Structure

Directory Structure & Files

The directory structure used allows you to separate different MVC applications into self-contained units. This helps keep related code organized, easy to find, and can make redistribution as packages considerably easier. To illustrate the typical module directory structure and files:

```
/app
.. /modules
.. .. /mod_{ModuleName}
.. .. .. /tmpl
.. .. .. .. default.php
.. .. .. helper.php
.. .. .. mod_{ModuleName}.php
.. .. .. mod_{ModuleName}.xml
```

A module is in its most basic form two files: an XML configuration file and a PHP controller file. Typically, however, a module will also include a view file which contains the HTML and presentation aspects.

/tmpl
: This directory contains template files.
  default.php
  : This is the module template. This file will take the data collected by mod_{ModuleName}.php and generate the HTML to be displayed on the page.

helper.php
: This file contains the helper class which is used to do the actual work in retrieving the information to be displayed in the module (usually from the database or some other source).

mod_{ModuleName}.php
: This file is the main entry point for the module. It will perform any necessary initialization routines, call helper routines to collect any necessary data, and include the template which will display the module output.

mod_{ModuleName}.xml
: The XML configuration file contains general information about the module (as will be displayed in the Module Manager in the administration interface), as well as module parameters which may be supplied to fine tune the appearance / functionality of the module.

While there is no restriction on the name itself, all modules must be prefixed with "mod_".

Implementation

Most modules will perform three tasks in the following order:

- Define the module namespace
- Include the helper.php file which contains the class to be used to collect any necessary data and render it
- Instantiate the helper class and call the display() method which will:
    - Invoke the appropriate helper class method to retrieve any data that needs to be available to the view
    - Include the template to display the output
- Include the template to display the output

Here are the contents of mod_listnames.php:

```
<?php
// Define the namespace
namespace ModulesListNames;

// Include the helper file
require_once __DIR__ . DS . 'helper.php';

// Instantiate the module helper and call its display() method
with(new Helper($params, $module))->display();
```