

Índice

Introdução	Pag. 4
Abstract.....	Pag. 5
Estruturação do projecto	Pag. 6
Requisitos mínimos	Pag. 7
Estrutura da base de dados.....	Pag. 8
Formulários VB.Net	Pag. 10
Aplicação (Ninpo Web Site Controller)	Pag. 10
Campos dos formulários.....	Pag. 10
Navegação entre registo	Pag. 11
Gestão de registos.....	Pag. 12
Páginas Web	Pag. 14
Funções.....	Pag. 14
Navegação pelas Páginas Web	Pag. 14
Listas de registos	Pag. 15
Registo completo	Pag. 15
Navegação entre registos	Pag. 15
Álbuns de fotografias	Pag. 17
Upload de fotografias	Pag. 17
Remover fotografias	Pag. 18
Conclusão	Pag. 19
Bibliografia.....	Pag. 20
Anexos	Pag. 21
Código	Pag. 22
Imagens	Pag. 48
Fluxogramas	Pag. 53
Manual de utilizador.....	Pag. 61

Introdução

Este projecto teve como objectivo o desenvolvimento de uma aplicação, em *VB.Net*, que permitisse alterar os conteúdos da base de dados, em *MySQL*, para os mesmos serem visualizados por todos numa página *Web*, em *PHP* e *HTML*. A página tem como objectivo apresentar notícias, eventos e albuns relacionados com uma arte marcial. A aplicação serve para fazer de *backoffice* da página.

A página *Web* foi desenvolvida de modo a que os conteúdos sejam apresentados de acordo com o que se encontra na base de dados. A aplicação tem como objectivo a criação e alteração desses conteúdos, não sendo assim necessário aceder somente a um *browser* para alterar os mesmo, nem escrever código. Com o uso da aplicação a actualização dos conteúdos torna-se mais prática e fácil de fazer para o responsável do mesmo. O acesso à aplicação é limitado a determinadas pessoas, pois a base de dados encontra-se directamente ligada à mesma, e os conteúdos da página *Web* estão dependentes da base de dados.

Tanto a página *Web* como a aplicação em VB estão em uso, o programa não é fornecido pois contém os dados das contas, no entanto a página pode ser acedida através do url <http://ninjutsutrlisboa.zxq.net>. Ter em conta que os conteúdos e a disposição da página podem sofrer alterações consoante as necessidades do dono da mesma.

Abstract

This project was developed to allow the user to change the contents of a web page without having to access it. The program itself is a desktop application with limited access to registered users. This was done as a safety measure to guarantee the website's integrity.

It was developed in VB.Net with a connection to a MySQL database. This application creates and changes the contents of the database; however, it can also use the web browser to manipulate the galleries of the webpage, working as its backoffice.

The webpage was developed in PHP and HTML that can be accessed by anybody, and has a connection to the MySQL database. It is accessible to all users who wish to explore it. PHP allows the contents of the page to change but not the page itself so access and navigation of the registries is made dynamically through the database to change the contents of the page.

Estruturação do projecto

O desenvolvimento do projecto foi dividido em várias fases, de modo a facilitar a orientação e ausência de erros das mesmas. O primeiro passo no desenvolvimento foi a criação da estrutura da base de dados seguida da mesma em MySQL (a estrutura da base de dados é apresentada na secção seguinte).

Após a base de dados estar completa foram inseridos dois registos em cada tabela, excepto na tabela Login pois só era necessário um utilizador de início, para a criação das páginas *Web*. Conforme as páginas foram sendo criadas ia-se verificando se os registos eram apresentados eram idênticos aos existentes na base de dados. Após todas as páginas apresentarem os registos correctamente e estarem estruturadas conforme o desejado o próximo passo foi a criação da aplicação.

Para do desenvolvimento da aplicação o primeiro passo foi a criação da ligação a base de dados de modo a se poder consultar e comparar registos. O Primeiro formulário a ser criado foi o de login; seguindo-se o de eventos, notícias, álbuns, ajuda, *about* e comentários. A ordem de desenvolvimento de cada formulário foi idêntica para todos sendo essa a seguinte: criação dos campos; função para preenchimento dos dados e apresentar nos campos respectivos; navegação; funcionalidades (novo, guardar, apagar e fechar); verificação de alterações; possibilidade de inexistência de registos.

