GXC-CMS Documentation

Introduction

GXC-CMS (http://www.gxccms.com) is an open-source cms, written based on Yii framework. Its main purpose is to provide a good foundation for DEVELOPERS to build web apps.

It has been developed by Tuan Nguyen at - http://nganhtuan.com.

If you have any questions/bugs/ideas/feedback while using it, please contact me at: nganhtuan63@gmail.com. Other method is using gxc forum (http://www.yiiframework.com/forum/index.php/topic/28652-open-source-yii-cms-gxc-cms/page__pid__195845__st__200#entry195845) or Github issues (https://github.com/nganhtuan63/GXC-CMS-2/issues)

3rd party libraries used

GXC-CMS uses some 3rd party libraries with some modifications.

1/ App directory structure:
http://www.yiiframework.com/wiki/155/the-directory-structure-of-the-yii-project-site/

2/ Rights Module:
http://www.yiiframework.com/extension/rights/

3/ HttpRequest with CSRF by Session:
http://www.yiiframework.com/wiki/274/how-to-validate-csrf-token-with-session/

4/ PassHash class - support password hashing
http://net.tutsplus.com/tutorials/php/understanding-hash-functions-and-keeping-passwords-safe/

5/ Minify extensions
https://bitbucket.org/TeamTPG/minscript/wiki/Home

6/ Settings component
http://www.yiiframework.com/extension/settings/

7/ Timepicker
http://www.yiiframework.com/extension/timepicker/

8/ Yii-debug-toolbar
http://www.yiiframework.com/extension/yii-debug-toolbar/

9/ Apcinfo Module
http://www.yiiframework.com/extension/apcinfo/

Getting started

1/ Check out the source code at https://github.com/nganhtuan63/GXC-CMS-2 to your machine.

For example: I checked out source code to my machine at:

/Applications/MAMP/htdocs/gxccms

2/ Use command line to generate a new cms app

>_ php /Applications/MAMP/htdocs/gxccms/core/cms/cms.php cmsapp /Applications/MAMP/htdocs/project1

3/ Chmod 777 backend/assets, backend/protected/runtime, web/assets, web/protected/runtime

4/ Create new database in mysql.

5/ Install the new app at with database info.

http://localhost/project1/backend/install

The installer will check the .locked file existed in common folder or not before installing.

GXC-CMS Structure

1/ GXC-CMS core has 2 main folders:

+ **yii**: Yii framework source code
+ **cms**: Gxc-cms source code

With a gxc-cms core, you can use it for multiple apps you generated as shown above.

2/ A generated App has a structure:

--**backend** (Backend/Admin)
--**common** (Share common classes/models/components,..between backend, web,...)
--**web** (Web frontend)
--**resources** (Store uploaded files)

Backend/Web has the structure of a basic MVC application from Yii Framework.

3/ Each folder backend/web has an **environment.php** file for its own configuration.

Things to check:

- Check to make sure CORE_FOLDER constant point to correct core folder.
- Environment.php has multiple environments for you to develop: _development, _test, _stage, _production. Modify the environment based on your needs.
- To switch environment, change line 3 of index.php in each application:

```
$environment = new Environment(Environment::DEVELOPMENT) ;
```

- Remember to make sure application id is the same with the application folder name. For example, the application id of backend folder must be 'backend'. Same rules applied for web folder and mobile later if needed. This is required for rights module to work correctly.

4/ Common folder structure:

- **components**: Component classes shared between apps (backend/web/...)
- **settings:** Classes and Views to manage general/system settings of the application.
- **messages:**

- **storages:** define storage mechanism for application. For example - LocalStorage is a class to upload file to Local Machine (resources folder). AmazonStorage is a class to upload file to S3 if supported.

- **content_type:** define content type for application - you can define multiple content types like article, events, products,...without changing database. You can generate by using Content Generator in Gii

- **layouts:** Manage layouts in application. You can add new layout here, define layout regions.

- **blocks:** Manage blocks to put in regions of layouts.

Notes:

- Using this structure, on production server, you need to change host file to each folder. For

example:

http://backend.project1.com -> point to backend folder.
http://project1.com -> point to web folder.
http://resources.project1.com -> point to resource folder.
…

- If you are not allowed to change host file, so you have to copy inner content of web content to the root folder (same level with backend,...) and modify **environment.php** to **point to correct core** folder.

- If you want to create a mobile version for your site, just create a new folder called mobile and clone source from web folder to it.

- Remember to clear cache if you change something but not see.

