

XOOPS Documentation Series



XOOPS 2.5.x Standard Module Structure Guidelines

Draft 1

All About XOOPS:

XOOPS is a Content Management System (CSM) and a Web Portal program that allows administrators to easily create dynamic websites with great content and many outstanding features. It is an ideal tool for developing small to large dynamic community websites, intra company portals, corporate portals, weblogs and much more. It can be installed on an Internet host with a PHP-capable web server (e.g., Apache) and a database (e.g., MySQL).

The main Project websites is www.xoops.org

XOOPS is released under the terms of the GNU General Public License (GPL) and is free to use and modify. It is free to redistribute as long as you abide by the distribution terms of the GPL.

What XOOPS stands for:

XOOPS is an acronym of eXtensible Object Oriented Portal System. Though started as a portal system, XOOPS is in fact striving steadily on the track of Content Management System. It can serve as a web framework for use by small, medium and large sites.

It is highly scalable, from a personal weblog or journal through a community website to an enterprise installation. It has numerous modules that can be used to create all the basic functionality required of the system, e.g. News, Forums, Downloads, Web Links etc. Customized modules can also be developed by users and integrated seamlessly with the core to serve specific needs such as eCommerce.

Pronunciation guide

The standard pronunciation of XOOPS follows English rules, and is pronounced [zoo'ps].

Key features of XOOPS

Database-driven

XOOPS uses a relational database (currently MySQL) to store data required for running a web-based content management system.

Fully Modularized

Modules can be installed, uninstalled, and activated or deactivated with a click using the XOOPS module administration system.

Personalization

Registered users can edit their profiles, select site themes, upload custom avatars, and much more!

User Management

XOOPS offers users the ability to search for users by various criteria, and send email or private messages to other users through a template-based messaging system.

Supported World-wide

XOOPS was created and is maintained by a team of several hard-working volunteers from all over the world. The XOOPS Community has more than a dozen official support sites around the world for support of non-English speaking users.

Multi-byte Language Support

XOOPS fully supports multi-byte languages, including Japanese, Simplified and Traditional Chinese, and Korean among others.

Versatile Group Permissions System

XOOPS comes with a powerful and user-friendly permissions system which enables administrators to set permissions by group.

Theme-based skinnable interface

XOOPS is driven by a powerful theme system. Both administrators and users can change the look of the entire web site with just a click of a mouse. There are also over 60 themes available for download!!

XOOPS has received numerous awards:

It was a Finalist for the Best Project of SourceForge 2008 Community Awards and finalist for Best PHP Open Source CMS organized by Packt Publishing. In addition, XOOPS received Awards from "Adobe Edge" as a Top CMS, received 5 out of 5 Stars from Germany's CHIP Magazine, or placed as a top CMS on a list published by ONDD.

XOOPS has over 6,600,000 downloads from the SourceForge Website and is with that the #1 CMS (Content Management System) on SourceForge.

Authors: Michael Beck (Mamba)

Published 2011-04-10. Copyright © 2011 XOOPS Project. Released under [Attribution-ShareAlike 3.0 \(CC BY-SA 3.0\)](https://creativecommons.org/licenses/by-sa/3.0/) license

This document describes recommended structure and GUI for XOOPS Modules

Table of Contents

<u>1. PREAMBLE</u>	<u>4</u>
<u>2. XOOPS 2.5.0 STANDARD MODULE</u>	<u>5</u>
2.1 GUI.....	5
2.2 THE DIRECTORY STRUCTURE.....	8
2.3 PHP/XOOPS VERSION REQUIREMENTS	10
2.4 HELP FILE	11
2.5 DOCUMENTATION	12
2.6 IMAGES.....	12
2.7 ICONS	13
2.8 LANGUAGE FILES	13
<u>3. TUTORIAL.....</u>	<u>14</u>
3.1 COPY FILES	14
3.2 EDIT FILES	15
3.3 TEST THE UPDATED MODULE.....	22
<u>APPENDIX.....</u>	<u>23</u>
3.4 DOCUMENTATION	23
3.5 TOOLS	23

Table of Figures

FIGURE 1: HOME TAB GENERATED BY MODULEADMIN CLASS.....	5
FIGURE 2: ABOUT TAB GENERATED BY MODULEADMIN CLASS	6
FIGURE 3: CONFIGURATIONS CHECK DONE BY MODULEADMIN CLASS.....	10
FIGURE 4: HELP EXAMPLE	11
FIGURE 5: ADMIN GUI BEFORE CONVERSION	14
FIGURE 6: ADMIN GUI AFTER CONVERSION	14

1. PREAMBLE

The TDM Team came up with several new modules (e.g. TDMDownloads), that had a new Admin GUI.

The TDM developers also created **TDMCreate**, a module to create XOOPS modules, providing the same GUI as their own modules. Since their GUI looked very nice, we've taken it as a starting point for our "XOOPS 2.5.x Standard Module GUI" Project, with the goal to bring some GUI consistency among XOOPS modules.

We also talked in the past about releasing updated "**XOOPS Module Packs**" for our users, focusing on some basic functionalities of the Website, e.g. Personal Website, Community Website, etc.

Our goal is to modify existing modules for the "XOOPS Module Packs" so all of them have the same consistent GUI and thus increasing user productivity. Of course, we would need to update the TDMCreate accordingly, so our module developers could create new modules with the same Standard Module GUI.