Após o formulário estar completo era testado de modo a ver se as funcionalidades existentes estavam a funcionar correctamente, para tal o formulário quando era aberto devia apresentar os dois registos já existentes na base de dados; de seguida um desses registos era alterado, sendo essa alteração verificada tanto na página *Web* como na base de dados. Seguidamente um novo registo era inserido e verificado, a funcionalidade de apagar era testada nesse novo registo. O fechar do formulário era testado somente após a verificação de alterações ser testada, para tal era usado um registo ao acaso e alterava-se, depois era mudado o registo sem se guardar as alterações onde era proposto para guardar as mesmas. Este processo era efectuado duas vezes, uma para escolher sim outra para escolher não, verificando assim o bom funcionamento da função.

Estando os formulários completos o processo de criar o *backoffice* de fotografias para adicionar e remover fotografias aos álbuns teve início. Esta parte do *backoffice* pode ser acedida pelo formulário Álbuns. O adicionar permite com que o álbum apresente fotografias e o remover permite remover as mesmas do servidor caso o registo seja apagado.

Estando este processo concluído teve início a escrita do relatório e anexos do mesmo para terminar o projecto.

Requisitos mínimos

Para se poder correr o NinpoWSC são necessários os seguintes requisitos:

Windows XP

- Processador Pentium 500 MHz (ou equivalente)
- 256 Mb de RAM
- Visual Basic.Net Framework (se não a tiver instalada fazer os updates ao Windows XP)
- 50 Mb de espaço livre no disco rígido
- Rato e teclado compatíveis com o Windows XP
- Ligação à internet

Windows Vista

- Processador Pentium 1 GHz 32-bit (x86) ou 64-bit (x64) (ou equivalente)
- 512 Mb de RAM
- 100 Mb de espaço livre no disco rígido
- Rato e teclado compatíveis com o Windows Vista
- Ligação à internet

Estrutura da base de dados

A base de dados usada na aplicação e na página web é a mesma, tendo o nome de **ninpodb** e sendo composta por quatro tabelas. Essas tabelas têm o nome de “eventos”, “fotos”, “login” e “noticias”.

Tabela eventos

Nome	Tipo	Tamanho
Codigo (chave primária)	Numero inteiro	128
Titulo	Varchar	128
Texto	Texto	-

Tabela fotos

Nome	Tipo	Tamanho
Codigo (chave primária)	Numero inteiro	128
Fotos 1	Varchar	256
Fotos 2	Varchar	256
Fotos 3	Varchar	256
Fotos 4	Varchar	256
Link 1	Varchar	256
Link 2	Varchar	256
Link 3	Varchar	256
Titulo_evento	Varchar	128
Comentario	Texto	-

Tabela login

Nome	Tipo	Tamanho
Username (chave primária)	Varchar	20
Password	Varchar	20

Tabela noticias

Nome	Tipo	Tamanho
Codigo (chave primária)	Numero inteiro	128
Titulo	Varchar	128
Texto	Texto	-
Data	Varchar	128

Tabela cometa rios

Nome	Tipo	Tamanho
Codigo (chave primaria)	Interio	255
Codigo_noticia	Inteiro	255
Titulo_noticia	Varchar	128
Nome	Varchar	30
Texto	Texto	-

Formulários VB.Net

Aplicação (Ninpo Web Site Controller)

No desenvolvimento da aplicação em VB.Net foram criados formulários para alterar e inserir dados nas tabelas da base de dados. Para tal foi necessário fazer com que a ligação ao *MySQL* fosse permitida. Para tal foi necessário a instalação de dois pacotes de dados (os pacotes são disponibilizados juntamente com a aplicação).

Nesta parte do relatório será explicado o processo usado para efectuar essas alterações na base de dados, começando pelo formulário de Login.

O formulário Login é carregado automaticamente quando o programa é iniciado. Caso a validação do utilizador não seja efectuada não é possível ter acesso ao resto da aplicação.

Este formulário serve para permitir o acesso ao programa caso o utilizador exista (Anexos - Imagens, img. 1). Para validar um utilizador é necessário um nome de utilizador (username) e uma palavra-chave (password).

A validação de um utilizador efectua-se através de uma função chamada *Find_user* (Anexos - Código, pag. 21), onde o utilizador e a sua palavra-chave são procurados na base de dados. Caso existam, o utilizador é validado. Caso o utilizador não seja encontrado o utilizador é informado do mesmo (Anexos - Imagens, img. 2; Anexos - Fluxogramas, flux. 1). O encerramento do Login sem a validação de um utilizador faz com que a aplicação seja encerrada.

