City Business Logic Presents

WEB DATABASE APPLICATION CODE GENERATOR

Version 10

© City Business Logic 2001 - 2009

USER MANUAL

Index

	Page
Introduction	3
Features and Benefits	4
No-Nonsense License	5
FAQ	6
Support / Development	7
System Requirements	8
Guided Tour	9
Generated Application Features	24
Creating your own Skins	27
Creating your own Layouts	28
Translating your applications into another language	30
Modifying scripts	31
Uploading your application to a web server	32

http://www.citybusinesslogic.com

Introduction

PHP/MySQL Web Database Application Code Generator is a Windows application which automatically creates Ajax enabled, Data-Entry / Enhanced CRUD (Create, Read, Update and Delete) Web applications written in PHP, straight out of your MySQL Database in minutes without writing a line of code.

The applications it generates allow users to login, search, filter sort, edit, update and delete data in the database, using familiar and intuitive forms and grids.

How does it work?

The generator works by first connecting to your database to analyse it's structure. It attempts to sniff out foreign keys (lookup fields) and determines whether for example a drop down list, search form or many-to-many list is most relevant. Additionally it tries to extract user-friendly field labels from the field name.

It then presents you the database in a treeview, allowing you to tweak the default choices.

Once you are ready, you simply click the generate button and it uses the supplied, fully-customisable code templates, layouts, language definition and skins to produce a slick, ready-to-use Web Application in minutes, without writing a single line of code.

The PHP code it generates is designed to be easy to understand and maintain, making it relatively easy to adapt and extend the code to add new functionality, as well as being an excellent learning resource.

What kind of applications can you create with it?

It is best suited to creating back-office data entry applications, though with a bit of tinkering some users have used it as part of their front end applications.

Who is it aimed at?

It is equally useful for experienced programmers who want to save time (and money), and novice programmers who want to create applications that would otherwise be beyond their abilities. It is also perfect for students who prefer to learn by example.

Features & Benefits

- Saves Time and Money
- Increases productivity
- Excellent ROI (Return On Investment)
- Produces Valid XHTML 1.0 Strict Code
- Code is natural and easy to understand / extend / modify / maintain
- Produce Fully Functional Ajax Applications in Minutes
- White Label, Royalty Free Applications – Your clients will think you were up all night programming!
- Fully Customisable Templates – Modify Look+Feel
- Translatable into other languages for International Applications.
- Optimised for MySQL – no middleware to slow things down!
- Create applications for unlimited clients!
- 10th generation tool – the product of 8 years of on-going development

No-Nonsense License – NO HARDWARE LOCKING!

As a developer myself, I know how frustrating it is to spend money on development tools, then find that you can't use them on a client's system or your laptop because it is locked to a single pc.

Therefore a single-user license of *PHP/MySQL Web Database Application Code Generator* allows you to use it on as many workstations as you own – just don't give copies of your license to friends or colleagues – we need the revenue to justify continued development.

A site license allows *PHP/MySQL Web Database Application Code Generator* to be used on all the workstations owned by a company, including employees home PCs, laptops etc.

End User License Agreement – The Boring Legal Stuff!

Please Note: FCKEditor rich text editor is NOT part of *PHP/MySQL Web Database Application Code Generator*. It is open source software which is supported by it. It is generally free, but please consult the licence details in the fckeditor folder, or on their website: http://www.fckeditor.net/ for confirmation.

This license agreement governs PHP/MySQL Web Database Application Code Generator, ('Software'). By installing this software you enter into the terms of this binding contract between you ('you' or 'User') and City Business Logic Limited. If you do not agree with the terms of this license, discontinue its use and uninstall it. Installation of the Software constitutes acceptance of the terms of this License Agreement. The Demonstration version (without license details) may be freely distributed. The following applies to the licensed version of this software.

Grant of License: Subject to the terms and conditions of this Agreement, City Business Logic hereby grants you a nonexclusive license to install and use the object code version of the Software, a copy of which is provided herewith, on multiple personal computers.

Limitations: The Software is licensed, not sold, to you. You must retain all copyright and related notices of City Business Logic ownership and other rights in the Software in the product, labeling and documentation provided. Furthermore, you may not: (a) modify, translate, de-compile, reverse engineer, disassemble or otherwise decode the Software; (b) copy any of the Software other than as reasonably required for your own personal use of the Software in accordance with this Agreement; or (c) sublicense, sell, rent, lend, transfer, post, transmit, distribute or otherwise make the Licensed Software available to anyone else, except that you may permanently transfer the Software and accompanying materials provided you retain no copies and the recipient agrees to the terms of this Agreement. Trademarks: You acknowledge that City Business Logic, PHP/MySQL Web Database Application Code Generator and related logos and designs are © City Business Logic Limited and its licensors, and that no rights in such trademarks are granted to you by this Agreement.

Limitation of Liability: IN NO EVENT WILL CITY BUSINESS LOGIC BE LIABLE FOR ANY DAMAGES, INCLUDING LOSS OF DATA, LOST OPPORTUNITY OR PROFITS, COST OF COVER OR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, DIRECT OR INDIRECT DAMAGES ARISING FROM OR RELATING TO THE USE OF THE SOFTWARE, HOWEVER CAUSED ON ANY THEORY OF LIABILITY. THIS LIMITATION WILL APPLY EVEN IF CITY BUSINESS LOGIC HAS BEEN ADVISED OR GIVEN NOTICE OF THE POSSIBILITY OF SUCH DAMAGE. THE ENTIRE RISK AS TO THE USE OF THE SOFTWARE IS ASSUMED BY THE USER. BECAUSE SOME STATES DO NOT ALLOW THE EXCLUSION OR LIMITATION OF LIABILITY FOR CERTAIN INCIDENTAL, CONSEQUENTIAL OR OTHER DAMAGES, THIS LIMITATION MAY NOT APPLY TO YOU.

Disclaimer of Warranty: TO THE EXTENT PERMITTED BY APPLICABLE LAW ALL CITY BUSINESS LOGIC SOFTWARE, INCLUDING THE IMAGES AND/OR COMPONENTS, IS PROVIDED "AS IS" AND WITHOUT EXPRESS OR IMPLIED WARRANTY OF ANY KIND BY EITHER CITY BUSINESS LOGIC OR ANYONE ELSE WHO HAS BEEN INVOLVED IN THE CREATION, PRODUCTION OR DELIVERY OF SUCH SOFTWARE, INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTY OF MERCHANTABILITY, NONINFRINGEMENT OR FITNESS FOR A PARTICULAR PURPOSE. NO COVENANTS, WARRANTIES OR INDEMNITIES OF ANY KIND ARE GRANTED BY CITY BUSINESS LOGIC LIMITED TO THE USER.

Termination: If you do not accept the terms of this license, you agree to destroy all copies of the Software in your possession and control.

