

# Music Cloud Player



Documentation

January 8, 2013

Author: Humain2

## Requirements

### Running the script

- Account at some hosting service, or a
- Webserver installed, Apache is preferred
  - Any Linux distribution: Look for LAMP with PHP
  - Windows or MAC: XAMPP at [www.ApacheFriends.org](http://www.ApacheFriends.org) or MAMP
  - PHP 4 or higher is required.
  - MySQL database required.

### Editing the source-code

- Your favorite text editor
- Knowledge of PHP, Javascript, CSS

## Introduction

Music Cloud Player is one of the best solutions if you want to start your own music portal. It has a beautiful and intuitive interface built with the latest development tools to make it as easy to use for you.

Beside that there are many fine features :

## Features

### User Features:

- Powerful Music Player
- Visitors can be able to search music by album, song, artist
- Top songs, artists, albums information from LAST.FM API
- Playlist management
- Registration
- Multi-user
- Web 2.0 Design - Clean & Modern design
- API integration to grab songs, albums, artists information from LAST.FM
- Optimized for Ipad
- Share song by URL or Facebook
- You can Easily add tracks to your music website. (External or local songs).
- Ajax based / Ajax Upload

- One page application with javascript URL
- Cross-Browser compatible.
- Progress loading pagination
- Mysql Fulltext search

#### **Script/Admin Features:**

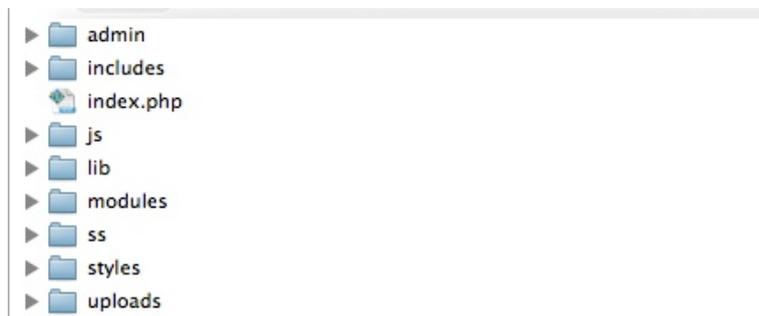
- View/Delete registered users.
- View/Edit/Delete songs
- Upload local song or add external song by URL
- Add/Delete administrators
- View Statistics
- Filter illegal username inside js/main.js (register function)

#### **Customization Features**

- Everything is customizable. You can edit the design, create your own template, edit the header and footer, add your own graphics, whatever you need.
- 100% open source code allows developers to make any sort of modifications needed.

### **File Overview**

The Music Cloud Player is a combination of php, js, css, images. The screenshot 1.1 gives a good overview of the directory structure.



#### **Admin folder :**

The admin directory contains the files for the admin panel. Admin theme is based on the framework Twitter Bootstrap 2.

**Includes folder :**

The directory includes contains configuration files and all PHP class.

**LIB folder :**

The lib directory contains the javascript frameworks files. The player is based on the soundmanager framework and use jquery-ui and YUI.

**ss folder:**

The ss directory contains the PHP files for actions like creating playlists, registration, login ....

**styles folder:**

The style folder contains the css stylesheets.

**uploads folder:**

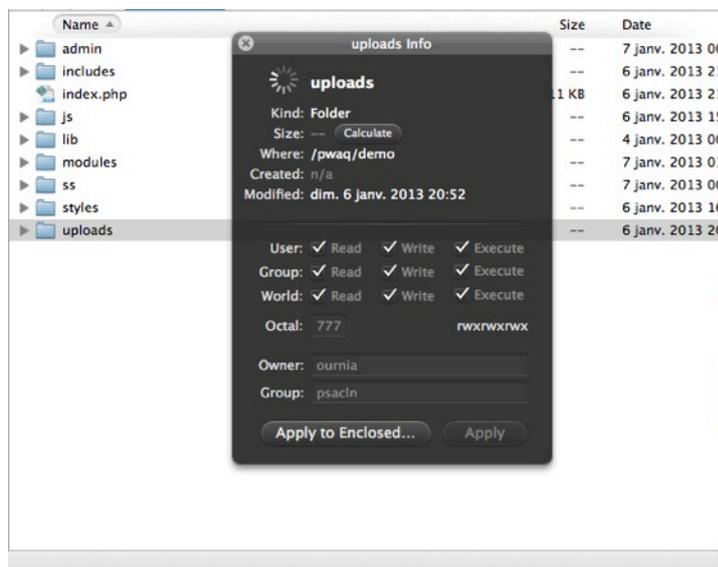
The uploads folder containing MP3 files uploaded by the administrator. This directory must be writable (chmod 777).