2. XOOPS 2.5.0 STANDARD MODULE

2.1 GUI

While we were working on implementing the TDM GUI, Mage came up with a ModuleAdmin class that would generate the whole Admin GUI based solely on the Menu definitions defined in /admin/menu.php. The new GUI would look similar to this one:

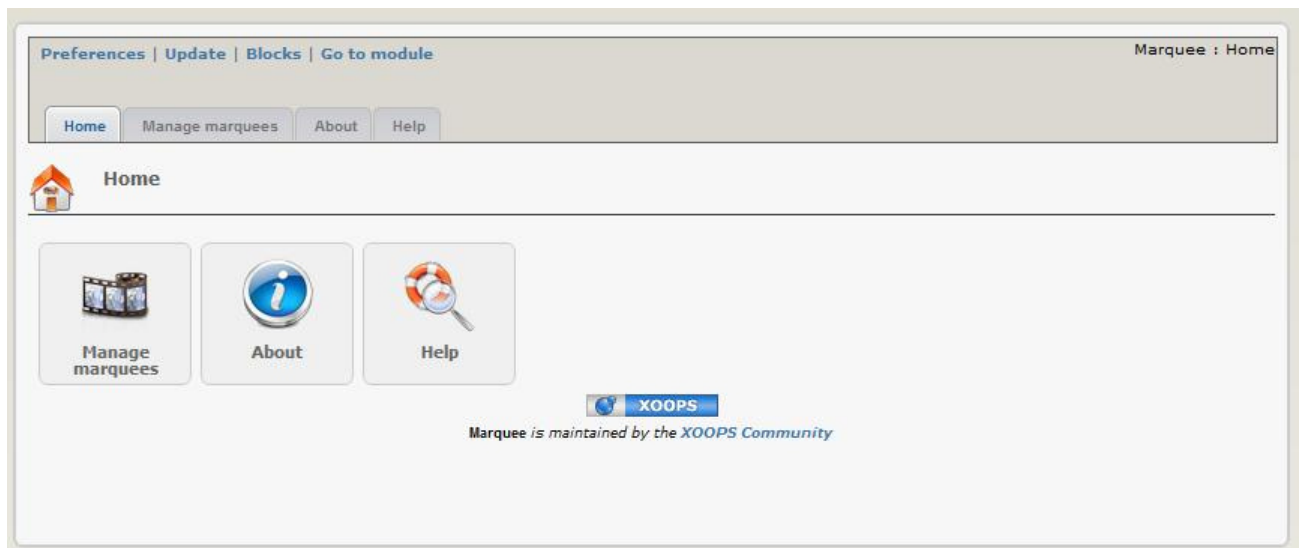


Figure 1: Home Tab generated by ModuleAdmin class

The ModuleAdmin class will also generate the "About" page:

