

EstMedia



Manual Técnico

Programação Visual 04/05

Trabalho elaborado por:
Gustavo Patrício nº 4757
António Espingardeiro nº 4328

Docente: Alpesh Ranchordas

Data de entrega: 30/06/05

Índice

Introdução	1
Descrição do Problema e Requisitos	2
Estrutura da Aplicação:	
Principal	3
FormLogin	4
FormClientes	5
FormIClientes	7
FormEmpregado	9
FormIEmpregado	11
FormFornecedores	13
FormIFornecedores	15
FormManutencao	16
FormVenda	19
FormHistorial	21
FormProdutos	22
FormAcerca	24
Ligação à B.D, Pesquisas	25
Excepções	26
Diagrama de Classes UML	27
Implementações futuras	28
Conclusão	29

Introdução

Nos tempos que decorrem somos confrontados com aplicações profissionais que possuem interfaces gráficos bastante elucidativos dos mais variados temas.

Hoje em dia a necessidade de que um programa tem que ser visualmente apelativo e de fácil uso é nota chave para o seu sucesso no mercado.

Para uma aproximação de tal objectivo foi necessário aplicar um leque de conhecimentos adquiridos ao longo do semestre que permitiram não só melhorar a performance da aplicação assim como a sua utilização por parte do utilizador.

O manual técnico tem como objectivo principal justificar todas as opções tomadas ao longo do desenvolvimento da aplicação tentando não descurar a estrutura definida na fase anterior.

Descrição do Problema

Inicialmente deparou-se com uma situação difícil por parte da empresa visto que era essencial criar uma base de dados que respondesse às necessidades da instituição e que de igual forma fosse fácil de utilizar.

Em pleno século 21 a resistência á mudança ainda é grande devido a vários factores nomeadamente o económico, mas mais importante que isso é o facto de como a empresa iria trabalhar de forma competitiva no mercado sem uma B.D informática coerente?

A EstMedia é hoje uma loja de informática que utiliza esta aplicação de forma rápida eficaz e com relativa facilidade.

Por último podemos afirmar que o objectivo desta aplicação e de muitas outras do segmento está cumprido, dar resposta às necessidades de cada instituição proporcionando-lhes as novas tecnologias num universo próprio de soluções.

Objectivos:

A segunda fase deste projecto consiste em transformar o código inicialmente programado em Modo Consola para Modo Gráfico utilizando os mecanismos de C#.

O objectivo principal do projecto é criar uma interface apelativa ao utilizador, que permita gerir de forma coerente a Base de Dados criada em Microsoft Access.

Assim sendo iremos continuar a implementar os seguintes requisitos:

Requisitos:

- Gestão de Empregados
- Gestão de Clientes
- Gestão de Serviços (Venda de Produtos, Manutenções)
- Gestão de Produtos Armazenados (Fornecedores)
- Gestão de Produtos Disponíveis

Estrutura da Aplicação

De seguida iremos retratar de forma sucinta a estrutura da aplicação, descrição de forms criadas, métodos, atributos e relações.

Principal

É aqui onde aplicação se inicia e termina. Onde podemos aceder a todas as opções postas ao nosso dispor pela aplicação.

Menu Items

Aqui encontra-se o menu principal da aplicação, para tal foi criado um mainMenuPrincipal onde foram definidos os seguintes Menus:

Arquivo: Clientes, Empregados, Fornecedores, Produtos, Serviços, Sair.

Janelas: Cascata, Tile Horizontal, Tile Vertical

Acerca de: Créditos da aplicação

Evento Principal Load

Métodos/Componentes Relevantes

StatusBar

A sua função é visualizar instruções ao Utilizador sobre qual o estado das suas operações, para tal é carregada no evento EstMedia_Load com a seguinte informação “Bem-vindo”.

Eventos do Menu Item ‘Janelas’

menuItemCascata_Click – posiciona as janelas em modo cascata

menuItemTHoriz_Click - posiciona as janelas em Tile Horizontal

menuItemTVertical_Click – posiciona as janelas verticalmente

Propriedades relevantes da janela

BackGroundImage = image.jpg

WindowState = Maximized

ISMdiContainer = true

Nesta situação foi clara uma utilização hierárquica de janelas para que o Form Estmedia seja o pai de todas as janelas que iram ser criadas futuramente.

Foi então necessário transmitir ao compilador informação (IsMdiContainer=true) que o FormPrincipal não era mais que um contentor (package) de forms.