## Installation steps

This part outlines all the steps in more detail along with configuration for each step.

First you need to download the script from codecanyon. The package need to be extracted then uploaded under the desired directory for your web hosting account. If you would like the Music Cloud Player website accessible via URL of type <http://www.mysite.com/music> you should upload the extracted files under "public\_html" folder for your account in a sub-folder "music". You can upload the files from your computer to your hosting account using an [FTP Client](#).

The second step it is to make uploads folder writable (chmod 777).



The next step is to create a new database that will be used for the Music Cloud Player. This can be easily accomplished following this [MySQL Tutorial](#).

After you need to import the file music-player.sql to your database using your PhpMyAdmin:

**Importing into the database "musicplayer"**

**File to Import:**

File may be compressed (gzip, bzip2, zip) or uncompressed.  
A compressed file's name must end in **.[format].[compression]**. Example: **.sql.zip**

Browse your computer:   (Max: 32MiB)

Character set of the file:

**Partial import:**

Allow the interruption of an import in case the script detects it is close to the PHP timeout limit. (This might be good way to import large files, however it can break transactions.)

Number of rows to skip, starting from the first row:

**Format:**

**Format-Specific Options:**

SQL compatibility mode:  ⓘ

Do not use AUTO\_INCREMENT for zero values ⓘ

After importation the tables will be added in the database :

Table	Action	Rows	Type	Collation	Size	Overhead
<input type="checkbox"/> vass_playlist		3	MyISAM	utf8_general_ci	6.4 KiB	224 B
<input type="checkbox"/> vass_songs		3	MyISAM	utf8_general_ci	3.4 KiB	-
<input type="checkbox"/> vass_update		0	MyISAM	utf8_general_ci	1 KiB	-
<input type="checkbox"/> vass_users		4	MyISAM	utf8_general_ci	3.2 KiB	-
<b>4 tables</b>	<b>Sum</b>	<b>10</b>	<b>InnoDB</b>	<b>latin1_swedish_ci</b>	<b>14 KiB</b>	<b>224 B</b>

↑ Check All / Uncheck All / Check tables having overhead

The last step is the configuration. You need to configure 2 files (config.inc.php and db.php) :

**includes/config.inc.php :**

```
<?PHP
//Configurations
$config = array (
// Site Title
'sitetitle' => "Music Cloud Player",
// URL
'siteurl' => "http://www.pwaq.com/demo/",
// Charset
'charset' => "UTF-8",
);
?>
```

