhecimentos que permitirão simplificar o código e melhorar o resultado final.	Até 01 Abril
Criação de uma class	Construção de uma class que permitira simplificar e remover redundâncias no código. Construção de controlos personalizados para o projecto	Até 01 Maio
Remodelação do código	Devido á construção da class foi possível modificar grande parte do código já escrito.	Até 10 Maio
Forms para autores	Elaboração dos forms que mostra/edita os autores e os seus dados	Até 17 Maio
Forms para categorias	Elaboração dos forms que mostra/edita as categorias dos livros	Até 24 Maio
Forms para editoras	Elaboração dos forms que mostra/edita as editoras e os seus dados	Até 01 Junho
Novas funcionalidades	Criação de funcionalidades que fazem o programa mais completo e utilizável	Até 10 Junho
Relataria	Elaboração do relatório	Até 15 Junho

Descrição do sistema

O software/programa que desenvolvi tem como o objectivo permitir aos funcionários da biblioteca gerir todos os leitores, livros, autores, editoras, categorias de livros e por fim poder requisitar livros a leitores.

O programe permitira adicionar novos livros, editar livros já existentes e verificar os dados de todos os livros existentes na biblioteca. Para a ajudar a pesquisar pela lista de livros criei vários motores de pesquisa que permitem pesquisar pelo nome/título do livro, pela editora, pelo autor e pela categoria.

Os leitores, autores, editoras e categorias têm todas funcionalidades similares, permitido adicionar, editar e ver os dados. Tendo também implementado os motores de pesquisa consoante a necessidade.

Para requisitar um documento o programa permite seleccionar um leitor e um livro, tendo sempre em conta se o livro já está requisitado. O programa permite prolongar a data de entrega dos livros, e por fim entregar o livro.

Sempre que se pretende inserir dados novos na base de dados, seja qual for os dados que se pretende assereir o programa verifica os dados para ter a certeza que não se introduz dados errados como por exemplo inserir letras no número de páginas de um livro.

Conhecimentos aplicados

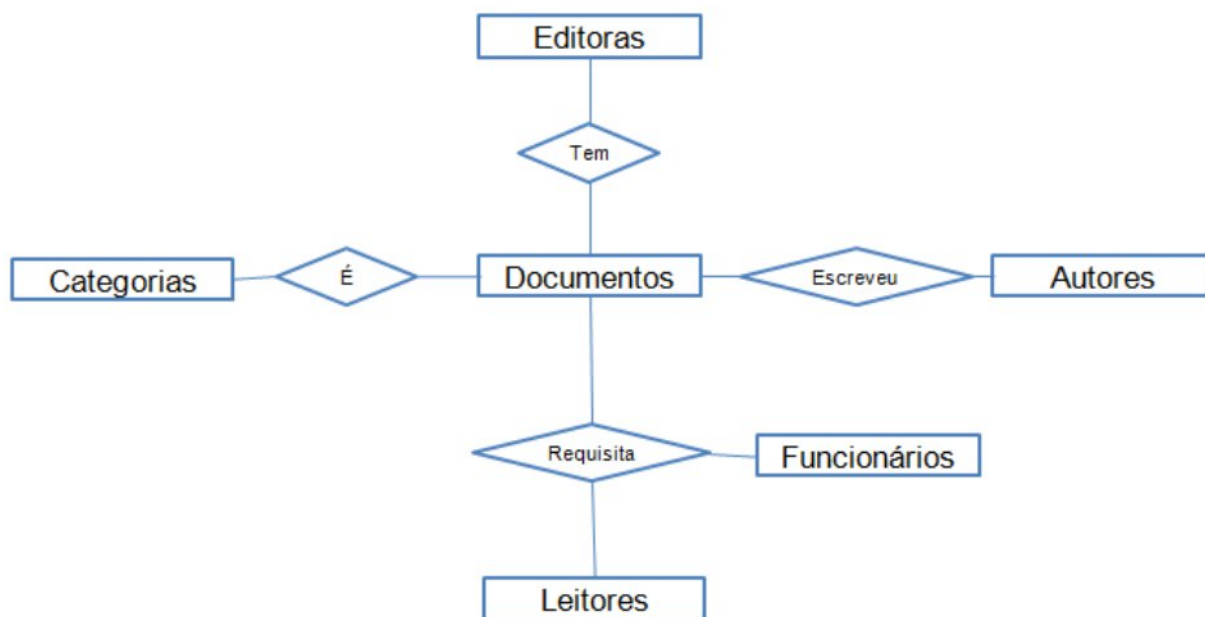
Durante o primeiro ano do curso aprende-mos os básicos da programação não gráfica, está foi uma grande base para podermos, no segundo, ano começar a trabalhar em ambientes gráficos. No segundo ano também aprendemos tudo sobre base de dados, ou seja, criar, editar, consultar, eliminar e muito mais. No terceiro e último ano do curso aprende-mos como ligar a programação gráfica às bases de dados.