Assim sendo para cada Form foi necessário exprimir condições de hierarquia, a instrução (Form.MdiParent=this) traduz a relação Pai/Filho.

O que acontece na realidade é que cada subForm do EstMedia contém um conjunto de operações completamente distintas, tendo sido necessário activar o mecanismo Mergeltens de forma a acrescentar as operações de cada subForm ao menu local de Opções.

Login

Este Form tem como objectivo validar os utilizadores da aplicação EstMedia, não é nada mais que um Form onde é introduzida uma palavra passe que dá acesso à área principal do programa.

FormClientes

Form onde se realiza a Gestão de Clientes.

Menu itens

Foram adicionados os seguintes menus:

Opções: Actualizar Dados, Inserir Cliente, Remover Cliente (As mesmas operações do botões).

Formatar: Tipo de Letra (permite mudar as fontes presentes nas textBoxes do utilizador, este pode personalizar o seu layout).

Evento FormCliente Load

Aqui é efectuado o carregamento dos dados existentes na base de dados transferindo informação através de uma conexão para as textBoxes oleDbDataAdapter1.Fill(dsClientes), através dos métodos mostraDados() e mostraPosicao.

Eventos associados aos Botões

<< - Primeiro Registo

> - Próximo Registo

< - Retrocesso

>> - Último Registo

Inserir Registo – Chama o FormICliente, para inserção de registos

Eliminar Registo – É mostrada ao utilizador uma MessageBox que confirma a eliminação do registo actual. Em caso afirmativo o programa executa o comando Delete.

Actualizar – faz a actualização da B.D em termos visuais (virtualmente visto que C# oferece-nos este mecanismo de forma a não sobrecarregar a aplicação em termos de processamento tornando-se mais eficaz).

Imprimir – imprime toda a informação presente no Form Cliente através do mecanismo PrintDialog.

Métodos/Componentes Relevantes

mostraDados() - neste método são definidas as ligações ás textBoxes que representam os atributos do cliente, estas iram servir de suporte á futura chamada dos botões de navegação.

mostraPosicao() – tem como objectivo mostrar uma referência para o utilizador se posicionar na navegação actual / total dos registos. O seu resultado é visível numa textBox não editável. Para tal foi necessário recorrer a 2 variáveis locais que representam numero de registos e (posicao actual + 1). O seu funcionamento pode ser descrito como: Se tabela = Vazia então “No records” senão mostra a posição respectiva em relação ao total de registos existentes.

Tooltip

A sua função é ajudar o utilizador, mostrando uma Tag de ajuda, no esclarecimento acerca de uma determinada opção disponível.

Tags adicionadas aos botões:

```
toolTip1.SetToolTip(buttonImprimir, "Imprimir Formulário");  
toolTip1.SetToolTip(buttonAnterior, "Registo Anterior");  
toolTip1.SetToolTip(buttonProximo, "Próximo Registo");  
toolTip1.SetToolTip(buttonPrimeiro, "Primeiro Registo");  
toolTip1.SetToolTip(buttonUltimo, "Último Registo");
```


FontDialog

Tem como objectivo formatar o texto visível ao utilizador, ou seja se este quiser modificar as fontes actuais das textBoxes, pode fazê-lo escolhendo a formatação desejada.

PrintDialog

Objecto que nos permite ter acesso á impressão de um documento através da instanciação da classe PrintDocument.

GroupBox Pesquisar (Destina-se unicamente ás pesquisas do utilizador)

CheckBoxCliente – Procura pelo nome de um cliente na B.D e mostrando esse registo ao utilizador, recorrendo ao uso da TextBox Pesquisar e do botão pesquisar.

LinkLabelCompras – Cliente com maior número de compras

Propriedades Relevantes da Janela

FormBorderStyle = FixedSingle

MaximizedBox = False

FormCliente

Este Form destina-se a inserção de dados de um novo Cliente (Novo Registo).

Eventos associados aos Botões

Confirmar – actualiza a B.D fisicamente, adiciona o registo e providencia uma MessageBox confirmando a operação.

Em termos de execução o que programa faz é adicionar uma linha á B.D existente através da execução do comando OleDbInsertCommand.

Cancelar – cancela e regressa ao Form anterior.

Métodos/Componentes mais Relevantes

errorProvider : Surge aqui como um novo mecanismo de excepção disponível para programadores, visto que disponibiliza um alerta para o utilizador quando este não preenche na totalidade os campos necessários á subscrição (típico dos formulários Online).