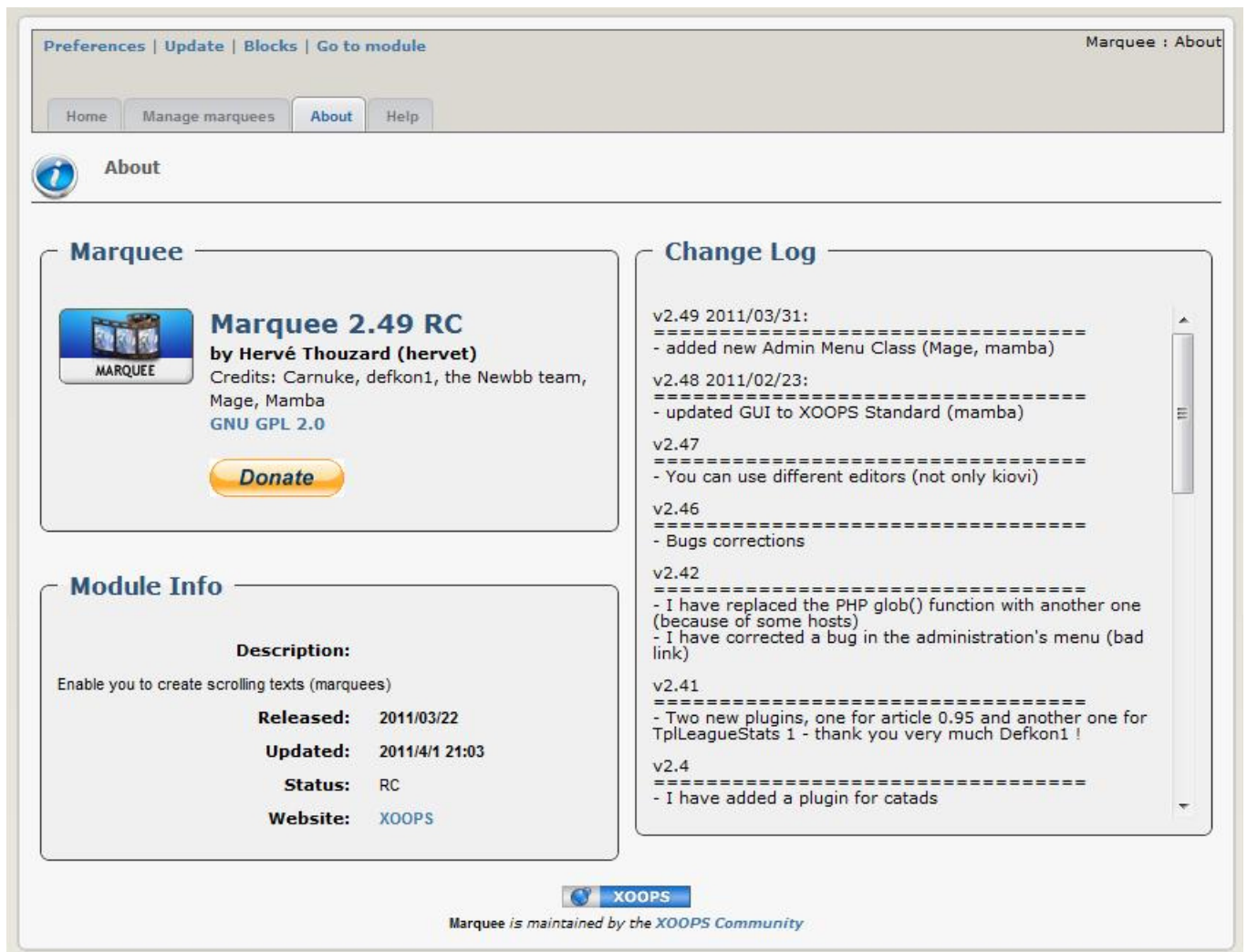
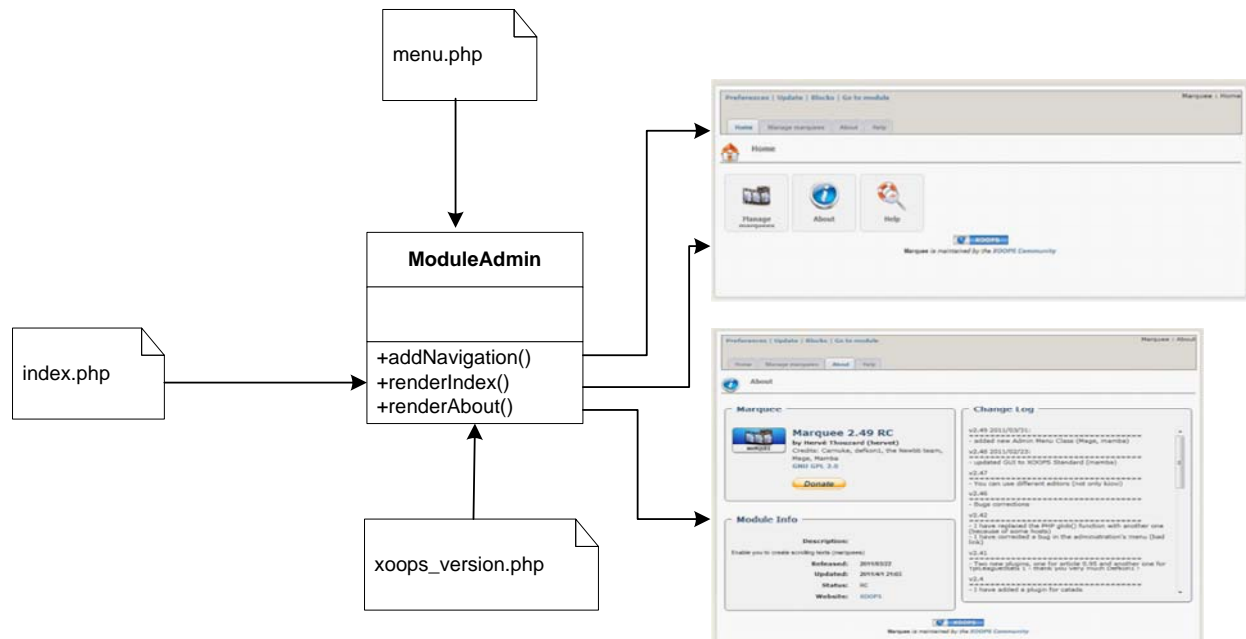


Figure 2: About Tab generated by ModuleAdmin class

The overall interaction diagram would look like this:



The object created from ModuleAdmin class takes:

- info from `/admin/menu.php` and creates the content of Index tab
- info from `xoops_version.php` and creates the content About tab

At this moment we're implementing this design with selected modules that will be part of a "XOOPS Basic Module Pack", and hopefully we'll get feedback from the users, so we could improve it.

The good thing is that since it would be done via the ModuleAdmin class and CSS file, it wouldn't impact the module design itself. In the future, the only thing needed would be to update the ModuleAdmin class to a newer version or a different version with a different GUI, but the module's "model" would stay intact.

Therefore, we also hope that this will help in migration to X3, since we'll achieve a consistency among the modules.