Com isto em mente, no desenvolvimento do projecto tentei incorporar todas estas matérias para que o produto final ficasse mais completo e realmente demonstrasse o que aprende-mos durante estes três anos.

Mas devido à constante inovação e criação de novos métodos de programar, o que aprende-mos na escola por vezes pode não chegar para a realização do que pretende-mos, logo além do que aprendemos na escola é necessário realizar pesquisas que permitem a recolha de conhecimentos não adquiridos. As classes e os controladores são dois exemplos que conhecimentos que não foram adquiridos no curso que realmente facilitaram e enriqueceram o produto final.

Base de dados

Diagrama ER



Tabelas

Livros (Id_livr, Titulo, N_paginas, Id_cate, Id_edit , Id_auto, Data_lan)

Categorias (Id_cate, Categoria)

Editoras (Id_edir, Editora)

Autores (Id_auto, Nome, Nacionalidade, Data_nas, Data_fal)

Leitores (Id_leit, Nome, Email, Morada, Data_nas, Telemovel)

Requisita (Id_leit, Id_doco, Data_req, Id_func, Data_ent, Data_dev)

Funcionários (Id_func, Nome, Email, Morada, Data_nas , Telemovel)

CBClass

Introdução

CBClass é um projecto auxiliar que foi criada com o objectivo de guardar todas funções e controlos que são utilizados ao longo da aplicação. A razão pela qual criei este projecto foi para fosse mais fácil a realização do projecto principal, visto que apenas necessitei de escrever as funções uma vez, sem esta teria de escrever as funções vezes sem conta.

Outra vantagem de ter um projecto auxiliar é que em aplicações futuras poderei utilizar este projecto assim não terei de voltar a desenvolver funções que já tive de criar para este projecto.

Funções Gerais

As primeiras funcionalidades inseridas neste projecto foram muito simples e gerais que podem ser utilizadas em quase todos os programas.

Guardar Definições

```
public static void saveFormProperties()
{
    Variables.formProperties[0] = Form.ActiveForm.Width;
    Variables.formProperties[1] = Form.ActiveForm.Height;
    Variables.formProperties[2] = Form.ActiveForm.Location.X;
    Variables.formProperties[3] = Form.ActiveForm.Location.Y;
}
```

Esta função como objectivo guardar as propriedades de local e de tamanho da janela aberta num array global.

Carregar Definições

```
public static void loadFormProperties(Form form, bool tool = false)
{
    if (!tool)
    {
        form.Width = Variables.formProperties[0];
        form.Height = Variables.formProperties[1];
        form.Location = new System.Drawing.Point(Variables.formProperties[2], Variables.formProperties[3]);
    }
    else
    {
        form.Width = Variables.formProperties[0] - 100;
        form.Height = Variables.formProperties[1] - 100;
        form.Location = new System.Drawing.Point(Variables.formProperties[2] + 100, Variables.formProperties[3] + 100);
    }
}
```

Esta função vai buscar os dados guardados pela função de *Guardar definições* e faz com que a janela aberta ganhe essas propriedades. Com o conjunto destas duas funções pode-se fazer com que uma nova janela abra exactamente no mesmo sitio que a que estava aberta previamente.

Funções MySQL

Além de funções gerais também criei funções relacionadas com MySQL (linguagem de base de dados). Todas as ligações, à base de dados, são criadas e executadas por as seguintes funções. Ou seja sempre que é necessário aceder à base de dados, em vez de aceder directamente do programa principal, chamo estas funções.

Apesar de parecer complicar o acesso a base de dados, na realidade simplifica e muito porque as funções que criei, no meu ver, são mais fáceis, simples e directas de usar.