Após a validação ter sido efectuada o formulário de Login é fechado e a janela principal, o Ninpo Web Site Controller, aparece no ecrã. O respectivo ícone na barra de tarefas está sempre presente enquanto o programa estiver a correr.

O Ninpo Web Site Controller (Anexos - Imagens, img. 3) é o que permite aceder a todos os outros formulários do programa de modo a gerir os conteúdos da página web. Os formulários existentes podem ser acedidos através do menu principal e todos eles serão considerados como *Child* do Ninpo Web Site Controller, que é considerado como o *Parent* (Anexos - Código, pag. 22). A estrutura do programa está representada em fluxograma (Anexos - Fluxogramas, flux. 2)

Sempre que se tentar fechar esta janela uma caixa de confirmação de saída irá aparecer (Anexos - Imagens, img. 4; Anexos - Código, pag. 22).

Campos dos formulários

Cada formulário tem a sua funcionalidade e a aplicação permite gerir coisas diferentes, o Eventos gere os eventos existentes ou futuros da modalidade (Anexos - Imagens, img. 5), no Noticias controlam-se as notícias relativas à modalidade (Anexos - Imagens, img. 6), o

Álbum permite gerir os álbuns de fotografias e criar novos, relacionados a eventos existentes (Anexos - Imagens, img. 7).

Nos formulários Álbum, Eventos e Noticias existem um conjunto de funções similares onde mudam apenas os nomes dos campos e a tabela da base de dados.

O formulário Álbum será o explicado uma vez que contem todas estas funções e ainda mais algumas não existentes nos outros formulários.

Existe ainda o formulário Comentários (Anexos - Imagens, img. 8) que permite apenas apagar os comentários inseridos na página Web, não permite alterar nem inserir registos como os outros formulários.

Quando o formulário é aberto, a ligação à tabela Fotos na base de dados é efectuada, sendo necessário para isso abrir uma ligação. Após ter a ligação aberta, todos os dados da tabela são copiados para um array através da função *Fill_array* (Anexos - Código, pag. 22). Usamos este array para visualizar e alterar a informação na base de dados. O código de cada álbum fica associado a uma posição através de outro array chamado *array_pos*. Este último array também é usado para a navegação no formulário. Para terminar, os campos são preenchidos com a informação que se encontra no array e as variáveis de navegação actualizadas (Anexos - Código, pag. 23; Anexos - Fluxogramas, flux. 3).

Neste formulário, em particular, existe ainda a função *Fill_title_events* que tem como objectivo ir buscar à base de dados todos os títulos dos eventos existentes, de modo a ficar associado a algum novo álbum que seja criado. O funcionamento desta função é similar ao da função *Fill_array*, com a diferença do número de colunas e do nome do array de posição que tem o nome *array_pos_title* (Anexos - Código, pag. 24).

Navegação entre registos

Os botões de navegação entre registos estão presentes, em todos os formulários, no canto inferior direito acompanhado de uma caixa que indica a posição do registo e o número de registos existentes. Caso não existam registos para serem visualizados os campos estarão em branco. Os botões de navegação são os de Primeiro, Anterior, Seguinte e Ultimo que permitem aceder aos registos. Tal como os nomes indicam, o Primeiro permite aceder ao primeiro registo existente (Anexos - Código, pag. 25); o Anterior ao registo anterior, caso se esteja no primeiro registo nada acontece (Anexos - Código, pag. 26); o Seguinte ao registo seguinte, caso se esteja no último registo nada acontece (Anexos - Código, pag. 26); e o Ultimo permite aceder ao último registo existente (Anexos - Código, pag. 27). Sempre que um destes botões é usado, o botão Adicionar fotografias é activado e a função *Check_changes* é chamada; esta função tem como objectivo verificar se o registo que está a ser visualizado sofreu alguma alteração que não tenha sido guardada antes de se visualizar outro registo, permitindo assim não se perder informação (Anexos - Código, pag. 27; Anexos - Imagens, img. 9). Caso existam alterações, aparece uma caixa de texto a perguntar se o utilizador deseja guardar as mesmas.

A função *Check_changes* utiliza a função *Update_álbum* que serve para actualizar um registo já existente. Esta função actualiza tanto a informação na base de dados como no array de informação do formulário, que é reconhecida através do seu código (Anexos - Código, pag. 28; Anexos - Fluxogramas, flux. 4).

Presentes somente no formulário Álbum, existem ainda os botões para se alterar o título, no entanto os títulos disponíveis são os existentes nos eventos.