Basicamente o que este controlo faz é testar se as textBoxes introduzidas pelo utilizador estão totalmente preenchidas, caso contrário não insere dados na B.D e providencia uma alerta visual localizando rapidamente no formulário os campos não preenchidos.

helpProvider : HelpProvider é um processo de ajuda constante ao utilizador que providencia informações relevantes sobre os contextos a preencher nas textBoxes. Basicamente podemos utilizar este mecanismo em qualquer Form activando a propriedade HelpProvider = true.

Propriedades Relevantes da janela

ShowDialog – Não é possível regressar á janela anterior (típico dos menus inserção), isto através da propriedade activa no Form anterior através do botão Inserir Cliente. Essencialmente comporta-se como uma propriedade existente numa Form normal.

FormBorderStyle = FixedDialog

HelpButton = true

MaximizeBox = False

MinimizedBox = False

ControlBox = False
ShowinTaskbar = False

FormEmpregados

Form onde se realiza a Gestão de Empregados.

Menu Itens

Foram adicionados os seguintes menus:

Opções: Actualizar Dados, Inserir Empregado, Remover Empregado (As mesmas operações do botões).

Formatar: Tipo de Letra (permite mudar as fontes presentes nas textBoxes do utilizador, este pode personalizar o seu layout).

Evento FormEmpregados Load

Aqui é efectuado o carregamento dos dados existentes na base de dados transferindo informação através de uma conexão para as textBoxes oleDbDataAdapter1.Fill(dsEmpregados), através dos métodos mostraDados() e mostraPosicao().

Eventos associados aos Botões

<< - Primeiro Registo

> - Próximo Registo

< - Retrocesso

>> - Último Registo

Inserir Registo – Chama o FormIEmpregado.

Eliminar Registo – É mostrada ao utilizador uma MessageBox que confirma a eliminação do registo actual. Em caso afirmativo o programa executa o comando Delete.

Imprimir – Faz a impressão do Form Empregados.

Métodos/Componentes Relevantes

mostraDados() - neste método são definidas as ligações às textBoxes que representam os atributos do empregado, estas iram servir de suporte á futura chamada dos botões de navegação.

mostraPosicao() – tem como objectivo mostrar uma referência para o utilizador se posicionar na navegação actual / total dos registos. O seu resultado é visível numa textBox não editável. Para tal foi necessário recorrer a 2 variáveis locais que representam numero de registos e (posicao actual + 1). O seu funcionamento pode ser descrito como: Se tabela = Vazia então “No records” senão mostra a posição respectiva em relação ao total de registos existentes.

GroupBox Pesquisar (Destina-se unicamente ás pesquisas do utilizador)

ComboBox – Adiciona as seguintes opções:

```
comboBoxCategoria.Items.Add("Técnico");  
comboBoxCategoria.Items. ("Atendimento");
```

TextBoxPesquisa – Pesquisa por Nome

Link – Empregado com maior número de vendas

Tooltip

A sua função é ajudar o utilizador, mostrando uma Tag de ajuda, no esclarecimento acerca de uma determinada opção disponível.

Tags adicionadas aos botões:

```
toolTip1.SetToolTip(buttonImprimir, "Imprimir Formulário");  
toolTip1.SetToolTip(buttonAnterior, "Registo Anterior");  
toolTip1.SetToolTip(buttonProximo, "Próximo Registo");  
toolTip1.SetToolTip(buttonPrimeiro, "Primeiro Registo");  
toolTip1.SetToolTip(buttonUltimo, "Último Registo");
```

FontDialog

Tem como objectivo formatar o texto visível ao utilizador, ou seja se este quiser modificar as fontes actuais das textBoxes, pode fazê-lo escolhendo a formatação desejada.

PrintDialog

Objecto que nos permite ter acesso á impressão de um documento através da instanciação da classe PrintDocument.

Propriedades Relevantes da Janela

FormBorderStyle = Sizable
MaximizedBox = False

FormEmpregados

Este Form destina-se a inserção de dados de um novo Empregado (Novo Registo).

Eventos associados aos Botões

Confirmar – Actualiza a B.D fisicamente, e providencia uma MessageBox confirmando a operação.

Cancelar – cancelar e regressa ao Form anterior.

Métodos/Componentes Relevantes

ListBoxes

Surgem aqui com o objectivo de poder facilitar a selecção do tipo de empregado e o seu salário, não são nada mais que campos pertencentes à B.D e que futuramente iram ser actualizados com os valores do utilizador.