Government Rights: If used or acquired by the Government, the Government acknowledges that (a) the Software constitutes "commercial computer software" or "commercial computer software documentation" for purposes of 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-3, as applicable and (b) the Government's rights are limited to those specifically granted to you pursuant to this License. The contractor/ manufacturer is CITY BUSINESS LOGIC LIMITED. Export Control Obligations: You will not export or re-export any Licensed Software in violation of any law, regulation, order or other governmental requirement (including, without limitation, the U.S. Export Administration Act, regulations of the Department of Commerce and other export controls of the U.S.). You shall, at your own expense, promptly obtain and arrange for the maintenance of all non-U.S.A. government approvals, if any, and comply with all applicable local laws and regulations as may be necessary for performance under this Agreement.

FAQ – Frequently Asked Questions

Can I redistribute an application I created with PHP/MySQL Web Database Application Code Generator?

Yes, Royalty Free!

Previous versions supported Classic ASP, SQL Server and Access. Why was this support discontinued?

In 2001 I wrote my first Web Application Code Generated which produced Classic ASP applications for Microsoft Access – hard to believe that it was a popular combination back then.

Over time, due to changing trends and customer requests, support was added for SQL Server, MySQL and PHP. However, in recent years Classic ASP users have generally migrated to ASP.Net or joined the PHP camp.

Whilst you can use SQL Server and Oracle with PHP, the most popular database for use with PHP is MySQL.

Version 7 used middleware which allows you to write database code without worrying whether the underlying database was MySQL or SQL Server or Oracle (or even Microsoft Access!) However there was a downside, in the form of a substantial performance penalty.

Therefore, it was decided to focus entirely on the PHP/MySQL combination – ensuring optimised code with less clutter.

Can PHP/MySQL Web Database Application Code Generator be used to create Blogs / Forums / Shopping Baskets / social networking sites … ?

No, there are lots of excellent free programs available in all these categories that you can download and install into your site.

Support / Development

After reading the FAQ section you still have any problems, questions or observations then please create a trouble-ticket on our website and we will get back to you asap.

We can undertake development and design work at competitive rates – please contact us for a friendly, no-obligation quote.

System Requirements

PHP/MySQL Web Database Application Code Generator will run on any PC with at least 512mb Ram, 20mb free disk space, running Windows 2000, XP or Vista.

In order to run your generated applications locally, it is recommended that you install MySQL, PHP and either IIS or Apache web server on your PC.

IIS comes with Professional editions of Windows 2000 / XP, and Business and Ultimate versions of Vista.

If you have a different version of Windows then simply install Apache which is free.

MySQL: http://www.mysql.com

PHP: http://www.php.net

Apache: http://httpd.apache.org/download.cgi

WampServer (**W**indows **A**pache **M**ySQL **P**HP) will install all of them for you:
http://www.wampserver.com/en/download.php

You will probably want a front-end for MySQL to make it easier to create your database. I generally use:

DB Tools: http://www.dbtools.com.br

phpMyAdmin: http://www.phpmyadmin.net/home_page/index.php

GUIDED TOUR

In order to understand the process of creating an application in PHP/MySQL Web Database Application Code Generator, lets look at the steps involved in the demo project I used for the website.

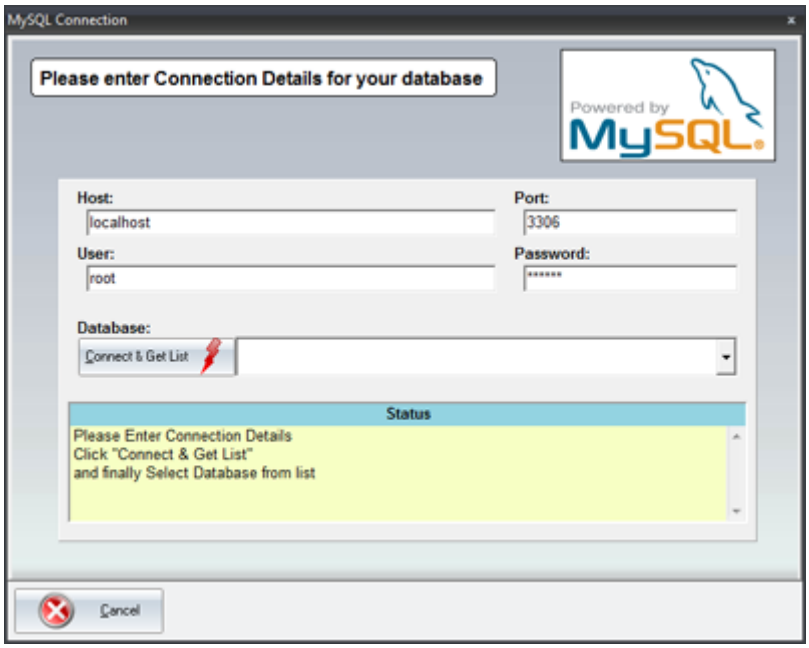
I chose a DVD database for the example as DVDs are something that most people are familiar with, along with concepts such as genre, director and so on.

Ok, Lets get started!

Connecting to your Database

1. Launch PHP/MySQL Web Database Application Code Generator, and click the

button.
2. You will be prompted to enter details to connect to your database. If connecting to a MySQL database on your local pc then leave the host as localhost, and port as 3306. Enter the correct username and password.



3. Click

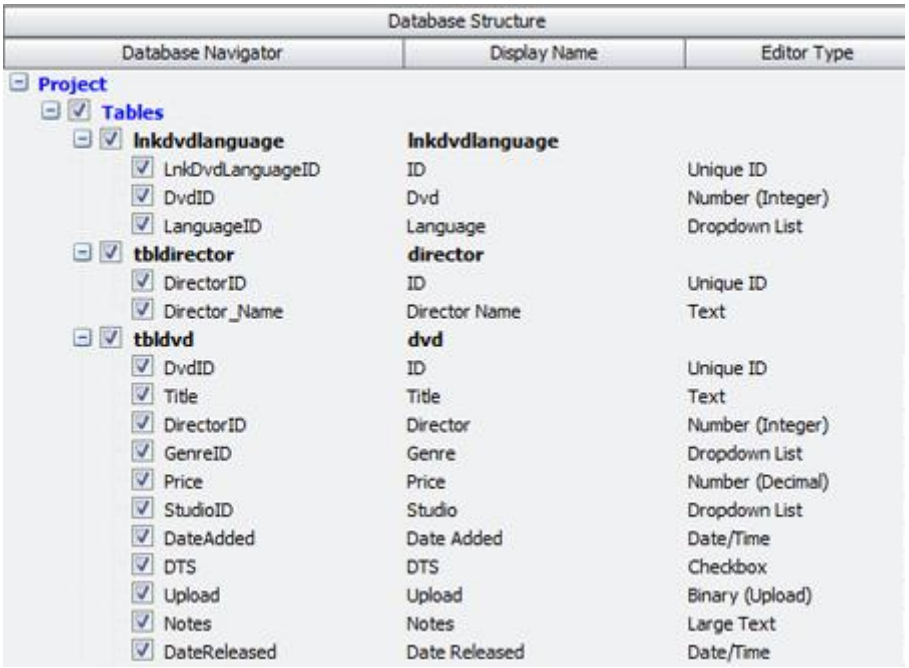
button. If you entered your connection details correctly then the listbox will show a list of your databases. Click the button

4. Click the

button to close the database connection window.