Criação de uma Pesquisa

```
public mysql(string columns, string tables, string condition = null)
{
    string cmdStr = "SELECT " + columns + " FROM " + tables;

    if (condition != null && condition != "")
        cmdStr += " WHERE " + condition;

    con.Open();
    cmd = new MySqlCommand(cmdStr, con);
    rdr = cmd.ExecuteReader();
}
```

Este é o código que permite criar uma nova pesquisa. Esta função recebe três parâmetros.

- Colunas: são as colunas que a pesquisa vai devolver
- Tabelas: de que tabelas pertencem as colunas
- Condição: A condição é opcional, ou seja pode-mos escolher se é necessária uma condição ou não.

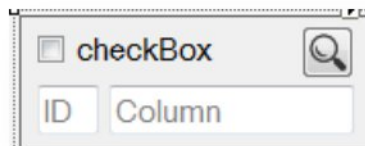
Controlo Pesquisa Estrangeira

Introdução

Os controlos personalizados são controlos que podem ser criados ao gosto do programador para realizarem variadas funções.

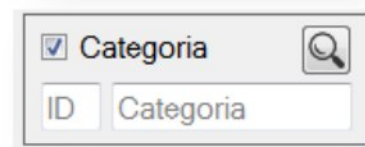
O primeiro controlador que crie tem com objectivo a pesquisa de registos em tabelas estrangeiras. Depois de seleccionar um registo numa tabela estrangeira os dados básicos desse registo serra mostrados no controlador.

Este é o aspecto do controlador original.



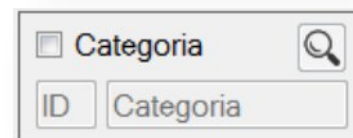
Original control interface showing a checkbox labeled 'checkBox' and a search button. Below it, there is a label 'ID' and a text box containing the word 'Column'.

Depois de alterar algumas configurações este é o aspecto do controlador.



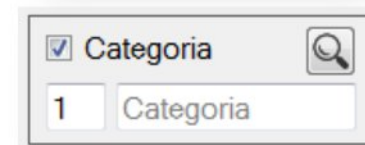
Control interface after configuration. The checkbox is now checked and labeled 'Categoria'. The text box below 'ID' now contains the word 'Categoria'.

A checkbox permite bloquear ou desbloquear a pesquisa.



Control interface with the 'Categoria' checkbox unchecked. The text box below 'ID' still contains the word 'Categoria'.

A caixa de texto ID é onde ira fica o id do registo caso já esteja algum seleccionado, caso contrário fica um texto fantasma a dizer ID, que desaparece quando se clica na caixa de texto. Se colocarmos o id de um registo e de seguida carregarmos enter ele automaticamente preenche o campo seguinte, neste caso com o nome da categoria associado a esse id.



Control interface with the 'ID' text box containing the number '1'. The 'Categoria' checkbox is checked. The text box to the right of 'ID' is empty.

O Botão com a lupa ira abrir uma nova janela que permite seleccionar um registo de todos os registos existentes, que neste caso seria todas as categorias. Depois de seleccionar o registo que pretende volta para a janela anterior e preenche o controlador com os dados seleccionados.



Control interface after selecting a record. The 'ID' text box contains '1' and the text box to the right of 'ID' now contains the word 'Aventura'.

Propriedades

Pare este controlador personalizado também é possível criar propriedades.

Para que este controlador fosse mais útil em várias situações em vez de ficar limitado apenas a um uso, criei várias propriedades que fazem o controlador dinâmico.

Lista de propriedades:

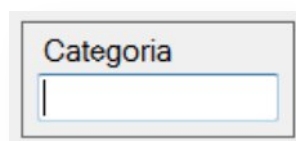
- **CheckBoxLocked:** Bloqueia a checkbox, logo não deixando a checkbox ser clicada.
- **ColumnName:** Nome da coluna onde estão os dados que serão colocados no controlador.
- **FormName:** Nome do formulário/janela em que o Botão da lupa abre para pesquisar registos.
- **IdColumnName:** Nome da coluna onde está o id.
- **IsChecked:** Se a checkbox está seleccionada ou não.
- **ReadOnly:** Define as caixas de texto para só para ler e esconde o Botão para pesquisar.
- **TableName:** Nome da tabela de queremos que o controlador pesquise
- **Text:** Texto que aparece na parte de cima do controlador
- **Value:** Valor que está na caixa de texto ID, logo se alguma registo está seleccionado esta propriedade guardará o id do registo seleccionado.