GXC-CMS Features

1. Content Management

*Folder **common/content_type** manage all content types of the application.*

+ Dynamic Content Type

- GXC is built to help you create dynamic content type. It comes with already defined content types: Article, Event (with start-date, event-date attributes), Product (with price attribute).

- To generate a new content type, go to backend/gii to generate.

- To add new field/attribute, check **EventObject.php** or **ProductObject.php** and its form render **object_form_render.php** in each folder for more information and how to define your own.

+ Dynamic Resource Binding

- You can easily add thumbnail, video, audio,..or any kind related to file to your content by defining the resource you want.

- Check **ArticleObject.php, Object.php - public static function Resources()** for more information

+ Dynamic Content Workflow

- Just like the Resource, content allows you to modify the workflow you want. (Who can create, edit, delete, publish,...)

- Check **Object.php - public static function Permissions()** for more information.

+ Term and Taxonomy

2. Resource Management

*Folder **common/storages** manage all storage mechanism of the application.*

- Gxc allows you to define your own storage mechanism you want (save resource to local, to other servers, or to Amazon S3,...).

- Check LocalStorage.php to implement based on your own.

- If you need to modify the place where you want to store resource file on local, check constant **RESOURCE_URL** and **RESOURCES_FOLDER** in **environment.php** file

3. User Management

- User component files are: **GxcUser.php** and **CDbUserIdentity.php**

- Integreate with Rights module - Here, gxccms modified rights a little bit to support its current app structure and it uses a cache file to improve the application speed.

- So when you update permissions with Rights, remember to click **Update Permission Cache File.**

4. Settings Management

*Folder **common/settings** defines settings used in application.*

- Settings uses **CmsSettings.php** extension.

- If you need to add/remove any settings, just modify the **XSettingForm.php** and its form render **x_settings_form.php (x is general or system).**

5. Multiple languages Management

*Folder **common/messages** manages all messages/languages of the application.*

- To add a new language to your site, just create a new folder with the format like **en_us** folder.

- If you want your application to support multiple languages. Just create a new language folder here. Clone info.ini to add information for the new language. To generate all messages of the apps, uses:

php /Applications/MAMP/htdocs/gxccms/core/yii/yiic.php message /Applications/MAMP/htdocs/project1/common/messages/config.php

6. Page Management

*Folder **common/layouts** manages all **frontend layouts** of your application.*
*Folder **common/blocks** manages all **frontend blocks** of your application.*

Layouts

- To create a new layout, clone the default folder.

- Put your own asset files into the new layout/assets folder (css,images,js,...)

- Define your layout information by info.ini

id - must be the same with the layout folder name.
types - the layout types (1 column, 2 column, blank,...) - it must be the same with the php file of layout.
regions - the regions that the layout support.

- A page in CMS can select a layout to apply to it.

- In **web** folder application, page can be accessed by using **PageController.php.** Homepage will use the **default homepage slug from settings** to render.With this structure, you can implement your own controller in frontend.

Blocks

- Block is a kind of Widget in Yii. It supports a way for you to supply content from backend and the render in frontend. For example, Html Block is a block that allows you to add a bunch of html content in backend and render that content in frontend.

- To create a new Block, use backend/gii to generate a new Block.

- **x_block_input.php :** view file to render form to add content in backend.
- **x_block_output.php :** view file to render block content in frontend.

- **XBlock.php** : class to handle to logic code of the Block.

Menu

- Use Menu to create Menu items in your application.

- To render the menu, use **menu** block or create your own block.

Content List/Queue

- Content list is a way to retrieve a list of content based on content type, content term, tags,...

- You can select contents in manual mode or auto mode.

- To render the content list in frontend/web, use **listview** block or you can create another block based on your demands.

7. Cache Management

- Cache is handle by Cache module, you need **supply password** for the cache module in environment.php

- Access cache module management at **http://yoursite/cache**

- Remember to clear cache if you change something but not see.

GXC-CMS Development Workflow

1. Design and HTML/CSS.

2. Create new layout in common folder or modify the default layout directly.

3. Check and define regions/layout types based on the design.

4. Create Block to handle the render content for each region.

5. Implement Caching Strategy if needed before deploying.

Questions/Ideas/Problems?

If you have any questions/bugs/ideas/feedback while using it, please contact me at: nganhtuan63@gmail.com. Other method is using gxc forum (http://forum.gxccms.com/) or Github issues (https://github.com/nganhtuan63/GXC-CMS-2/issues)