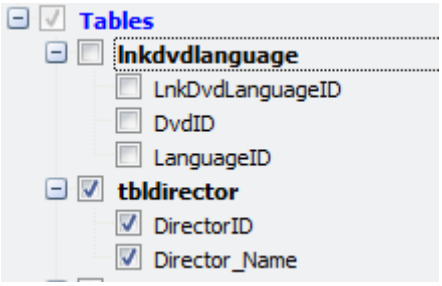
Choose Options

1. After connecting to the database, it will display a tree-view of the database structure, grouped by table.



Database Structure		
Database Navigator	Display Name	Editor Type
Project		
Tables		
lnkdvdlanguage	lnkdvdlanguage	
LnkDvdLanguageID	ID	Unique ID
DvdID	Dvd	Number (Integer)
LanguageID	Language	Dropdown List
tbldirector	director	
DirectorID	ID	Unique ID
Director_Name	Director Name	Text
tbldvd	dvd	
DvdID	ID	Unique ID
Title	Title	Text
DirectorID	Director	Number (Integer)
GenreID	Genre	Dropdown List
Price	Price	Number (Decimal)
StudioID	Studio	Dropdown List
DateAdded	Date Added	Date/Time
DTS	DTS	Checkbox
Upload	Upload	Binary (Upload)
Notes	Notes	Large Text
DateReleased	Date Released	Date/Time

2. Initially all tables / fields are included. If you don't want to generate a data-entry / search form for a particular table, or there is a field that you don't want to be visible, simply remove the tick checkbox. When unselecting a table, all fields in that table will automatically be de-selected.



3. The Database Navigator column shows the actual table/field names from the database. The Display Name column contains the label that will be applied to the table / field. By default, It tries to clean up the field name to make it more user friendly. E.g. removing ID from a field called GenreID, to display the field to the user as Genre. Of course, it can't guess every possible cryptic fieldname, so these labels can be edited by you. For your first project don't get too bogged down with these kind of details – you can always go back and regenerate the application later.

4. The Editor Type column shows the available editors available depending on the underlying data type of the field. For example an Integer field can be a Dropdown List (populated from another table, or a hard-coded list of values), a checkbox, a radio-button list or an integer field. Properties has additional options, for example if you choose Number (Integer) as the editor type, there is an additional option "Linked to a record in related table" which when checked lets you select from an Ajax auto-complete list (which offers suggestions as you type) or a Search Form – which lets the user use search criteria to locate the lookup record. We will look at the various types of editors more closely in the next section.

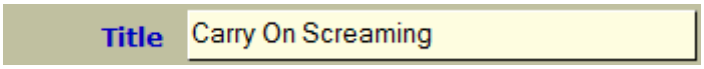
5. The program automatically tries to detect foreign keys – if it misses any, or has found a false-positive, you can easily correct the settings it has chosen. We will look at lookups in more detail in the next section.

6. Required Value checkbox determines whether or not the value can be left blank. It will be ticked by default if the field does not allow null values.
   Search Criteria checkbox determines whether or not the user can search on this field. For example, an end user may be unlikely to know the ID value of a record – so including the ID field on the filter list is just cluttering it up.
   Grid checkbox determines whether or not the field appears on the search results grid. Avoid including too many fields otherwise the user will have to scroll to see the whole grid.

Editors

There are many different types of editors for different types of fields. This is a summary of what each editor does, and when to use it.

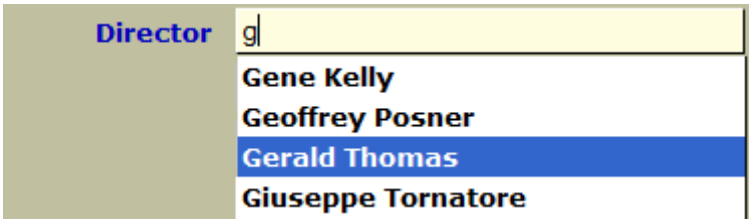
Standard Text Box



Editor Type: **Text** Properties -> Type: **Standard Text Box**

The Standard Text Box is a simple, familiar text editor.

Auto-Complete



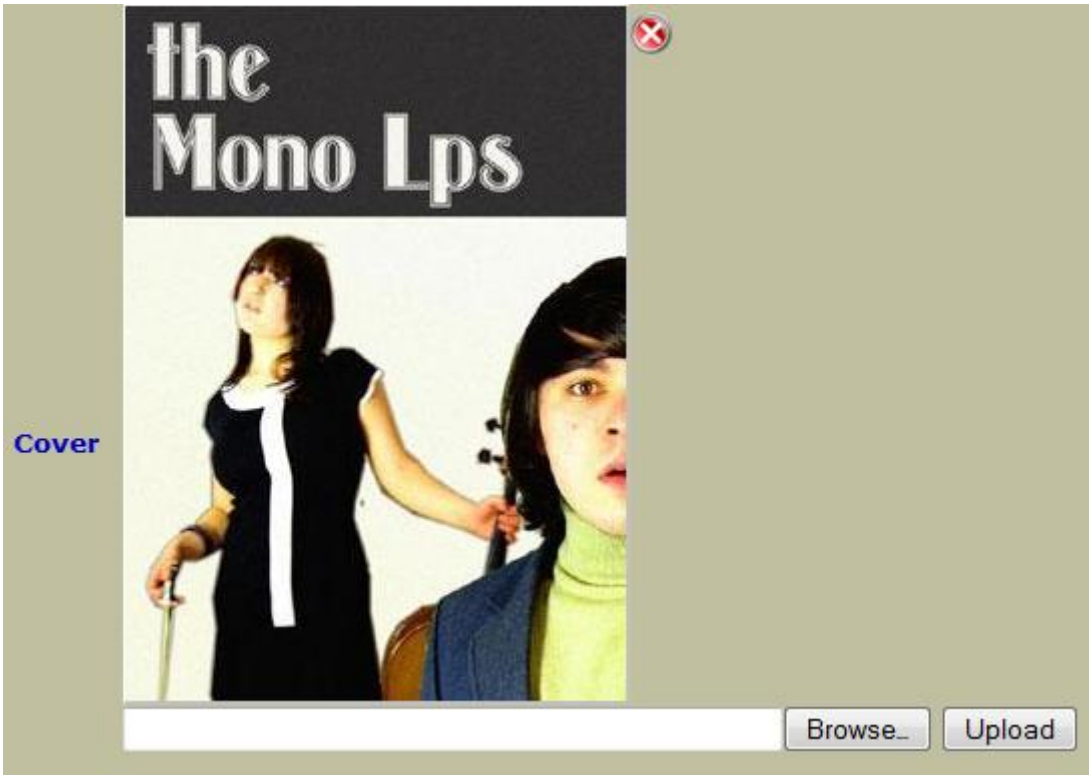
Editor Type: **Text** or **Number (Integer)** Properties -> Type: **Auto-Complete**

The Auto-Complete editor uses Ajax calls to display matching suggestions as you type. When the users sees the value they want, they simply click on it instead of having to type it in full.