2.2 The Directory structure

We would like to suggest following structure:

```

--- admin
--- blocks
--- class
--- css
--- docs (incl. a link to a manual)
--- images
    |--- admin
    |--- icons
--- include
--- language
    |--- english
        |--- help
--- sql
--- templates
--- test

```

In details, this is the content of the module directories:

Directory	Content
/admin	About.php: informations related to the module Footer.php: contains XOOPS Button and link to XOOPS community Header.php: links to required files Index.php: calls the ModuleAdmin class and sets some variables for it Menu.php: menu items required to generate the Menu Tabs Permissions.php: optional, if we want to set specific permissions
/blocks	All module blocks
/class	Files containing classes specific to the module
/css	Admin.css file is the minimum file required. It is used to style the Admin GUI

/docs	<p>This directory contains:</p> <ul style="list-style-type: none"> a) readme.txt file with basic information, module support site / support email, and bug tracker tool, description of the purpose of the modules and main functionalities, link to a PDF manual/tutorial or Wiki b) install.txt file which indicates how to install this module, e.g. background information, system requirements, incl. Upgrade information c) license.txt GNU License Text d) credits.txt file, listing those that have participated in the module development process e) lang_diff.txt file (if applicable) <p>The best is to put some of them in admin part of module (About tab, Help file)</p>
/images	<p>All images should be in "png" format Images needed for the module, e.g. logo.png</p>
- /admin	Images related to the Admin GUI
- /icons	All icons used in the module
/include	Other files related to this module
/language	Translations in local languages
---- /english	Translation in English. Should be available for each module. Other languages are optional
--- /help	Help.html: file with Help information in local language displayed via XOOPS system. Should be for each local language
/sql	
/templates	Templates for the module
/test	Selenium IDE files for testing the module

2.3 PHP/XOOPS Version requirements

Each module might have specific requirements for PHP and XOOPS version.

As we have stated, this Guide is for modules specifically designated for XOOPS 2.5.0, which requires PHP 5.2

In the `xoops_version.php` file you can specify your minimum requirements:

```
$modversion['min_php']=5.2;  
$modversion['min_xoops']="XOOPS 2.5";
```

By calling the configuration check in the `/admin/index.php`:

```
$index_admin->addConfigLabel(_AM_MYLINKS_CONFIG_CHECK) ;  
$index_admin->addLineConfigLabel(_AM_MYLINKS_CONFIG_PHP, $xoopsModule->getInfo("min_php"), 'php') ;  
$index_admin->addLineConfigLabel(_AM_MYLINKS_CONFIG_XOOPS, $xoopsModule->getInfo("min_xoops"), 'xoops') ;
```

you can see the result on the screen:

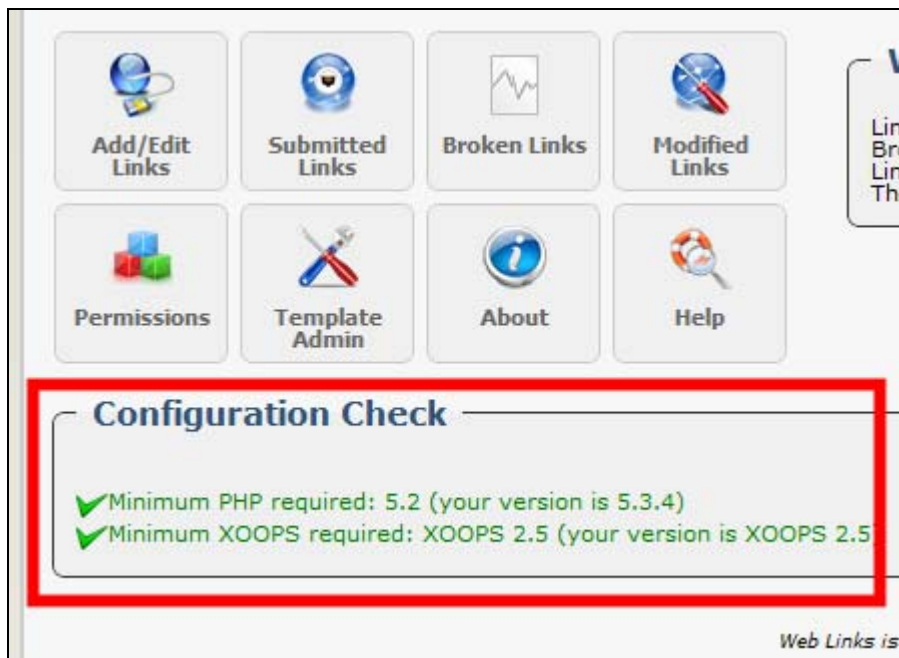


Figure 3: Configurations check done by ModuleAdmin class

2.4 Help file

The help file is an HTML file that follows the standard set by XOOPS 2.5.0

```
<div id="help-template" class="outer">
  <h1 class="head">Help:
    <a class="ui-corner-all tooltip" href="<{$xoops_url}>/modules/mymodule/admin/index.php"
      title="Back to the administration of MyModule"> MyLinks 

    </a></h1>
  <!-- -----Help Content ----- -->
  <h4 class="odd">Description</h4>
  <p class="even">
    This is description of your module.</p>
  <h4 class="odd">Install/uninstall</h4>
  <p class="even">
    Installation information for your module</p>
  <h4 class="odd">Tutorial</h4>
  <p class="even">
    You can find a more detailed Tutorial <a href="http://xxx.com ">here</a>
  <!-- -----Help Content ----- -->
</div>
```

The text in **red** indicates what should be customized for each individual module and language.