Itens adicionados às ListBoxes:

```
listBoxCategoria.Items.Add("Atendimento");  
listBoxCategoria.Items.Add("Técnico");  
listBoxSalario.Items.Add("1000");  
listBoxSalario.Items.Add("1500");  
listBoxSalario.Items.Add("2000");  
listBoxSalario.Items.Add("3000");
```

errorProvider : Basicamente o que este controlo faz é testar se as textBoxes introduzidas pelo utilizador estão totalmente preenchidas, caso contrário não insere dados na B.D e providencia uma alerta (pisca) localizando rapidamente no formulário os campos incorrectos.

helpProvider : HelpProvider é um processo de ajuda constante ao utilizador que providencia informações relevantes sobre os contextos a preencher nas textBoxes. Basicamente podemos utilizar este mecanismo em qualquer Form activando a propriedade HelpProvider = true.

Propriedades Relevantes da janela

ShowDialog – Não é possível regressar à janela anterior (típico dos menus inserção), isto através da propriedade activa no Form anterior através do botão Inserir Empregado.

FormBorderStyle = FixedSingle

Control Box = False
MaximizedBox = False
MinimizedBox = False

FormFornecedores

Form onde se realiza a Gestão de Fornecedores.

Menu Itens

Foram adicionados os seguintes menus:

Opções: Actualizar Dados, Inserir Fornecedor, Remover Fornecedor (As mesmas operações do botões).

Formatar: Tipo de Letra (permite mudar as fontes presentes nas textBoxes do utilizador, este pode personalizar o seu layout).

Evento FormFornecedores Load

Aqui é efectuado o carregamento dos dados existentes na base de dados transferindo informação através de uma conexão para as textBoxes `oleDbDataAdapter1.Fill(dsFornecedores)`, através dos métodos `mostraDados()` e `mostraPosicao`.

Eventos associados aos Botões

<< - Primeiro Registo
> - Próximo Registo
< - retrocesso
>> - Último Registo

Inserir Registo – Chama o FormIFornecedores, para inserção de registos

Eliminar Registo – É mostrada ao utilizador uma MessageBox que confirma a eliminação do registo actual. Em caso afirmativo o programa executa o comando Delete.

Actualizar – faz a actualização da B.D em termos visuais (virtualmente visto que C# oferece-nos este mecanismo de forma a não sobrecarregar a aplicação em termos de processamento tornando-se mais eficaz).

O componente StatusBar existente no Form pai é actualizado com a mensagem de “Dados actualizados”.

Imprimir – Faz a impressão do Form Fornecedores, através do mecanismo PrintDialog.

Métodos/Componentes Relevantes

mostraDados() - neste método são definidas as ligações às textBoxes que representam os atributos do cliente, estas iram servir de suporte á futura chamada dos botões de navegação.

mostraPosicao() – tem como objectivo mostrar uma referência para o utilizador se posicionar na navegação actual / total dos registos. O seu resultado é visível numa textBox não editável. Para tal foi necessário recorrer a 2 variáveis locais que representam numero de registos e (posicao actual + 1). O seu funcionamento pode ser descrito como: Se tabela = Vazia então “No records” senão mostra a posição respectiva em relação ao total de registos existentes.

GroupBox Pesquisa

CheckBox – Procura Marca do Fornecedor

Tooltip

A sua função é ajudar o utilizador, mostrando uma Tag de ajuda, no esclarecimento acerca de uma determinada opção disponível.

Tags adicionadas aos botões:

```
toolTip1.SetToolTip(buttonImprimir, "Imprimir Formulário");  
toolTip1.SetToolTip(buttonAnterior, "Registo Anterior");  
toolTip1.SetToolTip(buttonProximo, "Próximo Registo");  
toolTip1.SetToolTip(buttonPrimeiro, "Primeiro Registo");  
toolTip1.SetToolTip(buttonUltimo, "Último Registo");
```

FontDialog

Tem como objectivo formatar o texto visível ao utilizador, ou seja se este quiser modificar as fontes actuais das textBoxes, pode fazê-lo escolhendo a formatação desejada.

PrintDialog

Objecto que nos permite ter acesso á impressão de um documento através da instanciação da classe PrintDocument.

Propriedades Relevantes da Janela

FormBorderStyle = Sizable

MaximizedBox = False

FormIFornecedores

Este Form destina-se a inserção de dados de um novo Fornecedor (Novo Registo).

Eventos associados aos Botões

Confirmar – Faz recurso ao mecanismo **errorProvider**. Actualiza a B.D fisicamente, e providencia uma MessageBox confirmando a operação.

Cancelar – Cancela o Registo e regressa ao Form anterior.