The suggestions are populated from another table in the database, defined under *Table to Lookup*.

Upload Fields

Editor Type: **Text** or **Binary (Upload)** Properties -> Type: **Upload File**



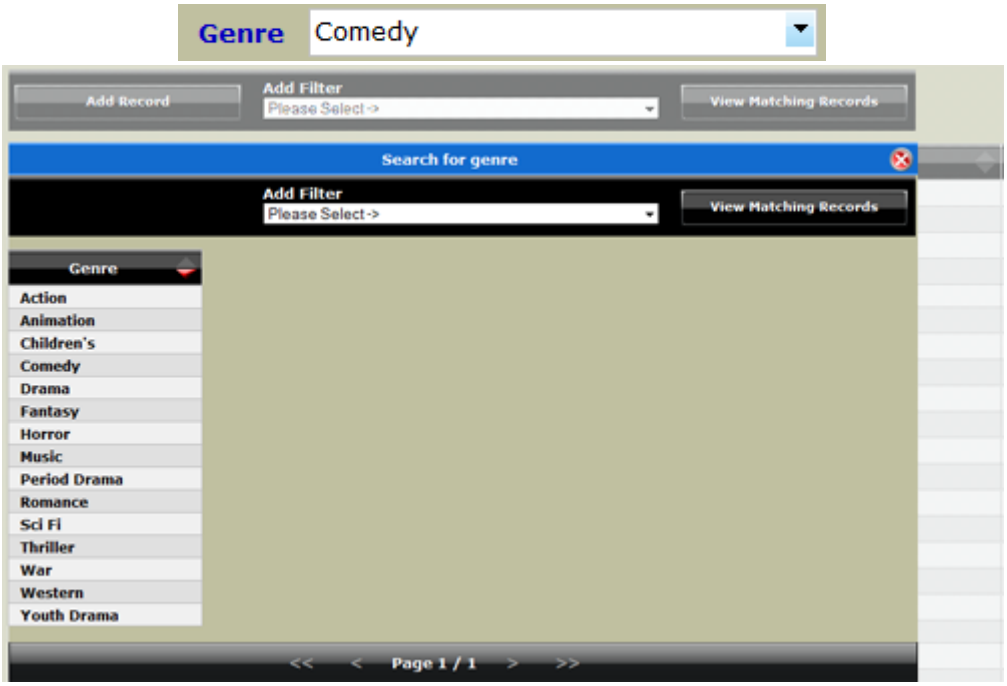
Editor Type: **Text** or **Binary (Upload)** Properties -> Type: **Upload Image**

You can upload images and files, and either store them on the file system, with the filename being stored in a text field, or the file can be stored in a blob (Binary Large Object) field.

Most people recommend storing files on the file system, the ability to store files in the database is only included for completeness.

Image uploads display the image, whilst files are represented by an icon graphic – the user can click on it to download it.

Search-Form Lookup

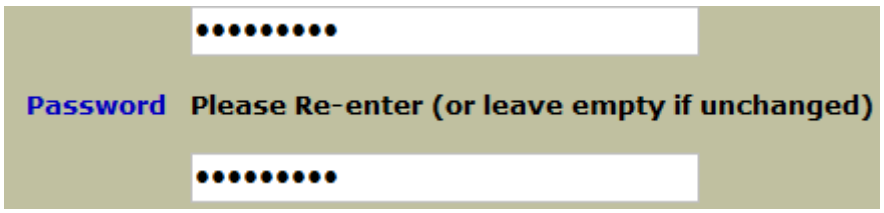


Editor Type: **Text** or **Number (Integer)** Properties -> Type: **Search Form**

The Search-Form field looks like a dropdown list, but when the user clicks on the arrow, it brings up a filterable search form to help the user locate the record they want.

This is particularly useful for customer tables for example. If you have 10,000 customers you don't want to list them all in a drop down list! Using a Search-Form would allow the user to search by First Name, Second Name, Post/Zip Code etc.

Password Field



Editor Type: **Text** Properties -> Type: **Password**

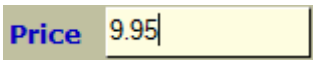
If you have a table of user records in your database which is used to authenticate users then it is a good idea to set the password field to the password editor. This will ensure that the actual password is not displayed (dots will be shown instead) and the user will be prompted a second time to confirm their choice of password.

You can choose to encrypt the password using either ther SHA1 or MD5 functions. These encrypt the password so that if a hacker gained access to your database, they could not guess the password just from seeing the stored value.

Hackers also use databases of common passwords encrypted so that they can possibly work out the password from the encrypted value. Therefore, you can add a salt value to the field before it is encrypted. The Salt is a random phrase or characters – keep it secret to keep your database secure.

Incidentally, the number of dots in the password field do not reflect the number of characters in the actual password entered. Since it has been encrypted, the value held will always be the same length. If you view source you will see that the value held is "UNCHANGED".

Integer / Decimal Field

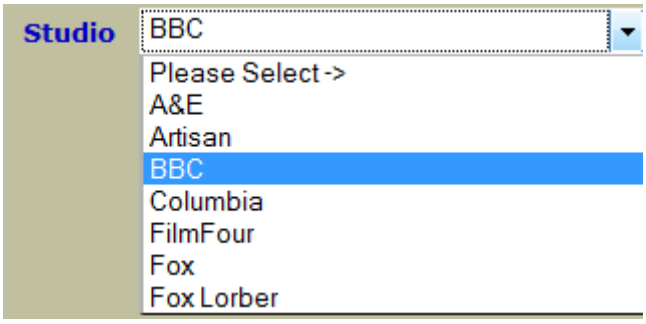


Editor Type: **Number (Integer)**

Editor Type: **Number (Decimal)**

The integer and decimal fields uses a javascript function to prevent the user from entering alphabetic characters. The decimal version allows you to specify the number of decimal places.

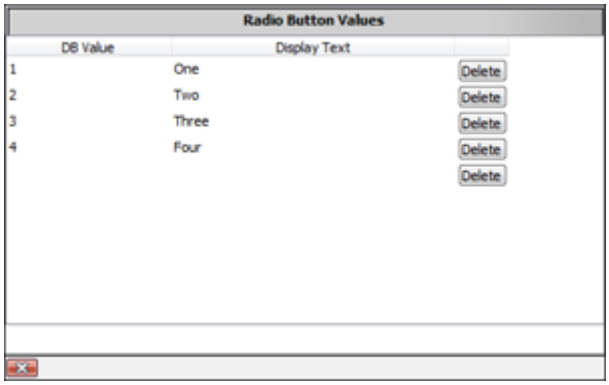
Dropdown List Field



Editor Type: **Dropdown List**

The dropdown list field can be populated from a database table, or from a hard-coded list of values which is held in an XML file.