You'll need to set the name is xoops_version.php: `$modversion['help'] = 'page=help';`

XOOPS will then look for "help.html" in the /language/local_language/help/ directory, and it will display it as in this example taken from MyLinks:

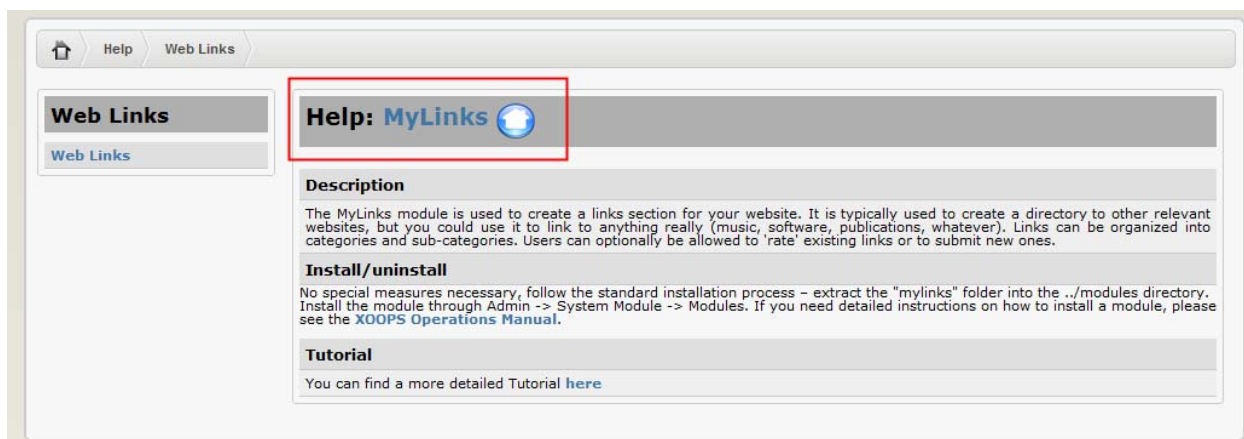


Figure 4: Help example

You can click on the "MyLinks" link and you'll get back to your module's Admin page

2.5 Documentation

There are several documents that are recommended for a module:

- a) **readme.txt** file with basic information, module support site / support email, and bug tracker tool, description of the purpose of the modules and main functionalities, link to a PDF manual/tutorial or Wiki
- b) **install.txt** file which indicates how to install this module, e.g. background information, system requirements, incl. Upgrade information
- c) **license.txt** GNU License Text
- d) **credits.txt** file, listing those that have participated in the module development process
- e) **lang_diff.txt** file (if applicable)

The best is to put some of them in admin part of module (About tab, Help file)

Each module should also have a Manual/Tutorial, either as a PDF document or XOOPS Wiki entry

2.6 Images

As part of the “Blue Move“, all module logos are based on the Blue theme:



Which then look like these logos:



Another standard image used is the XOOPS Microbutton:



2.7 Icons

In order to make the GUI consistent across all modules, we need to use the same icons for the same standard actions, e.g. Edit, Delete, Add, On, Off, etc. This way users, when they start using a new module, don't have to guess what a particular icon means.

Freedesktop did a very good job on categorizing icons:







<http://standards.freedesktop.org/icon-naming-spec/icon-naming-spec-latest.html>

and we'll be using it as a starting point.

In the past XOOPS Community decided to use the free Crystal icons:

<http://www.everaldo.com/crystal/?action=preview>

Over the next few weeks we'll be working on selecting and cataloguing the XOOPS Basic Icon Set. When done, it should look similar to the list below:

	Category	Name	Description	Icon name
	actions	go-down	The icon for the go down in a list action.	down.png
	actions	edit	The icon for the edit action.	edit.png
	actions	edit-copy	The icon for the copy action.	editcopy.png
	actions	edit-cut	The icon for the cut action.	editcut.png
	actions	edit-delete.	The icon for the delete action.	editdelete.png
	actions	edit-paste	The icon for the paste action.	editpaste.png

The standard size will be 16x16 or 22x22 – it will be decided by the community.

The goal is to have this “XOOPS Basic Icon Set” as part of XOOPS 2.5.2, placed in /uploads/icons, so all XOOPS modules could call the icons from there.

2.8 Language Files

Take a look at the two submitted files in /language/english directory: admin.php and modinfo.php.

Merge the content from it into the modules' respective admin.php and modinfo.php files, if the entries don't exist already there.

If the module doesn't have the file, just copy it over, including the whole /language/english/help directory. Please update the help.php file with information related to the module.

3. TUTORIAL

In this tutorial we'll show how to convert an existing module to the new Admin GUI. We'll take the xDirectory 1.5 module with this GUI:

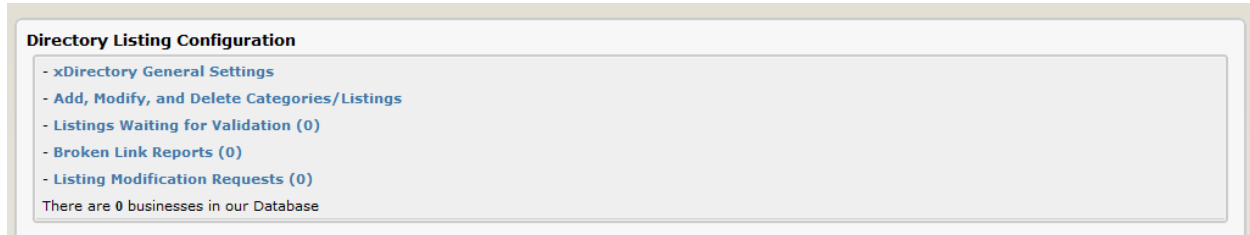


Figure 5: Admin GUI before conversion

and we'll convert it into this:

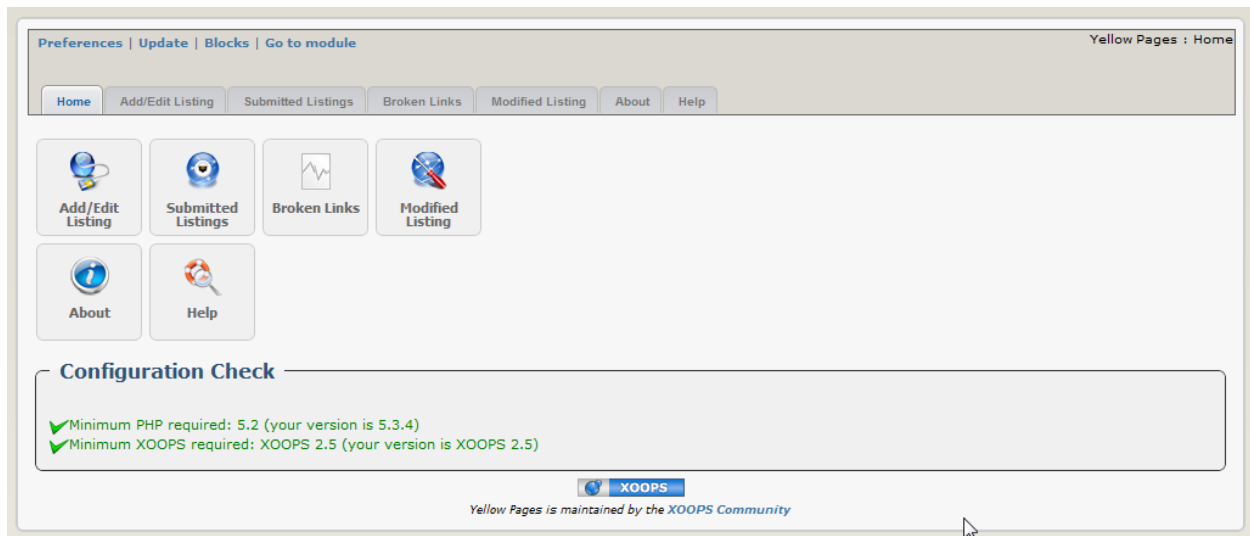


Figure 6: Admin GUI after conversion

3.1 Copy files

Take the provided files from the provided "XOOPS 2.5.0 GUI Conversion Patch" and copy them to the existing module following these steps:

1. Copy the following files/directories:
 - a. about.php (in /admin)
 - b. footer1.php (in /admin)
 - c. header1.php (in /admin)
 - d. the whole /class directory
 - e. the whole /css directory

- f. the whole /docs directory
- g. the whole /images/admin directory
- h. changelog.txt (in /language/english)
- i. the whole directory /language/english/help directory

3.2 Edit files

/admin

Index.php

- rename the /admin/index.php to main.php
- copy from your Conversion files /admin/index.php to /admin

menu.php

- open /admin/menu.php and do the same: rename all occurrences of "index.php" to "main.php",
- while in menu.php, add two extra menus:
 - Home (on top)

```
$adminmenu[$i]['title'] = _MI_XDIRECTORY_ADMIN_HOME ;
$adminmenu[$i]['link'] = 'admin/index.php' ;
$adminmenu[$i]['desc'] = _MI_XDIRECTORY_ADMIN_HOME_DESC ;
$adminmenu[$i]['icon'] = 'images/admin/home.png' ;
```

- About (at the bottom)

```
$adminmenu[$i]['title'] = _MI_XDIRECTORY_ADMIN_ABOUT;
$adminmenu[$i]['link'] = 'admin/about.php';
$adminmenu[$i]['desc'] = _MI_XDIRECTORY_ADMIN_ABOUT_DESC;
$adminmenu[$i]['icon'] = 'images/admin/about.png';
```

- While you're there, add for each of the original menu items:

```
$adminmenu[$i]['icon'] = 'images/admin/about.png';
```

Since there is no icon associated with each menu item.

- change the counting from absolute to relative, i.e. from:

```
$adminmenu[1]['title'] = _MI_XDIR_ADMENU2;
$adminmenu[1]['link'] = "admin/index.php?op=linksConfigMenu";
$adminmenu[2]['title'] = _MI_XDIR_ADMENU3;
$adminmenu[2]['link'] = "admin/index.php?op=listNewLinks";
```

To this:

```
$i = 1;
$adminmenu[$i]['title'] = _MYLINKS_ADMIN_HOME ;
$adminmenu[$i]['link'] = 'admin/index.php' ;
$adminmenu[$i]['desc'] = _MYLINKS_ADMIN_HOME_DESC ;
$adminmenu[$i]['icon'] = 'images/admin/home.png' ;
$i++;
$adminmenu[$i]['title'] = _MI_XDIR_ADMENU2;
$adminmenu[$i]['link'] = "admin/main.php?op=linksConfigMenu";
$adminmenu[$i]['desc'] = _MYLINKS_ADMIN_HOME_DESC ;
$adminmenu[$i]['icon'] = 'images/admin/home.png' ;
```