CBCheckBoxLocked	False
CBColumnName	Categoria
CBFormName	categoria
CBIdColumn	id_cate
CBIsChecked	False
CBReadOnly	False
CBTableName	categorias
CBText	Categoria
CBValue	ID

Controlo de Pesquisa Local

Introdução

Ao contrário do controlo de pesquisa estrangeira, este controlo permite pesquisar directamente na tabela pretendida, e em tempo real, ou seja ao mesmo tempo em que escrevemos a pesquisa esta a ser actualizada.



Propriedades

Apesar de este controlo ser mais simples e directo, também necessita de algumas propriedades.

Lista de propriedades:

- ColumnName: Nome da coluna em que pretendemos pesquisar.

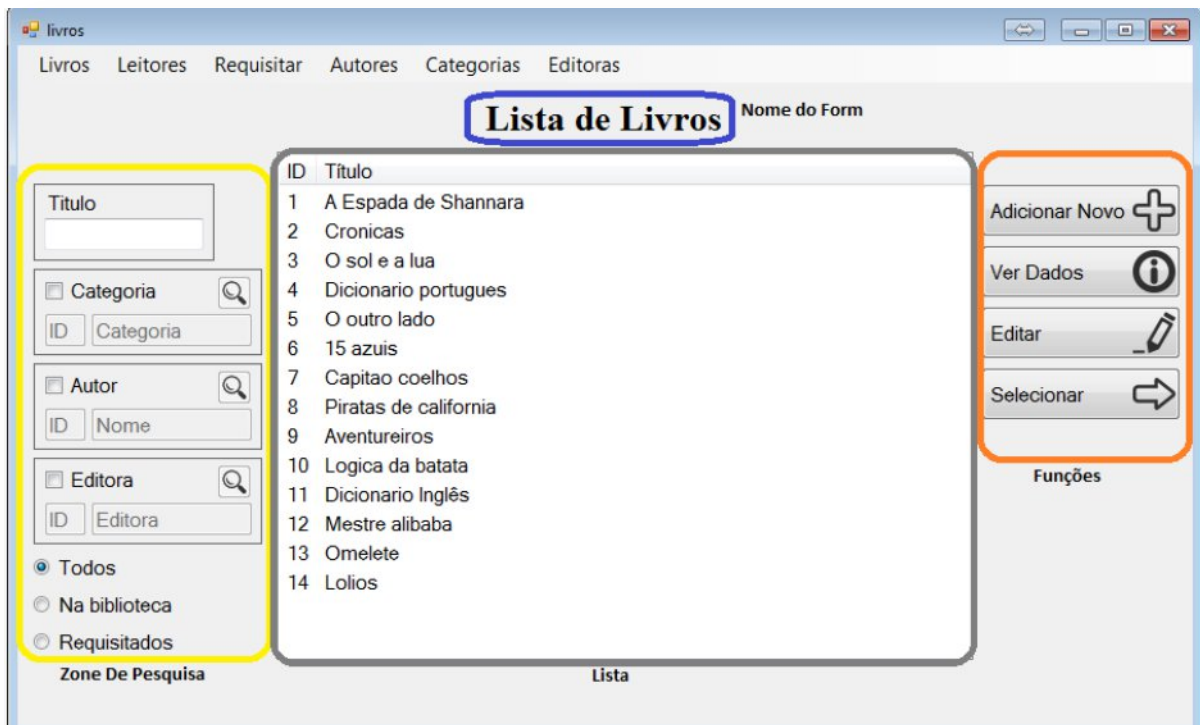
CBColumnName	categoria
CBReadOnly	False
CBText	Categoria
CBValue	

- ReadOnly: Coloca a caixa de texto so para ler.
- Text: Texto que aparece por cima da caixa de texto.
- Value: Texto que esta escrito na caixa de texto.