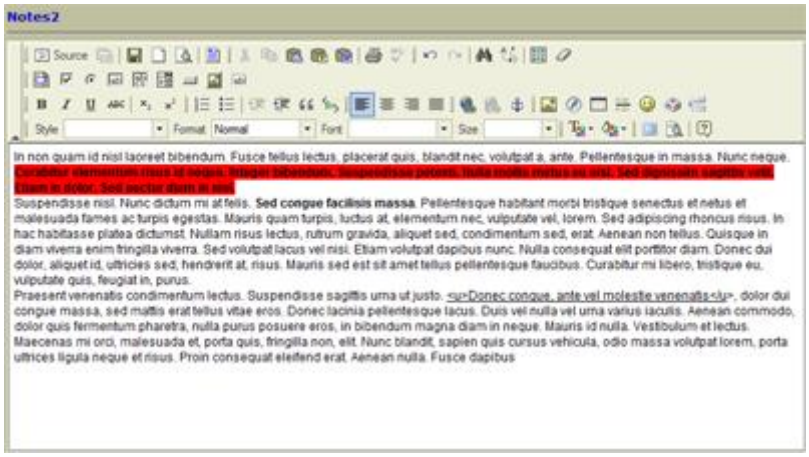
Radio Button Field



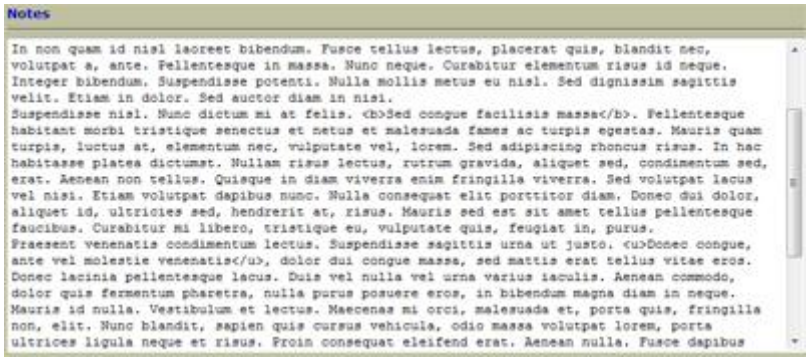
Editor Type: **Radio Button**

The Radio Button editor shows the list of options specified in a hard-coded list of values which is held in an XML file.

Large Text Field



Editor Type: **Large Text** Properties -> Use HTML Editor: **Ticked**



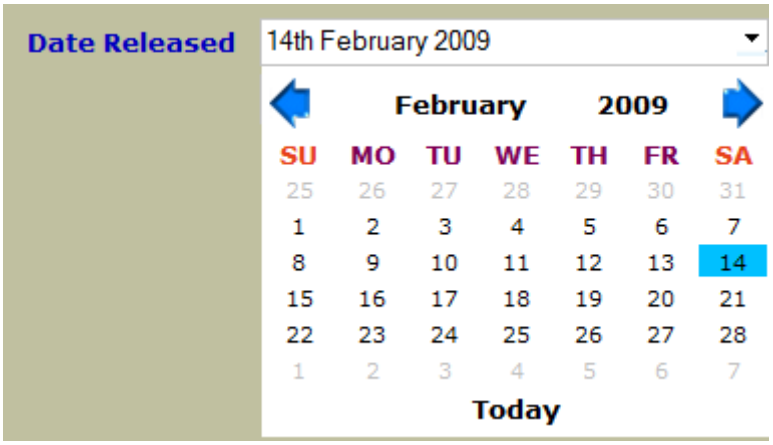
Editor Type: **Large Text** Properties -> Use HTML Editor: **Un-ticked**

The Large Text editors are used for text fields in the database and allow large quantities of text to be edited.

You have a choice of plain-text or HTML Editor – which links into the FCKEditor open source Rich Text Editor which contains facilities to change typeface, size, colour etc.

Please Note: FCKEditor is not part of PHP/MySQL Web Database Application Code Generator, it is supported for convenience. If you have any problems with the editor then please check the forums at http://www.fckeditor.com

Date Field (Calendar)



Editor Type: **Date/Time**

The date editor allows the user to type the date and/or select it from a pop up calendar. It looks at the end users browser country settings and displays the date month and year in the appropriate order for that country.

Checkbox Field

Editor Type: **Checkbox**

The Checkbox field is useful for true/false (tinyint) values.

Dual List Field

Editor Type: **DualList**

Dual List Editors allow you to handle many-to-many relationships.

For example, in our demonstration DVD database, we want to record the languages available on the DVD – but a normal dropdown list isn't any good because DVDs often have more than 1 language soundtrack.

The DualList shows all the available values on the left, and the selected ones on the right, with buttons to transfer values between the lists.

When there are too many "available" options to show them all, there is an option to combine it with auto-complete so that the user types in a few letters from the value they want and a list of matching options will be shown.

How does it work?

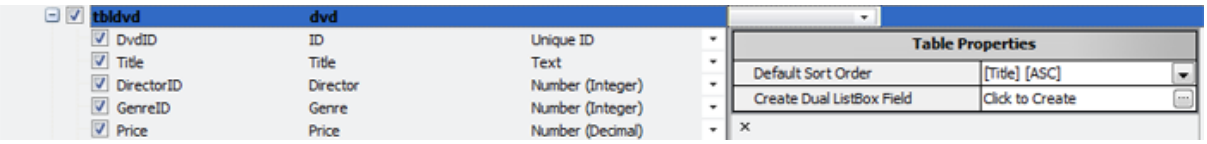
To associate multiple lookup values in this way, we need a "middle" table – a table that is simply used to link the other 2 tables.

In our example database, we have a table of DVD records called tbldvd, with an identity field called dvdid. There is a language table called tbllanguage with an identity field called languageid. The middle or link table is called lnkdvdlanguage and contains the fields dvdid and languageid.

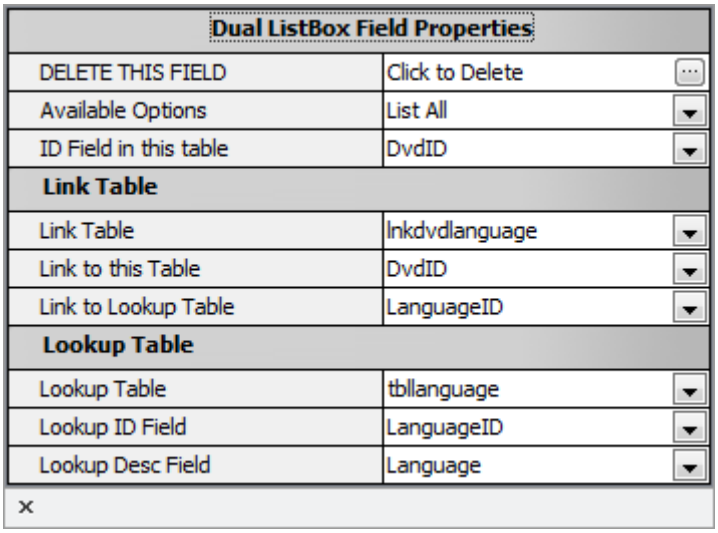
It may sound complicated if you don't have a lot of experience with databases, but it's really quite easy once you get your head around it!

How to create a Duallist Field?

To create a Duallist field, first select the table where you want the field to appear. In this example, tbldvd and click the “Create Dual ListBox Field” … button.