- Change from `_MI_MYMODULE_` to `_MI_XDIR_` in the variables.
- Select icons from the Crystal icons set (or other similar set) that you want for each menu and copy them to the `/images/admin` directory. The size of the images should be 32x32
- Now add for each old menu entry the line with image and description, and change the names accordingly:

```
$adminmenu[$i]['desc'] = _MYMODULE_ADMIN_HOME_DESC ;
$adminmenu[$i]['icon'] = 'images/admin/home.png' ;
```

- When done, this how it should look like this:

```
$i = 1;
$adminmenu[$i]['title'] = _MI_XDIR_ADMIN_HOME ;
$adminmenu[$i]['link'] = 'admin/index.php' ;
$adminmenu[$i]['desc'] = _MI_XDIR_ADMIN_HOME_DESC ;
$adminmenu[$i]['icon'] = 'images/admin/home.png' ;
$i++;
$adminmenu[$i]['title'] = _MI_XDIR_ADMENU2;
$adminmenu[$i]['link'] = "admin/main.php?op=linksConfigMenu";
$adminmenu[$i]['desc'] = _MI_XDIR_ADMENU2_DESC ;
$adminmenu[$i]['icon'] = 'images/admin/addlink.png' ;
$i++;
$adminmenu[$i]['title'] = _MI_XDIR_ADMENU3;
$adminmenu[$i]['link'] = "admin/main.php?op=listNewLinks";
$adminmenu[$i]['desc'] = _MI_XDIR_ADMENU3_DESC ;
$adminmenu[$i]['icon'] = 'images/admin/submittedlink.png' ;
$i++;
$adminmenu[$i]['title'] = _MI_XDIR_ADMENU4;
$adminmenu[$i]['link'] = "admin/main.php?op=listBrokenLinks";
$adminmenu[$i]['desc'] = _MI_XDIR_ADMENU4_DESC ;
$adminmenu[$i]['icon'] = 'images/admin/brokenlink.png' ;
```



```

$i++;
$adminmenu[$i]['title'] = _MI_XDIR_ADMENU5;
$adminmenu[$i]['link'] = "admin/main.php?op=listModReq";
$adminmenu[$i]['desc'] = _MI_XDIR_ADMENU5_DESC ;
$adminmenu[$i]['icon'] = 'images/admin/modifiedlink.png' ;
$i++;
$adminmenu[$i]['title'] = _MI_XDIR_ADMIN_ABOUT;
$adminmenu[$i]['link'] = 'admin/about.php';
$adminmenu[$i]['desc'] = _MI_XDIR_ADMIN_ABOUT_DESC;
$adminmenu[$i]['icon'] = 'images/admin/about.png';

```

header.php / footer.php

- If there are no header.php / footer.php files in the /admin directory, rename the header1.php / footer1.php files that you copied earlier to header.php / footer.php
- If there are already header.php / footer.php files, then merged the content from header1.php / footer1.php into header.php / footer.php and delete header1.php / footer1.php

xoops_version.php

Add info needed for XOOPS to create the menu tabs:

```
$modversion['system_menu'] = 1;
```

Replace:

```

$modversion['help'] = "xdir.html";
$modversion['license'] = "GPL see LICENSE";

```

With

```

$modversion['help']      = 'page=help';
$modversion['license']   = 'GNU GPL 2.0';
$modversion['license_url'] = "www.gnu.org/licenses/gpl-2.0.html/";

```

Change version to 1.60

```
$modversion['version'] = 1.60;
```

Change Official Version to 0, since it is not an official XOOPS module:

```
$modversion['official'] = 0;
```

Add:

```
$modversion["author"] = Adam Frick;
```

```
//about
$modversion['status_version'] = 'RC';
$modversion['release_date'] = '2011/04/02';
$modversion["module_website_url"] = "http://www.xoops.org/";
$modversion["module_website_name"] = "XOOPS";
$modversion["module_status"] = "RC";
$modversion["author_website_url"] = "http://www.FolsomLiving.com/";
$modversion["author_website_name"] = "Adam Frick";
$modversion['min_php']=5.2;
$modversion['min_xoops']="XOOPS 2.5";
```

/admin/about.php

Replace all `_AM_MYMODULE_` with `_AM_XDIR_`

/admin/index.php

Replace all `_AM_MYMODULE_` with `_AM_XDIR_`

/language/english/modinfo.php

Add there:

```
//1.60
// index.php

define('_MI_XDIR_ADMIN_HOME',"Home");
define("_MI_XDIR_ADMIN_HOME_DESC","Back to Home");
define("_MI_XDIR_ADMIN_ABOUT" , "About");
define("_MI_XDIR_ADMIN_ABOUT_DESC" , "About this module");

define _MI_XDIR__ADMENU2_DESC" , "Add/Edit Links");
define _MI_XDIR__ADMENU3_DESC" , "List New Links");
define _MI_XDIR__ADMENU4_DESC" , "List Broken Links");
define _MI_XDIR__ADMENU5_DESC" , "List Requested Modifications");
```

If not existing, create:

/language/english/admin.php

Add there:

```
//1.60
```

```
// About.php
define("_AM_XDIR_ABOUT_RELEASEDATE",      "Released: ");
define("_AM_XDIR_ABOUT_UPDATEDATE",      "Updated: ");
define("_AM_XDIR_ABOUT_AUTHOR",          "Author: ");
define("_AM_XDIR_ABOUT_CREDITS",         "Credits: ");
define("_AM_XDIR_ABOUT_LICENSE",         "License: ");
define("_AM_XDIR_ABOUT_MODULE_STATUS",    "Status: ");
define("_AM_XDIR_ABOUT_WEBSITE",         "Website: ");
define("_AM_XDIR_ABOUT_AUTHOR_NAME",     "Author name: ");
define("_AM_XDIR_ABOUT_CHANGELOG",       "Change Log");
define("_AM_XDIR_ABOUT_MODULE_INFO",     "Module Info");
define("_AM_XDIR_ABOUT_AUTHOR_INFO",     "Author Info");
define("_AM_XDIR_ABOUT_DESCRIPTION",     "Description: ");

// Configuration
define("_AM_XDIR_CONFIG_CHECK","Configuration Check");
define("_AM_XDIR_CONFIG_PHP","Minimum PHP required: %s (your version is %s)");
define("_AM_XDIR_CONFIG_XOOPS","Minimum XOOPS required: %s (your version is %s)");
```

Modify module logo to the Blue Move logo style



main.php

- open main.php and rename all occurrences of “index.php” to “main.php”, except the “redirect_header” entries like this:

```
redirect_header("index.php",1,_MD_VOTEDELETED);
```

They need to stay the same, i.e. don't replace the index.php here.

The best is if you do it one by one to make sure that you're changing the correct ones.

As next, let's add XOOPS microbutton and link to each page, by adding a footer to the other tabs (it is already in Home and About tabs). To do so, we need to add:

```
include "footer.php";
```

before each

```
xoops_cp_footer();
```

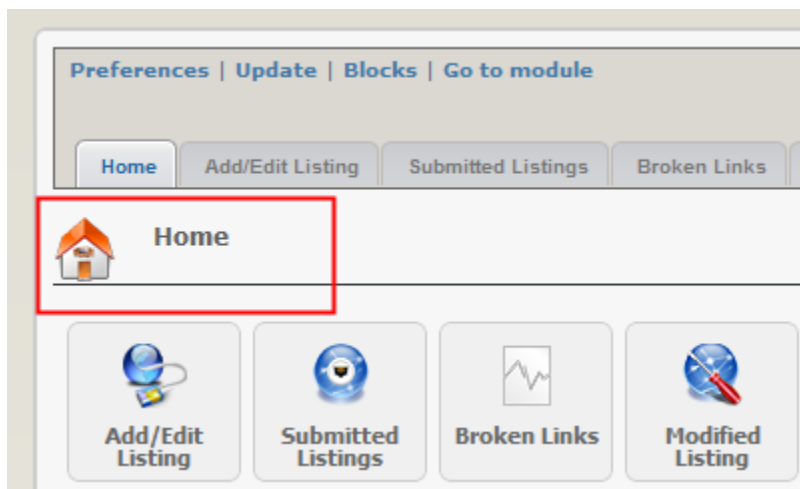
Note: In some situations, we might get an error:

Notice: Undefined variable: xoopsModule in file /modules/xdirectory/admin/footer.php line 22

Make sure that you add **\$xoopsModule** to the list of defined global variables for that function, for example, in function listNewLinks() in line 90:

```
global $xoopsDB, $xoopsConfig, $mysys, $eh, $mytree, $xoopsModule;
```

As last thing, we'll add navigation icons for each page. Right now they are only in the "Home" and "About" tabs:



To do that, you need to add on top of your /admin/main.php file:

```
include_once XOOPS_ROOT_PATH."/modules/" . $xoopsModule->getVar("dirname") .  
"/class/admin.php" ;
```

And then, after each:

```
xoops_cp_header() ;
```

that is part of a function being called by a click on the tab, add following two lines:

```
$index_admin = new ModuleAdmin() ;  
echo $index_admin->addNavigation('main.php?op=listNewLinks') ;
```

where the text in **red** is the name of the link that is being called by the particular Tab. For example, the link for a tab showing "Submitted links", if you look in menu.php is:

```
$adminmenu[$i]['link'] = "admin/main.php?op=listNewLinks";
```

Therefore the entry for navigation will be:

```
echo $index_admin->addNavigation('main.php?op=listNewLinks');
```

We can also comment out the heading for that page that we don't need anymore:

```
// echo "<h4>"._MD_WEBLINKSCONF."</h4>";
```

After we have made these changes, we can see the result:



The last thing that we need to do is to update the changelog.txt

3.3 Test the updated module

The best way to test module functionality is to create “use cases” for the module – it should be done by the original module developer, if he is still available, because s/he knows the best the module purpose.

Since this is a repetitive and boring process, it makes sense to automate it with tools such as [Selenium IDE](#) – an extension for Firefox. You can record all the use cases and then run them after each code changes to make sure that these changes didn't breake any functionality.

As we work to improve the quality of XOOPS modules, our goal is to have Selenium IDE test available for all modules that are officially maintained by the XOOPS Community.

APPENDIX

There are many very useful documents and tools for XOOPS module developers:

3.4 Documentation

1. [QA Tool for Modules](#)
2. [XOOPS Coding Standards](#)
3. [Module Development Guide](#) (old)
4. [XOOPS Wiki](#)

3.5 Tools

1. [TDMCreate](#) – XOOPS module generator
2. [PHP Code Sniffer](#)
3. [Selenium IDE](#)
4. [Firebug/FirePHP](#)

Over next few months we'll work on consolidating and updating these documents, and on cleaning and updating the Wiki.

Your help is needed and appreciated!