Formulários

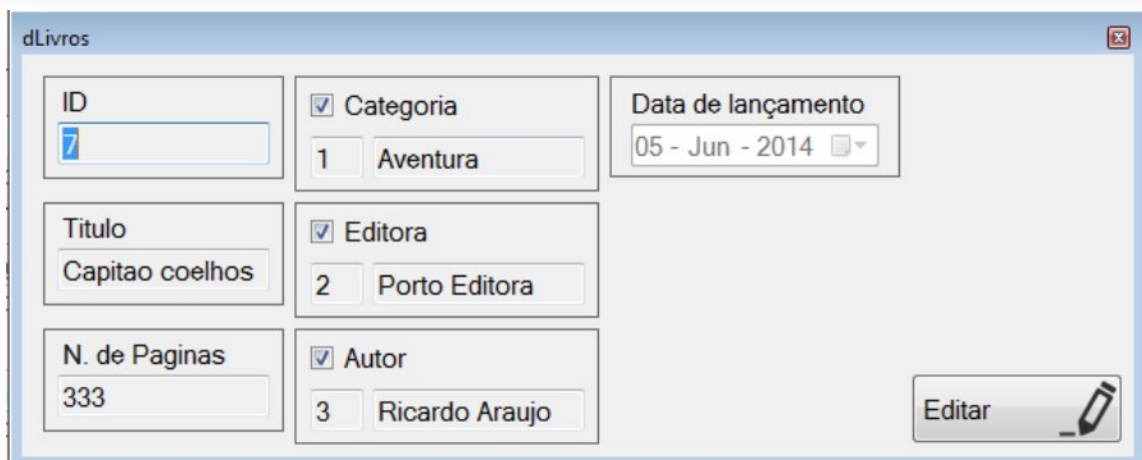
Introdução

No projecto existem dois tipos de forms. O primeiro é o que mostra uma lista de registos de uma certa tabela, e o segundo tipo de formulário é o que mostra todos os dados de um só registo.



The screenshot shows the 'livros' application window. The title bar is 'livros'. The menu bar includes 'Livros', 'Leitores', 'Requisitar', 'Autores', 'Categorias', and 'Editoras'. The main area is titled 'Lista de Livros' and 'Nome do Form'. On the left, there is a 'Zone De Pesquisa' section with search filters for 'Titulo', 'Categoria', 'Autor', and 'Editora', each with a search icon and a 'Todos' radio button. The main list displays 14 books with columns 'ID' and 'Titulo'. On the right, there is a 'Funções' section with buttons: 'Adicionar Novo' (plus icon), 'Ver Dados' (info icon), 'Editar' (pencil icon), and 'Selecionar' (arrow icon).

ID	Titulo
1	A Espada de Shannara
2	Cronicas
3	O sol e a lua
4	Dicionario portugues
5	O outro lado
6	15 azuis
7	Capitao coelhos
8	Piratas de california
9	Aventureiros
10	Logica da batata
11	Dicionario Inglês
12	Mestre alibaba
13	Omelete
14	Lolios

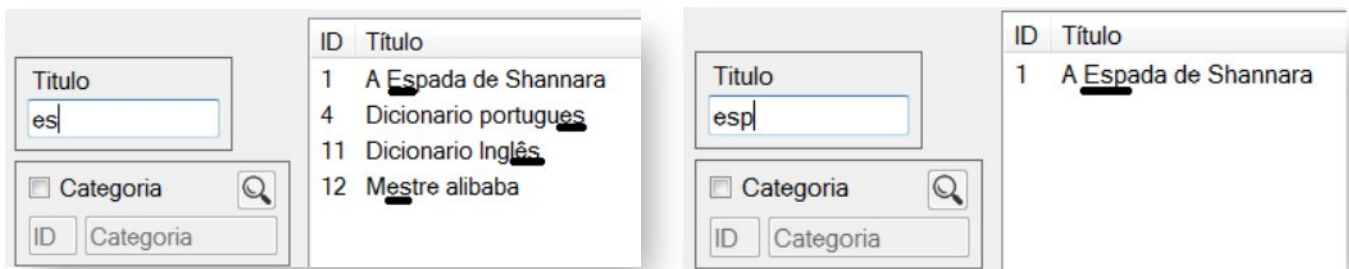


The screenshot shows the 'dLivros' application window. It displays a detailed view of a book with the following fields: 'ID' (7), 'Categoria' (1 Aventura), 'Data de lançamento' (05 - Jun - 2014), 'Titulo' (Capitao coelhos), 'Editora' (2 Porto Editora), 'N. de Paginas' (333), and 'Autor' (3 Ricardo Araujo). There is an 'Editar' button with a pencil icon.