It will add a new field to the end of the table called Duallist 1. Go to the properties column and set the values.



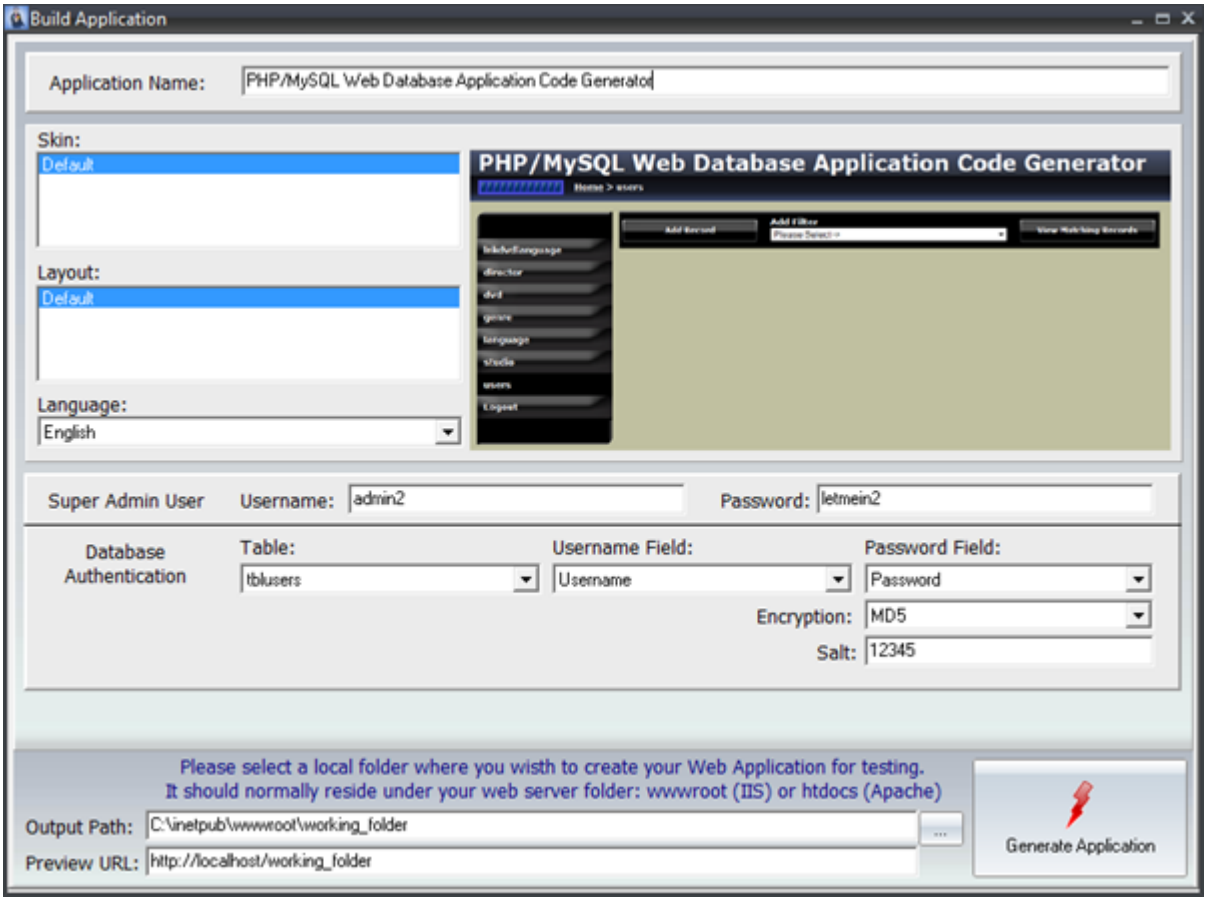
Available Options determines whether it should list all available options or provide an auto-complete field.

The remaining options determine the tables/fields to use.

Generate Application

When you are finished tweaking options and ready to generate and try your Web Application, click the

button.



Application Name is the title that appears at the top of each page in the Web Application.

You can create your own **Skin** files, **HTML Layouts** and **Language Translation** Files (more on that later) for now use the defaults.

Every application should have a hard-coded Super Admin user that can login even if the authentication table in the database is empty.

If you wish to authenticate users against a database table, select the **Table**, the **Username Field** and **Password Field**, along with the **Encryption** used (if any) and the **Salt** (a random collection of letters and numbers to make it more difficult for hackers to get your passwords).

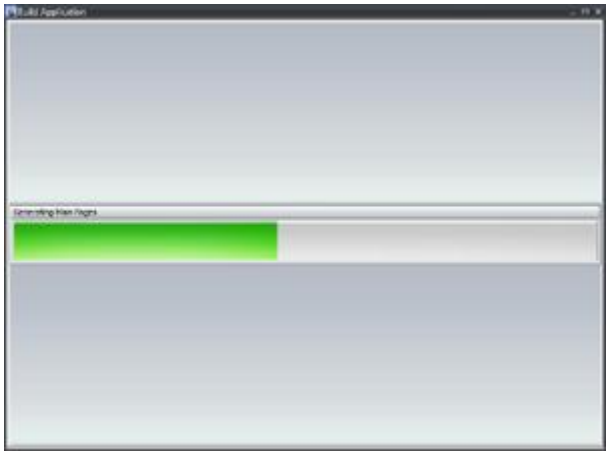
Output Path is the folder where you want to create your application. Ideally it should reside under your Web Server home folder.

Preview URL is the URL to use when previewing your application. If the output path is under wwwroot (Default IIS root folder) or htdocs (Default Apache root folder) then the program will try to figure out the URL from that. If you renamed your web server home directory then you will need to enter the Preview URL manually.

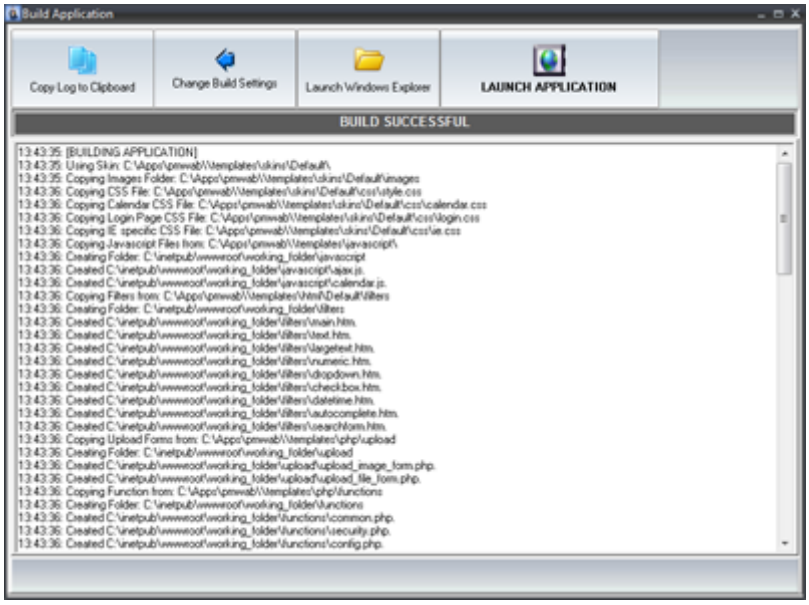
When you are happy with all the settings, click

button.