**includes/db.php :**

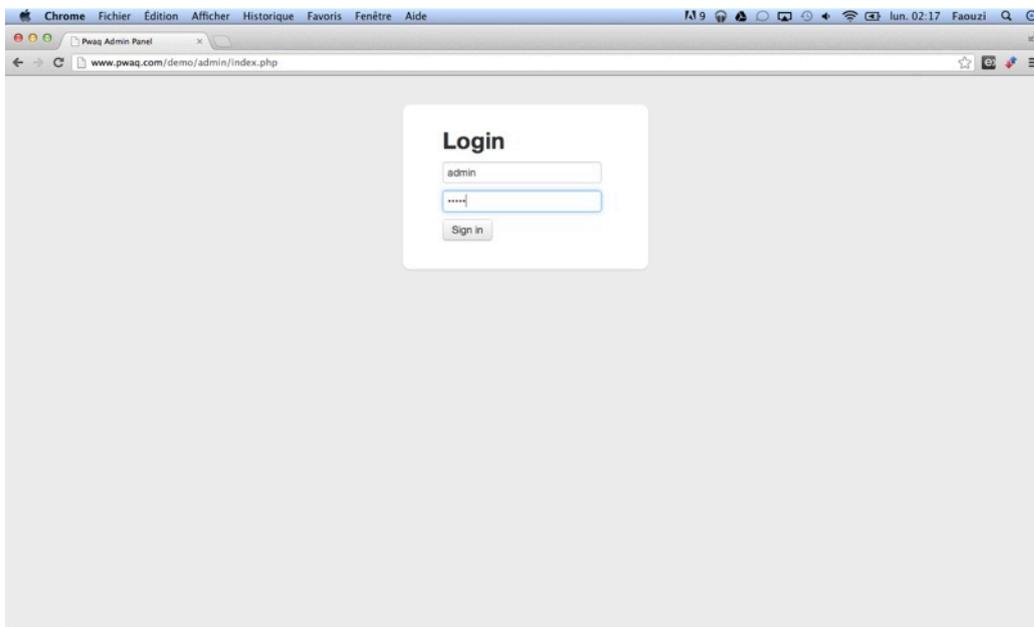
```
<?PHP
// Database server
define ("DBHOST", "localhost");
// Database name
define ("DBNAME", "pwaq");
// Database USER
define ("DBUSER", "pwaq");
// Database PASSWORD
define ("DBPASS", "yourpassword");
// NOT EDIT THIS PART
define ("PREFIX", "vass");
define ("COLLATE", "utf8");
$db = new db;
?>
```

Once you have finalized the Music Cloud Player installation you should login to :

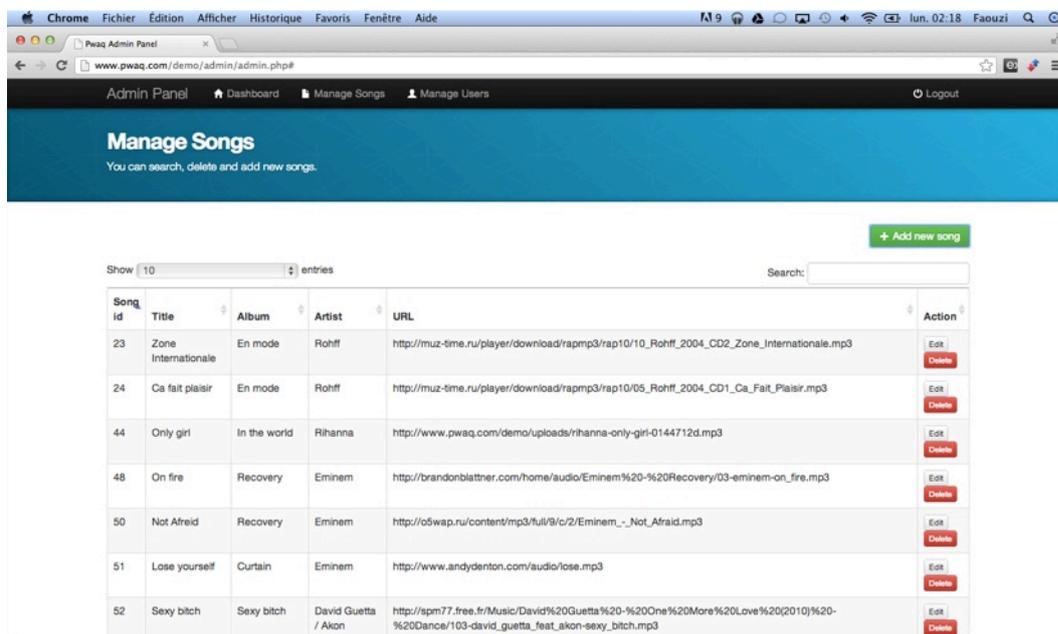
<http://www.yourdomain.com/admin/>

**Login : admin**

**Password : admin**



Once you are logged in as an administrator you can manage your portal.



I hope you enjoy using Music Cloud Player as much as I had fun making it!

Best regards

Humain2