Métodos/Componentes Relevantes

errorProvider : Basicamente o que este controlo faz é testar se as textBoxes introduzidas pelo utilizador estão totalmente preenchidas, caso contrário não insere dados na B.D e providencia uma alerta (pisca) localizando rapidamente no formulário os campos incorrectos.

helpProvider : HelpProvider é um processo de ajuda constante ao utilizador que providencia informações relevantes sobre os contextos a preencher nas textBoxes. Basicamente podemos utilizar este mecanismo em qualquer Form activando a propriedade HelpProvider = true.

Propriedades Relevantes da janela

ShowDialog – Não é possível regressar à janela anterior (típico dos menus inserção), isto através da propriedade activa no botão do Form anterior Inserir Fornecedor.

ControlBox = False

MaximizedBox = False

MinimizedBox = False

FormManutenção

Este Form destina-se a inserção de novos Registos de Manutenções.

Eventos associados aos Botões

Confirmar – Actualiza a B.D fisicamente, e providencia uma MessageBox confirmando a operação.

Cancelar – cancelar e regressa ao Form anterior.

Botões **Procura** – Existem com o objectivo de o utilizador poder consultar informação sobre determinado Cliente, Empregado ou Produto, antes de preencher o formulário final. Para tal foi necessário criar um objecto de cada classe para garantir o acesso á visualização das mesmas.

Métodos/Componentes Relevantes

ComboBox

Surge aqui com o objectivo de poder facilitar a selecção do tipo de Pagamento Efectuado.

Items adicionados á ComboBox:

```
comboBoxPagamento.Items.Add("Pronto Pagamento");  
comboBoxPagamento.Items.Add("MultiBanco");  
comboBoxPagamento.Items.Add("Crédito");
```

ListBox

Surge aqui com o objectivo de poder facilitar a selecção do tipo de Categoria da Manutenção.

Items adicionados á listBox:

```
listBoxCategoria.Items.Add("1 - Gama Baixa");  
listBoxCategoria.Items.Add("2 - Gama Normal");  
listBoxCategoria.Items.Add("3 - Gama Média");  
listBoxCategoria.Items.Add("4 - Gama Alta");  
listBoxCategoria.Items.Add("5 - Gama Máxima");
```

NumericUpandDownList

A quantidade foi um exemplo claro na utilização destes cursores que incrementam ou decrementam valores conforme o utilizador desejar.

DateTimePicker

Este mecanismo é utilizado para seleccionar o dia de um calendário actual de sistema. Não é nada mais que uma comboBox que ao desdobrar mostra um calendário correspondente ao mês.

Tooltip

A sua função é ajudar o utilizador, mostrando uma Tag de ajuda, no esclarecimento acerca de uma determinada opção disponível.

Tags do tooltip:

```
tooltip1.SetToolTip(buttonProcurarCliente, "Procure um Cliente");  
tooltip1.SetToolTip(buttonProcurarEmpregado, "Procure um Empregado");
```

errorProvider : Basicamente o que este controlo faz é testar se as textBoxes introduzidas pelo utilizador estão totalmente preenchidas, caso contrário não insere dados na B.D e providencia uma alerta (pisca) localizando rapidamente no formulário os campos incorrectos.

helpProvider : HelpProvider é um processo de ajuda constante ao utilizador que providencia informações relevantes sobre os contextos a preencher nas textBoxes. Basicamente podemos utilizar este mecanismo em qualquer Form activando a propriedade HelpProvider = true.

Propriedades Relevantes da Janela

FormerBorderStyle = Sizable

MaximizedBox = False

FormVenda

Este Form destina-se a inserção de novos Registos de Vendas.

Eventos associados aos Botões

Confirmar – Faz recurso ao mecanismo **errorProvider**. Actualiza a B.D fisicamente, e providencia uma MessageBox confirmando a operação.

Cancelar – Cancela o Registo e regressa ao Form anterior.

Botões **Procura** – Existem com o objectivo de o utilizador poder consultar informação sobre determinado Cliente, Empregado ou Produto, antes de preencher o formulário final. Para tal foi necessário criar um objecto de cada classe para garantir o acesso á visualização das mesmas.

Métodos/Componentes Relevantes

ComboBox

Surge aqui com o objectivo de poder facilitar a selecção do tipo de Pagamento Efectuado.