Gestão de registos

Existem ainda no formulário os botões de controlo de um registo, sendo eles o Novo, Guardar, Apagar, Fechar e, exclusivamente no Álbum, o botão Adicionar fotografias. O Novo é usado para criar um novo registo (caso não existam já sessenta registos). Caso esta condição se verifique, a função *New_álbum* é chamada e, no caso do álbum, o botão Adicionar fotografias é desactivado (Anexos - Código, pag. 28). A função *New_álbum* actualiza a variável *Save_type* para *true*, cria um código novo, sendo o número seguinte ao último existente, coloca o último título de evento existente no respectivo campo e limpa os campos restantes (Anexos - Código, pag. 29).

O botão Guardar tem duas funcionalidades, serve tanto para fazer a actualização de um registo já existente, como para guardar um novo na base de dados. A escolha da função a ser utilizada é feita através da variável *Save_type* (referida no botão Novo). Caso seja *False*, a função usada é o *Update_álbum*; caso seja *True*, a função passa a ser o *Save_álbum* (Anexos - Código, pag. 29). No entanto, só é possível guardar ou actualizar caso os campos obrigatórios estejam preenchidos (para se saber quais é que são em cada formulário consultar o formulário de ajuda no separador respectivo ao formulário). A função *Save_álbum* usa a ligação para inserir os dados presentes nos campos na tabela da base de dados. Após ser inserido, o novo álbum vai buscar toda a tabela de novo à base de dados para actualizar o array de informação, usando novamente a função *Fill_array*: actualizando de seguida a variável *code* e a *Save_type* para *False*. No fim desta função é apresentada uma caixa de mensagem para lembrar o utilizador de inserir as fotografias do álbum; sendo esta caixa de mensagem exclusiva do Álbum (Anexos - Código, pag. 30).

De seguida existe o botão Apagar que permite ao utilizador apagar um registo. Neste botão, o botão de Adicionar fotografias é activado e a função *Delete_álbum* é chamada (Anexos - Código, pag. 31). Esta função tem como objectivo apagar um registo da base de dados. Para isso ser possível usa-se o código do registo para o fazer, de seguida a informação do array é actualizada usando, mais uma vez, a função *Fill_array* e actualizando os campos existente após esta actualização (Anexos - Código, pag. 31). No caso particular do form Noticias, quando um registo é apagar os comentários correspondentes a esse registo também serão apagados.

Para terminar os botões, temos o botão de Fechar que faz com que o formulário encerre (Anexos - Código, pag. 32). Para que tal aconteça, a função *Close_álbum* é chamada, mas

não antes da função *Check_changes* para verificar alterações. A função *Close_álbum* simplesmente encerra a ligação e o formulário (Anexos - Código, pag. 32).

Exclusivamente no Álbum existe ainda mais dois botões chamados Adicionar fotografias e Remover fotografias. O primeiro botão abre uma janela do browser onde é possível fazer o *upload* de fotografias do álbum, de modo a que se tenha uma ideia das fotos existentes no álbum quando se está a visualizar a página web (Anexos - Código, pag. 33); o funcionamento desta página será explicado mais à frente no relatório. O segundo botão tem também abre a janela do browser mas neste caso é para confirmar a remoção das fotografias do álbum. O funcionamento destes dois botões é idêntico, mudando apenas o destino dos mesmos.

Páginas Web

As páginas web têm como objectivo permitir a qualquer utilizador visualizar os conteúdos de texto, imagens e pacotes de fotografias existentes. Notar que esta página é uma página informativa e não tem áreas restritas aos utilizadores. A estrutura da página está representada em fluxograma (Anexos - Fluxogramas, flux. 5). É possível aceder a página através do url <http://ninjutsutrlisboa.zxq.net/index.PHP>.

Funções

Na criação da estrutura das páginas web foram criados ficheiros para conterem funções (quer funções usadas para todos, quer funções específicas para ficheiros específicos). O ficheiro *functions.PHP* contém as funções que são usadas em geral e acedidas por quem necessitar. Neste ficheiro encontra-se a função para se ligar à base de dados (Anexos - Código, pag. 34).

Existem ainda ficheiros de funções mais específicas para as tabelas da base de dados, eventos, fotos e notícias; sendo os nomes para cada respectivamente *functions_eventos.PHP*, *functions_fotografias.PHP* e *functions_noticias.PHP*. Estes três ficheiros de funções contêm uma função idêntica onde apenas a tabela acedida é que se altera, tendo como objectivo ir buscar os registos existentes na tabela e apresentar por ordem ascendente do código do registo (Anexos - Código, pag. 34 - Notar que só uma das funções foi apresentada). No entanto, os ficheiros *functions_noticias.PHP* e *functions_fotografias.PHP* contêm mais duas funções, cada um, do que o outro ficheiro.