Pesquisar

Colunas não estrangeiras

Na zona de pesquisa pode haver os controladores de pesquisa estrangeira ou os controladores de pesquisa local.



The image shows two side-by-side screenshots of a search interface. Each interface has a 'Titulo' search box, a 'Categoria' checkbox with a magnifying glass icon, and a table with 'ID' and 'Título' columns. In the left screenshot, the search box contains 'es' and the table lists four items: '1 A Espada de Shannara', '4 Dicionario portugues', '11 Dicionario Inglês', and '12 Mestre alibaba'. In the right screenshot, the search box contains 'esp' and the table lists only one item: '1 A Espada de Shannara'.

O controlador de pesquisa local actualiza a pesquisa ao mesmo tempo que escreve-mos. Assim se escrevermos "es", o controlador começa logo a pesquisar como podemos ver na imagem. Quando se adiciona mais uma letra a lista de livros actualiza mais uma vez, e mostra apenas os títulos com "esp", que neste caso é apenas um.

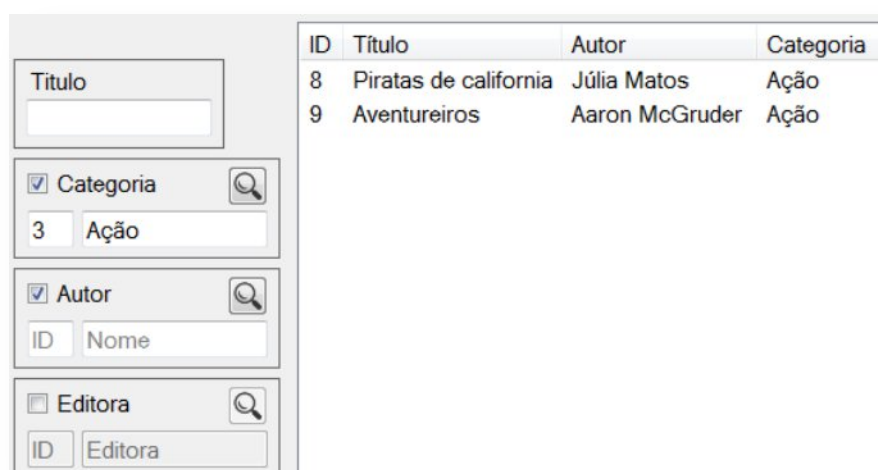
Colunas estrangeiras

Como podemos ver na imagem ao seleccionarmos o controlador, na lista é adicionada uma nova coluna correspondente ao controlador seleccionado