Items adicionados ás ComboBoxes:

```
comboBoxPagamento.Items.Add("Pronto Pagamento");
```

```
comboBoxPagamento.Items.Add("MultiBanco");  
comboBoxPagamento.Items.Add("Crédito");
```

Tooltip

A sua função é ajudar o utilizador, mostrando uma Tag de ajuda, no esclarecimento acerca de uma determinada opção disponível.

Tags do Tooltip:

```
toolTip1.SetToolTip(buttonProcuraCliente, "Procure um Cliente");  
toolTip1.SetToolTip(buttonProcuraEmpregado, "Procure um Empregado");  
toolTip1.SetToolTip(buttonProcuraProduto, "Procure um Produto");
```

NumericUpandDownList

A quantidade foi um exemplo claro na utilização destes cursores que incrementam ou decrementam valores conforme o utilizador desejar.

DateTimePicker

Este mecanismo é utilizado para seleccionar o dia de um calendário actual de sistema. Não é nada mais que uma comboBox que ao desdobrar mostra um calendário correspondente ao mês.

errorProvider : Basicamente o que este controlo faz é testar se as textBoxes introduzidas pelo utilizador estão totalmente preenchidas, caso contrário não insere dados na B.D e providencia uma alerta (pisca) localizando rapidamente no formulário os campos incorrectos.

helpProvider : HelpProvider é um processo de ajuda constante ao utilizador que providencia informações relevantes sobre os contextos a preencher nas textBoxes. Basicamente podemos utilizar este mecanismo em qualquer Form activando a propriedade HelpProvider = true.

Propriedades Relevantes da Janela

FormerBorderStyle = Sizable

MaximizedBox = False

FormHistorial

Neste Form é possível a visualização do historial de serviços requisitados na empresa.

Menu Itens

Foram adicionados os seguintes menus:

Opções: Actualizar Dados (A mesma operação do botão).

Eventos associados aos Botões

Actualizar – Faz a actualização dos registos depois de uma filtragem, ou seja lista todos os tipos de produtos disponíveis em stock, basicamente comporta-se como um Reload do DataGrid.

Imprimir – Faz a impressão do Historial de serviços sendo para isso necessário recorrer ao processo PrintDialog.

Métodos/Componentes Relevantes

GroupBox (Destina-se unicamente às pesquisas do utilizador)

ComboBox – Surge aqui com o objectivo de filtrar a visualização dos dados conforme o tipo de serviço Venda ou Manutenção.

DataGrid

Tem como objectivo fazer uma listagem extensa de registos (é uma forma alternativa de mostrar os dados para além das textBoxes abordadas anteriormente), é mais eficaz em termos de localização visto que basta rodar o scroll num grande universo de posições para localizar-mos o desejado.

Contém referências dos campos presentes na tabela serviços visualizando esses mesmos campos.

PrintDialog

Objecto que nos permite ter acesso á impressão de um documento através da instanciação da classe PrintDocument.

Tooltip

A sua função é ajudar o utilizador, mostrando uma Tag de ajuda, no esclarecimento acerca de uma determinada opção disponível.

Tag adicionada ao botão:

```
toolTip1.SetToolTip(buttonImprimir, "Imprimir Formulário");
```

Propriedades Relevantes da Janela

FormBorderStyle = Sizeable

MaximizedBox = False

FormProdutos

Form onde se realiza a Gestão de Produtos.

Menu Itens

Foram adicionados os seguintes menus:

Opções: Actualizar, Salvar (As mesmas operações do botões).

Eventos associados aos Botões

Actualizar – Carrega os dados presentes na tabela Produtos.

Guardar – Actualiza a B.D fisicamente através do comando Update.

Métodos/Componentes Relevantes

GroupBox Pesquisar

RadioButtons – Têm como principal objectivo representar a selecção da pesquisa pela marca ou pelo modelo. Só é permitida uma selecção de cada vez.

ComboBox – Efectua a listagem conforme a categoria do produto (Computadores Pessoais, Consumíveis, Hardware ou Software).

Link – Produto mais requisitado.

DataGrid

Tem como objectivo fazer uma listagem extensa de registos (é uma forma alternativa de mostrar os dados para além das textBoxes abordadas anteriormente), é mais eficaz em termos de localização visto que basta rodar o scroll num grande universo de posições para localizarmos o desejado.

Contém referências dos campos presentes na tabela serviços visualizando esses mesmos campos.

PrintDialog

Objecto que nos permite ter acesso á impressão de um documento através da instanciação da classe PrintDocument.

Tooltip

A sua função é ajudar o utilizador, mostrando uma Tag de ajuda, no esclarecimento acerca de uma determinada opção disponível.