No ficheiro *functions_noticias.PHP* as funções são usadas para listar e inserir comentários. A função “lista_comentarios” selecciona os comentários correspondentes a notícias a ser visualizada (Anexos - Código, pag. 35); por seu lado a função “inserir_comentario” actualiza a base de dados quando um comentário é inserido (Anexos - Código, pag. 35).

Estas funções no *functions_fotografias.PHP* são usadas também para actualizar a base de dados após fazer o upload das fotografias, “inserir_fotos” (Anexos - Código, pag. 36), enquanto que a outra função serve para procurar a existência do código na tabela da base de dados (Anexos - Código, pag. 36); o funcionamento destas funcionalidades estão explicadas mais a frente.

Navegação pelas Páginas Web

A navegação entre os diferentes tipos de informação é feita através dos links que se encontram no lado esquerdo das páginas. Estes links estão sempre presentes em qualquer

zona da página Web, com a excepção do *Home*; no entanto se o utilizador já se encontrar na página o link não existirá (Anexos - Código, pag. 37; Anexos - Fluxogramas, flux. 6). A divisão da coluna de navegação e a parte central da página é feita através de dois CSS que contêm um código similar.

Listas de registos

As listas de registos apresentam todos os registos existentes numa tabela por ordem crescente, em relação ao código de cada registo. A tabela vai ter menos uma coluna como tabela correspondente da base de dados, pois o código do registo não é apresentado. Este processo é feito através de um ciclo *While* que percorre todos os registos (Anexos - Código, pag. 38). A coluna final da tabela é um *link* que permite ter acesso ao registo completo, uma vez que o texto do registo nas tabelas está limitado. Para o registo ser o correcto quando se passa da lista para o registo completo, quando o *link* é usado, a variável *\$i* leva o código do registo em causa.

Registo completo

Quando um registo completo está a ser visualizado tem-se acesso aos textos completos. Dependendo do tipo de registo a ver visualizado (evento ou notícia) o número de campos existente difere.

Quando o registo completo é aberto é procurado logo de início um valor para a variável *\$i* de modo a que o registo correcto seja visualizado. Esta variável contém o valor do código do registo. O *\$i* também não pode ter um valor inferior a zero nem maior que o número de registo que a tabela, este problema é tratado no início da página logo após o valor de *\$i* ter sido lido.

Os conteúdos do registo são apresentados em caixas de texto, correspondendo cada caixa de texto a uma coluna da base de dados (Anexos - Código, pag. 38).

Notar que a página das notícias contém comentários, que não acontece nos eventos, que podem ser inseridos e visualizados por qualquer utilizador. Os comentários são apresentados após a notícia completa, estado a caixa para inserir o comentário seguido dos comentários existentes, se for o caso. Conforme a notícia é alterada os comentários também alteram para os da notícia correspondente.

Navegação entre registos

Quando se está a visualizar um registo completo é possível navegar entre eles sem ter de voltar a lista para visualizar todos. Este processo é efectuado através de *links* que se encontram no fim das caixas de texto de cada registo. A navegação é efectuada num processo similar ao da aplicação em VB.Net através do Início, Anterior, Próximo e Final.

Os *links* de navegação só estão activos caso existam registos para tal. Uma vez que a visualização de um registo é feito através da variável $\$i$, se estiver a ser visualizado, o primeiro registo os *links* para o Início e Anterior não estarem activos. O mesmo acontece com os *links* para o Próximo e o Final, caso o $\$i$ seja igual ao número de registos existentes estes *links* estarão inactivos (Anexos - Código, pag. 39).

Caso nenhum dos casos anteriores seja aplicável, o botão Início atribui o valor zero a $\$i$, o *link* Anterior reduz um valor à variável, o Próximo incrementa um valor a $\$i$ e o Final atribui o valor do número de registos à variável $\$i$.

Álbuns de fotografias

Nesta página Web é possível ter acesso aos conteúdos dos álbuns de fotografias. É possível visualizar fotografias que demonstrem o conteúdo do álbum completo. Estas fotografias estão dispostas numa tabela que contém no máximo quatro células, sendo o número de células dependente do número de fotografias.