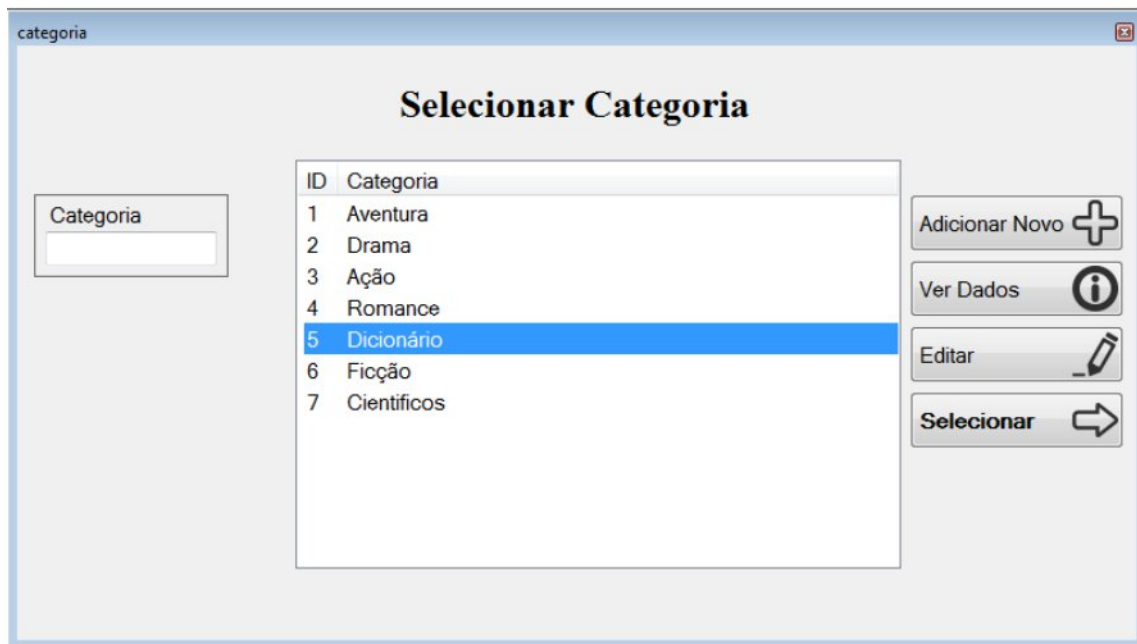


Isto faz com que não seja necessário estarem todas as colunas na lista quando o utilizador do programa pode apenas necessitar de ver uma das colunas, fazem assim o utilizador pode escolher quais as colunas que pretende ver e as que pretende esconder.

Caso o utilizador queira pesquisar livros apenas de Acção, então basta escrever o na categoria "acção" e carregar a tecla ente. O programa ira então verificar se acção é uma categoria valida, se for então ira pesquisar os livros só de acção. Isto é valido para todas as colunas estrangeiras e não só para as categorias.



Caso o utilizador pretenda pesquisar por uma categoria, mas prefira ver a lista de todas as categorias antes de escolher, então pode usar o Botão com a lupa que abre uma nova janela com todas as categorias, de onde pode escolher a categoria de pretende. Mais uma vez isto é aplicável a todas as pesquisas de colunas estrangeiras.



Funções

Os formulários com a lista de registos têm também três ou quatro funções.

Lista de funções:

- Adicionar novo: Abre o formulário de detalhes com os campos todos em branco preparados para serem preenchidos com dados para depois serem inseridos na base de dados.
- Ver dados: Abre o formulário de detalhes com os dados do registo que estava seleccionado.
- Editar: Abre o formulário de detalhes com os dados do registo, e com os campos desbloqueados para que possam ser editados.
- Seleccionar: Dependendo do formulário em esta, abre outro formulário onde o registo seleccionado serve como chave estrangeira. Por exemplo se estivermos no formulário das categorias e clicarmos o Botão seleccionar com o registo acção seleccionado então ira abrir o formulário dos livros e pesquisar apenas por livros coma categoria acção

Verificação de dados

Todos os formulários de detalhes, que são os únicos formulários onde são inseridos ou editados dados, têm verificação de dados, ou seja antes de os dados serem inseridos na base de base são submetidos a alguns testes para prevenir a inserção de dados incorrectos na base de dados.

As verificações podem ser tão simples com não permitir caixas de textos vazios, ou podem ser verificações mais complexas como verificar se um livro que se esta a tentar requisitado já esta requisitado.

Caracterização do produto final

O produto final permite a gestão completa de todos os autores, editoras, categorias, livros e leitores de uma biblioteca. Esta gestão consiste em adicionar, remover, editar e consular os registos existente.

Além de gerir o mencionado também permite registar os requisitos feitos pelos leitores da biblioteca.

O programa foi desenvolvido com objectivo de criar um programa simples mas útil e completo. No meu ver os objectivos foram cumpridos.

.

Meios de Divulgação

Visto que este projecto apenas é útil e relevante para as bibliotecas, a melhor maneira divulgar o produto não é grandes publicidades com cartazes, internet ou mesmo televisão.

Visto que o publico de que pretende alcançar é muito específico e pequeno é mais adequado apresentar o produto apenas as pessoas interessada. Com isto quero dizer, comunicar directamente com as bibliotecas utilizando o correio electrónico, chamadas telefónicas ou mesmo apresentações à biblioteca.

Conclusão

Com a realização deste projecto foi possível experienciar como é realizar um produto de dimensões mais elevadas do que as que estamos habituados. Durante os três anos realizamos muitos exercícios e trabalhos de programação, mas nunca com esta dimensão. Por vezes, quando apenas fazemos aplicações simples e com somente uma funcionalidade, assumimos que juntar tudo para fazer uma aplicação de maior dimensão é fácil. Na realidade, depois de experienciarmos, é muito diferente.

Ao longo do desenvolvimento deste projecto foi possível aplicar muitos dos conhecimentos obtidos durante os três anos, e realmente entender a funcionalidade e a importância destes.

Com a realização deste projecto foi possível entender que por vezes não é preciso ser o melhor para conseguir o que queres, mas sim temos de ter a força de vontade para alcançar o que queremos.