Tags adicionadas ao ToolTip:

```
toolTip1.SetToolTip(buttonImprimir, "Imprimir Formulário");
```

Propriedades Relevantes da Janela

FormerBorder = FixedSingle

MaximizedBox = False

FormAcerca

Este Form visa dar a conhecer ao utilizador os autores do projecto assim como os agradecimentos e créditos do programa.

- Ligação à Base de Dados

A ligação á base de Dados é feita adicionando um objecto OleDbDataAdapter ao form. Em seguida irá ser configurada, através de um Wizard, a Connection (conexão) ao local onde está localizada a B.D.

Em seguida é necessário seleccionar os campos da nossa tabela, e gerar um DataSet. Por último são gerados automaticamente os seguintes comandos SQL(OleDbCommand): Insert, Delete, Select Update.

Este processo é executado para as várias forms existentes, de forma a que cada tabela tenha o seu DataSet correspondente.

Em qualquer dos casos é necessário copiar a base de dados para a pasta Debug e proceder a uma alteração no caminho relativo da aplicação com o objectivo que esta possa ser executada correctamente em qualquer directório de qualquer máquina.

- Pesquisas

As pesquisas efectuadas na aplicação recorrem á utilização de um DataView que é criado cada vez que se necessita de uma nova pesquisa. Este tem que ser referenciado para a tabela específica onde se pretende efectuar a pesquisa.

Existem dois tipos de pesquisa: uma pelo parâmetro introduzido pelo utilizador e outra com parâmetros predefinidos na aplicação.

No caso de se necessitar de fazer uma pesquisa específica através de um determinado parâmetro introduzido pelo utilizador na B.D, o resultado da pesquisa é a filtragem de um único registo, caso esse exista.

Na segunda situação a pesquisa pode resultar em 0,1 ou mais registos, consoante os existentes na tabela.

Os resultados das pesquisas são visualizados de duas formas: em textBox ou DataGridView dependendo da situação em que o utilizador se encontra.

- Exceções

Nesta aplicação foram utilizados 2 mecanismos de tratamento de exceções distintos.

O primeiro e já referenciado anteriormente, `ErrorProvider`, e o mecanismo `Try & Catch`.

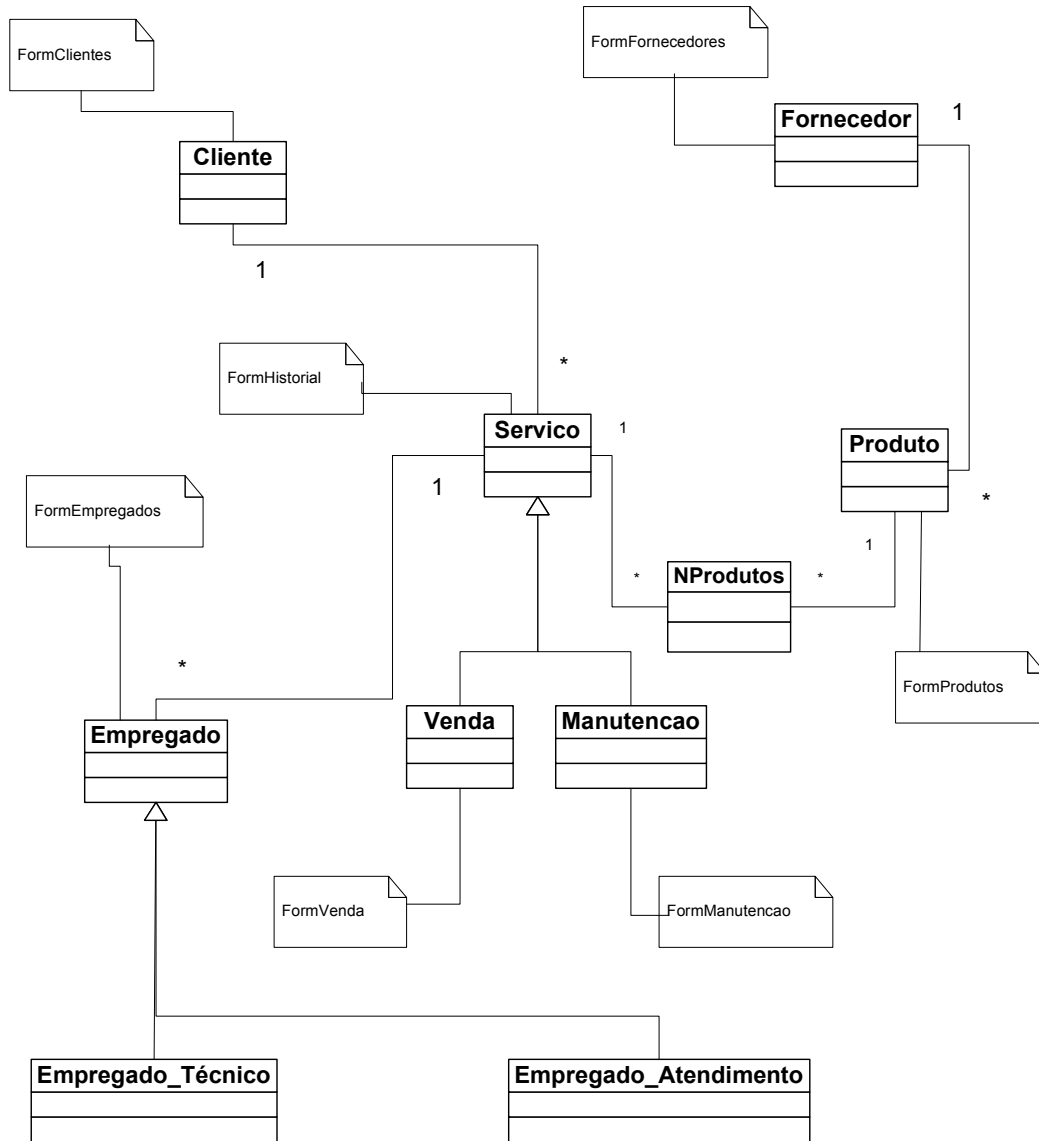
Nesta aplicação surgiram diversos tipos de exceções, generalizando, na leitura da B.D e na inserção na B.D.

