

Documentación General

Nombre del proyecto

IFC CONTROLLER ACTION CONFIGURATION & MODULES

Resumen

Recursos [Zend_Application_Resource](#), sistema de parametrización de [Zend_Controller_Action](#) a través de ficheros de configuración y módulos para [Zend Framework](#) (versión 1.11).

Descripción

El proyecto incluye tres partes diferenciadas pero todas relacionados con Zend Framework, que son las siguientes:

1. [Zend_Application_Resource](#).

El proyecto contiene algunas clases que heredan de la clase abstracta [Zend_Application_Resource_ResourceAbstract](#) proporcionada por el *framework* y que se utiliza para poder establecer parametrizaciones del *framework* a través del fichero de configuración (.ini, .xml, ...) que se carga en el *bootstrap* de la aplicación (se puede obtener una referencia a este concepto en la documentación de ZF, concretamente en <http://framework.zend.com/manual/en/zend.application.theory-of-operation.html>).

Con estas clases he permitido parametrizar mediante esta funcionalidad ofrecida por ZF, algunos aspectos que no estaban implementados en la versión 1.11, que era la versión estable que había en el momento que empecé el desarrollo de este proyecto.

2. Sistema de parametrización de [Zend_Controller_Action](#).

Esta es la parte principal de éste proyecto. En esta parte he desarrollado un sistema que permite parametrizar las acciones ([Zend_Controller_Action](#)) a través de valores definidos en alguna base de datos (ficheros planos, base de datos relacionales, etc.); el proyecto también tiene una clase, `ifc\zend\resource\ModuleConfigurator`, que hereda de [Zend_Application_Resource_ResourceAbstract](#) que permite parametrizar este sistema a través del fichero de configuración de la aplicación ZF ([application.ini](#)) permitiendo definir las implementaciones de los plugins que ejecutan las configuraciones.

A continuación se detalla las distintas partes que componen este sistema:

1. El sistema está preparado para poder obtener los valores de los parámetros de cualquier tipo base de datos, no obstante el proyecto solo tiene implementado una clase que permite obtener los parámetros de un fichero de configuración, del tipo .ini, almacenado en un directorio que puede ser localizado desde el directorio raíz del módulo al cual pertenece la acción que se está ejecutando en cada llamada; esta clase es `ifc\zend\controller\action\helper\configurator\Standar`
2. Posteriormente a la carga de los valores de los parámetros de cada acción el sistema ejecuta, a través de un configurador, la ejecución de todas las configuraciones correspondientes a la acción; el configurador, tiene que ser una clase que implemente la interfaz `ifc\zend\controller\action\configurator\ConfiguratorInterface`. El configurador es el que gestiona y coordina la aplicación de las parametrizaciones asociadas a la acción que se ejecuta en cada llamada.

El proyecto contiene la implementación de una clase abstracta,

ifc\zend\controller\action\configurator\ConfiguratorAbstract, que contiene ciertas funciones que se han considerado que pueden ser globales a los distintos configuradores que se pueden necesitar implementar, y también la implementación de un configurador, *ifc\zend\controller\action\configurator\Basic*, que permite se definan configuraciones a nivel de módulo (se aplican para todos los controladores y acciones de cada controlador), a nivel de controlador (se aplican a todas las acciones del controlador indicado) y a nivel de acción (se aplican solo a la acción indicada); no obstante el sistema permite redefinir configuraciones por si se necesita ejecutar una configuración en varios controladores y/o acciones pero en algún controlador y/o acción es necesario cambiar el valor de los parámetros que se aplican generalmente.

3. Las configuraciones que gestiona y coordina el configurador, son clases que tienen que implementar la interfaz *ifc\zend\controller\action\configuration\ConfigurationInterface*, o también pueden heredar de la clase abstracta *ifc\zend\controller\action\configuration\ConfigurationAbstract*, que proporciona el proyecto y donde se han implementado ciertas funcionalidades que se ha considerado que pueden ser comunes en la mayoría de las configuraciones.

Las configuraciones, son las que almacenan los valores de los parámetros, obtenidos de la base de datos, y las que ejecutan las operaciones de parametrización, que se requieren en la acción o por alguno de los componentes que ésta utiliza.

El proyecto proporciona configuraciones para parametrizar ciertas acciones requeridas por las acciones y para ciertos componentes; todas ellas se encuentra definidas dentro del espacio de nombres *ifc\zend\controller\action\configuration*.

3. Módulos para aplicaciones.

El proyecto también contiene la implementación de dos módulos que pueden ser utilizados en multitud en varias aplicaciones web debido a la funcionalidad que implementan; no obstante los módulos se encuentran en una fase alpha por lo que no ofrecen todas las funcionalidades que deberían ofrecer, además de necesitar ser más ampliamente testeados.

1. Módulo de autenticación (acceso al sistema).

El módulo ofrece un sistema de autenticación basado en objetos de clases mapeados sobre una base de datos a través del ORM (*Object Relational Mapper*) [Doctrine](#) (versión 2.0.1), permitiendo utilizar cualquier sistema gestor de base de datos soportado por [Doctrine](#).

El módulo tiene implementado las funcionales de inicio de sesión (*Log in*) y cierre de sesión (*Log out*).

2. Módulo de control de excepciones.

El módulo permite capturar las excepciones para posteriormente registrar las en los registros (*logs*) del sistema, configurados en el *bootstrap* de la aplicación Zend, y posteriormente enviar una respuesta con el código HTTP de estado correspondiente y la página asociada al código de error con la información del error en el caso que se haya activado o no la opción de mostrar dicha información.