Os *links* existentes em baixo destas fotografias permitem ter acesso ao álbum completo, podendo estar dividido até três partes; dependendo do número de partes, os *links* são colocados ou não na página Web. Por fim, esta página dispõe ainda de um comentário relativo ao álbum em questão; este encontra-se numa caixa de texto situada por baixo dos *links*. (Anexos - Código, pag. 40).

Upload de fotografias

Esta página Web faz parte do *backoffice* da aplicação em VB.Net, consequentemente utilizadores não registados na aplicação não conseguem ter acesso à mesma. O acesso a esta página tem de ser feito através da aplicação em VB.Net uma vez que o código do álbum é transmitido pela aplicação; caso o código não corresponda a um código de um álbum existente o botão que permite enviar os dados ficará inactivo.

A função desta página é permitir ao utilizador associar imagens a um álbum de fotografias, de modo a que se seja possível ter uma ideia do tipo de imagens dentro do álbum, sem ser necessário descarregá-lo. O número máximo de imagens que se pode associar é quatro, tendo de ser um dos formatos Jpeg, Jpg, Gif e Png.

O funcionamento desta página é simples uma vez que só se têm de escolher as imagens desejadas através das caixas *File upload* que aparecem, uma vez carregado o botão “*browse...*”. Cada caixa de texto tem um botão à frente, o que fará com que o *path* do ficheiro escolhido apareça na caixa respectiva. O número da fotografia está indicado antes da respectiva caixa (Anexos - Código, pag. 43). O processo para verificar o formato da imagem é idêntico para as quatro imagens, sendo este processo representado apenas uma vez (Anexos - Fluxogramas, flux. 7).

Quando esta página Web é apresentada, um ciclo para copiar as imagens para a pasta de Imagens é corrida, mas como o *path* das mesmas não existe ainda, o ciclo não tem efeito (Anexos - Código, pag. 45). Este ciclo tem efeito quando se carrega no botão “Enviar” e as caixas de texto com os *paths* encontram-se preenchidas; no entanto, os *paths* têm de ser válidos e o formato das imagens tem de estar entre os suportados, caso contrario o efeito será nulo para a imagem.

Após as imagens desejadas estarem seleccionadas o utilizador deve carregar no botão “Enviar” para mandar as imagens para a pasta correcta e actualizar a base de dados através

da função “inserir_fotos”; esta função tem como objectivo actualizar os campos correspondentes às fotos, caso existam (o código desta função encontra-se nos anexos, ver Funções mais acima).

Remover fotografias

Tal como a página anterior, faz parte do *backoffice* da aplicação. Esta página permite remover as imagens que se encontram no servidor associadas ao álbum. Para tal, o utilizador tem apenas de carregar no botão *OK* e as mesmas serão removidas. O código do álbum é apresentado na página, caso o utilizador deseje confirmar com o que é apresentado na aplicação *NinpoWSC*.

Tal como a página anterior, se o código não corresponder a nenhum existente na base de dados, a remoção não é permitida, pois o botão estará inactivo. O processo de remoção de uma fotografia é exemplificado num fluxograma (Anexos - Fluxogramas, flux. 8) sendo o processo idêntico para as todas as fotografias.

Conclusão

Através do uso desta aplicação, a gestão da página *Web* é fácil e acessível a pessoas sem grande experiência em fazer essa mesma gestão. A facilidade de acesso que o programa permite aos conteúdos da página *Web*, facilita o processo de gestão da mesma tornando-se numa boa ferramenta de trabalho para quem tem de estar de a actualizar/inserir conteúdos na página constantemente.

Bibliografia

- @ http://macoratti.net/net_mysql.htm
- @ <http://dev.mysql.com/downloads/connector/net/5.1.html>
- @ http://www.codeproject.com/useritems/VBnet_to_mySQL_Server.asp
- @ <http://forums.devshed.com/net-development-87/getting-vb-net-to-open-a-web-page-on-a-79500.html>
- @ <http://pt2.PHP.net/manual/en/function.unlink.PHP>
- @ http://pt.PHP.net/file_Exists
- @ <http://en.allexperts.com/q/VB-NET-3306/create-setup-disk-VB.htm?zIr=5>
- @ http://visualbasic.about.com/od/usingvbnet/a/SetupProj02_3.htm
- @ <http://www.PHP.net/>
-  Alexandre Pereira e Carlos Poupa (2004) **Linguagens WEB** Edições Sílabo
-  Pedro Coelho (Março 2001) **HTML 4 & XHTML** 3ª edição FCA