If everything is ok you will see a progress bar:



When complete a report screen will be shown, summarising actions taken to generate the application.

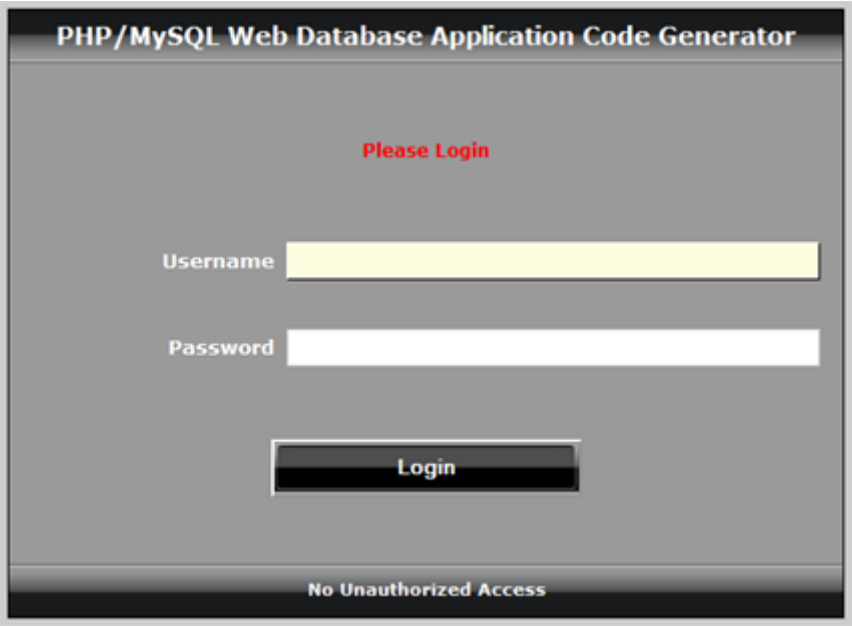


If there are any errors they will be listed, otherwise it will say “BUILD SUCCESSFUL”. Click the

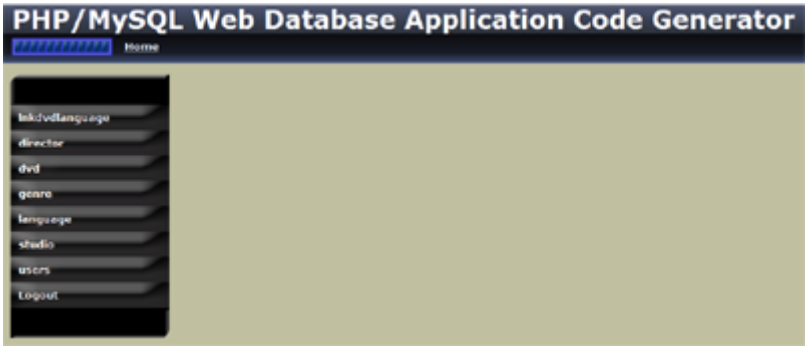
button. Your freshly brewed web application will launch in your default browser.

Generated Application Features

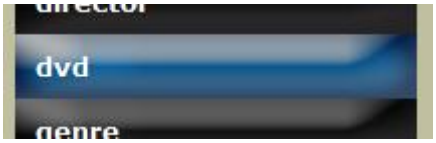
The first page of the web application you will encounter is the login page. Attempting to go directly to another page of the application without logging in first will redirect the user back to the login page.



Once successfully logged in, you will see the main homepage.



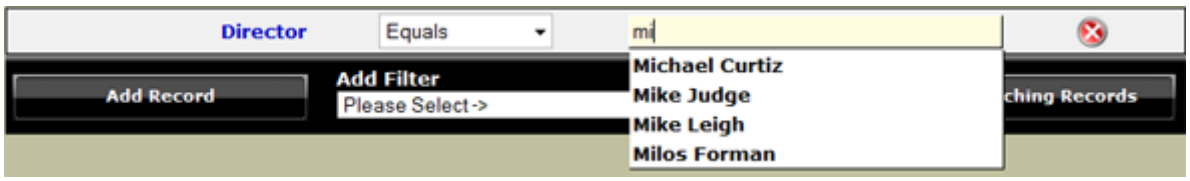
The topbar shows the title of the application, the Ajax activity indicator, a “bread crumb” trail showing the current page, and the menu which takes you to the tables in your application.



After selecting one of the tables, you will be presented with the top control bar.

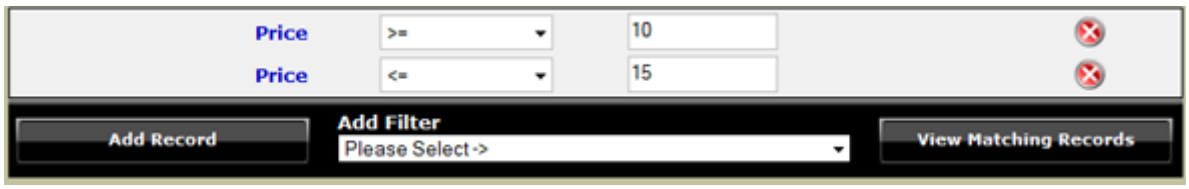
It contains a button to create a new record, and a filter selection box with a list of fields that the current table can be filtered by.

Selecting a filter will prompt you for the operator (e.g. Equals, Not Equals, Sounds Like, etc) and the value.

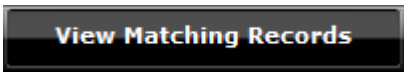


The editor used for the filter value will be the same as the one chosen for the data entry forms.

The user can select as few or as many filters as they require. For example, if you wanted to see items with a price between $10 and $15 you would select the price field twice and enter the appropriate criteria.



Filters can be removed by clicking the button. After selecting the criteria click the



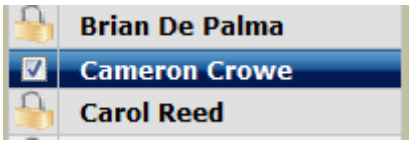
button.

The columns can be sorted by clicking on them. First click means ascending, next click means descending, next click means unsorted. Multiple columns can be selected, simply select them in the order you wish the data to be grouped by.

A default sort order for a table can be selected before you generate the application. To do so, click on the relevant table in the treeview, go to properties and select “Default Sort Order”.

At the bottom of each page is a page navigator allowing the user to tab through pages of results.