A primeira situação ocorre no caso da tabela correspondente da B.D se encontrar sem registos, lançando uma exceção.

No segundo caso podem ocorrer diversos tipos de erros originados na introdução de dados pelo utilizador, como por exemplo, o não preenchimento dos campos obrigatórios e o formato inválido dos dados inseridos.

Em qualquer das situações é lançada uma exceção para tratar que será capturada e tratada de forma a que aplicação não quebre o seu ritmo normal de funcionamento.

Diagrama de Classes UML



Implementações Futuras

Visto que este projecto se insere num contexto curricular do 2º ano de Eng.Informática não nos foi possível elaborar o máximo implementações desejadas, devido a um prazo estipulado pelos docentes da disciplina.

No entanto gostaríamos de reflectir sobre os aspectos que futuramente possam ser desenvolvidos no domínio desta aplicação.

- A ligação Online é hoje uma das vertentes mais utilizadas no domínio de B.D e como tal achamos que seria importante uma fusão entre a aplicação e uma página de Internet que nos pudesse facultar os dados actualizados a qualquer momento.

- O facto de as compras Online estarem a crescer exponencialmente leva a pensar que esta loja poderia ter um serviço de apoio aos clientes e um portal para requisitar encomendas.

- No domínio das pesquisas ficaram por implementar as seguintes:

- Produto Mais Vendido
- Empregado com maior número de vendas
- Empregado pela categoria
- Cliente com mais compras

Talvez num futuro próximo poderão existir de acordo com as necessidades reais da empresa visto que estamos a partir de uma base fictícia onde não temos contacto directo com os utilizadores ou gestores de negócio.

Conclusão

Damos por concluída a segunda fase do projecto que conta ainda com a terceira fase de apresentação.

O desenvolvimento desta aplicação visual representou uma novidade, visto que havíamos sido avaliados até então através de programação de consola que é essencial á base de toda a construção de programas profissionais, mas esta precisa obrigatoriamente de uma componente visual apelativa.

Ao termos contacto com o Visual Studio.NET foi essencial perceber que a programação orientada a eventos representa um padrão de comportamentos que se espera do utilizador, tentando prever as mais diversas situações.

No seguimento deste factor surge outro ponto relevante: a criação instâncias manualmente, o que em C# modo Windows Application se resume simplesmente a arrastar controlos para a nossa área de trabalho e automaticamente essa instanciação de objectos é feita gerando o código correspondente.

Um dos aspectos importantes da cadeira é que nos aproximámos do mundo real das aplicações, tudo isto graça aos mecanismos que o C# disponibiliza e que possuem um reaproveitamento da base do sistema operativo como seja por exemplo: as Janelas, e um conjunto de propriedades que definem comportamentos habituais do nosso Windows XP e das suas aplicações reais.