To delete a record, click on the checkbox(es) of the records you wish to delete,



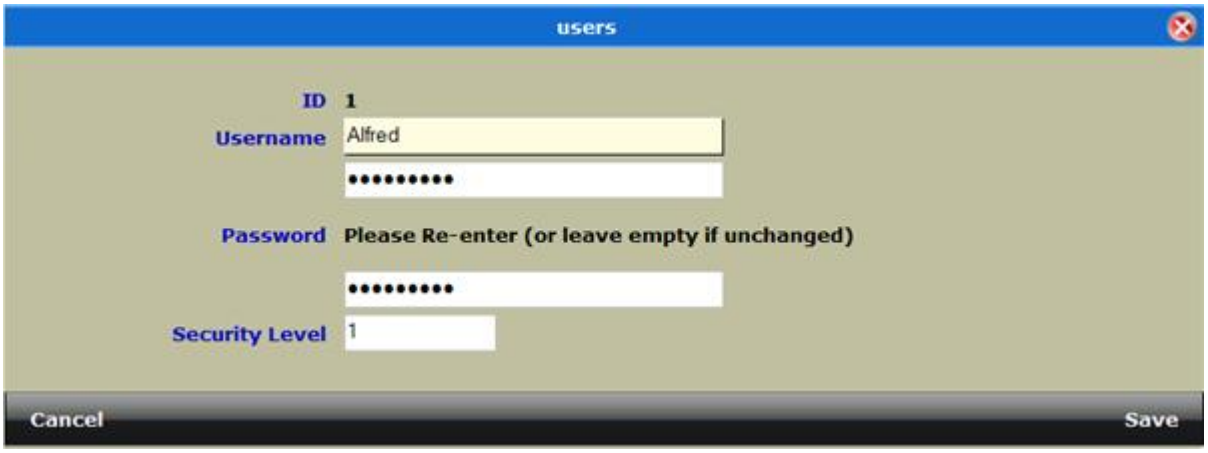
To select all records on the page, click the checkbox at the top of the grid.

and click the

button on the page navigator at the bottom of the screen.

Records with a padlock cannot be deleted as they are the detail records in a master-detail relationship. To delete them you would need to go to the master table, filter the records on the field linked to this table, delete them and then delete the records in the detail table.

To view a record in more detail simply click on the row. To edit the record double-click on it instead.



Creating your own Skins

You can customise the appearance of the web applications you generate with PHP/MySQL Web Database Application Code Generator.

They consist of CSS files, images and a preview image which shows the end result and is displayed when the skin is selected.

Begin by navigating to your templates folder in Windows Explorer. The normal location is c:\program files\pmwdacg10\templates. Go into the skins folder, duplicate the folder called default and give it a new name.

The CSS files are laid out in a logical order with related items grouped together under a descriptive comment. If you are new to CSS try running your application in Firefox with Firebug installed. http://getfirebug.com/

Click the inspect option in firebug, and highlight an item in your web application. It will show the name of the CSS class and the settings, and even let you change them on the fly. This is a good way of finding where to make the changes that you want to make.

Once you've created your new skin, take a screen dump of it in the browser and size it to 467x200 pixels and save it as a PNG file with the same name as the folder itself. E.g if you called your skin myskin, call the image myskin.png.

You are free to share your skin files with others. I hope to make new skins available soon.

Creating your own Layouts

Most of the HTML of your web application is contained in the html folder under c:\program files\pmwdacg10\templates

To create a revised layout, first duplicate the folder called default, and rename it to whatever you want to call the layout.

Edit_forms and **view_forms** folders contain a file for each type of editor, and the page that ties them all together. If for example you wanted to change some aspect of checkboxes throughout your application, you would edit the checkbox.htm file in the edit_forms folder.

The **filters** folder contains the html used when a filter is selected from the dropdown list.

Menus contains the list item html for each entry in the table navigation menu.

Page_layouts has the main page layouts.

You are free to share your layout files with others. I hope to make new layouts available soon.

Translating your applications into another language

All the text and messages contained in your generated application come from XML files in the c:\program files\pmwdacg10\templates\language folder.

To begin translating, first duplicate the folder called English and name it as the language you wish to create, e.g. Spanish.

Go through each file in turn, either with an XML editor, or using notepad if you don't have one.

Under the heading:

```
<?xml version="1.0" encoding="windows-1252"?>
<root>
  <languagedefinitions>
```

Comes the actual language elements such as:

```
   <definition Key="TEXT-ADDRECORD"
     Translation="Add Record"
     Comment="Used on Add Record button" />
```

The "Key" item is used to identify it in the templates and should not be translated.

"Translation" is where you should translate the English text.

"Comment" is an explanation of the context in which the text is used, and does not need to be translated.

You are free to share your language definition files with others. I hope to make additional ones available in future.

Modifying scripts

When you want to make changes to your generated application, you need to decide whether it is a change you would like to effect all your applications, or just one.

If it is a general change that you would like to automatically be included in every application you generate with PHP/MySQL Web Database Application Code Generator, then it is better to change the templates. But if it is a change that you only want to appear on perhaps one page of one application then it is better to make the changes to the generated pages.

Either way, it is worth keeping track of what you have changed. That way you can re-apply the changes if you want to use a newer version of the core templates.

The PHP code is stored in the c:\program files\pmwdacg10\templates\php folder and the javascript in c:\program files\pmwdacg10\templates\javascript

Uploading your application to a web server.

When uploading your application to a live web server there are a few points you need to consider.

Database Connection Details

The database on your web server may have different connection details to your local database. To change them open /data_access_layer/functions/database.php file and edit the following values:

$db_host
$db_user
$db_pwd
$db_name
$db_port

To reflect the correct connection details for your database.

Upload File Locations

If you have any file upload fields in your database then you need to move the location of the upload folders.

On your local copy of the application there are two folders – **/temp**, which is used for images that have been uploaded, but not yet saved. **/upload** is the folder that files are moved to once the record has been saved. The temp folder is cleared out periodically.

On your live server you should move these folders to a location that is not accessible from the web.

If the root web folder on your web server is located at /httpdocs, then don't put your upload folders at /httpdocs/temp and /httpdocs/upload, instead place them at /temp and /upload.

You then need to edit /functions/config.php and edit the following values:

UPLOADED_FILE_PATH
TEMP_FILE_PATH
FULL_TEMP_FILE_PATH
FULL_UPLOADED_FILE_PATH

The first 2 are relative to the location of the /edit files. So for example, if your paths are like this:

/temp
/uploads
/httdocs/edit

You would set:

```
define("UPLOADED_FILE_PATH", "../../uploads/");
define("TEMP_FILE_PATH", "../../temp/");
```

The "../" tells it to go up one level.

The Full Path variables need to know the full path of the folders. If you don't know the path, create a file called test.php containing the following:

```
<?php echo realpath(substr($PHP_SELF, 1)); ?>
```

Upload the file and view it from your browser by going to http://mydomain.com/test.php which will show you the full path, which will be something like /var/www/vhosts/mydomain.com/httpdocs/

Use this information to set the Full_x paths, e.g.

define("FULL_TEMP_FILE_PATH", "/var/www/vhosts/mydomain.com/temp");
define("FULL_UPLOADED_FILE_PATH", "/var/www/vhosts/mydomain.com/uploads");

Why should uploads not be in web accessible folders?

If a hacker uploaded a PHP script, or found an exploit that tricks PHP into interpreting an image file as PHP then they can give themselves permissions to access your server.

In PHP/MySQL Web Database Application Code Generator applications, a proxy script is used to retrieve images and files from web-inaccessible folders.

© 2001- 2009 City Business Logic – All Rights Reserved