

 marketful <small>fulfillment</small>	Documentación del archivo: class-loader.php
	Autor: Mauricio Alcala
	Fecha: 13 de agosto de 2018
	Versión: 1.01
	Autoriza: Adolfo Yanes

Descripción general:

La clase loader se encarga de cargar con sus métodos los filtros y las acciones.

Funciones PHP

```

/*****
 * @Clase MKF_Loader
 *
 * @atributos privados ($actions, $filters)
 *
 */
class MKF_Loader {

    protected $actions; // The array of actions registered with WordPress.
    protected $filters; // he array of filters registered with WordPress.
/*****

    // Initialize the collections used to maintain the actions and filters.
    /*****
    * @función __construct()
    * Inicializa la colección usada para mantener las acciones de
    * los filtros.
    */
    public function __construct()
    {

        $this->actions = array();
        $this->filters = array();

    }
/*****
// Add a new action to the collection to be registered with WordPress.
/**
 * @función add_action(@string (hook), @string, @string, @integer = (10), @integer = (1))
 * Asigna al atributo actions el valor de add con los parametros
 * recibidos.
 */
    public function add_action( $hook, $component, $callback, $priority = 10, $accepted_args = 1 )
    {
        $this->actions = $this->add( $this->actions, $hook, $component, $callback, $priority,
$accepted_args );
    }
/*****

// Add a new filter to the collection to be registered with WordPress.
/*****
 * @función add_filter(@string (hook), @string, @string, @integer = (10), @integer = (1))
 * Agrega un nuevo filtro a la coleccion para ser registrada con
 * WordPress
 */
    public function add_filter( $hook, $component, $callback, $priority = 10, $accepted_args = 1 ) {
        $this->filters = $this->add( $this->filters, $hook, $component, $callback, $priority,
$accepted_args );
    }
/*****

```

```

// A utility function that is used to register the actions and hooks into a single collection.
/*****
 * @función add (@string, @string, @string, @string, @string, @integer, @integer )
 *
 * Una función utilitaria que es usada para registrar las acciones
 * y los hooks en una sola colección.
 */
private function add( $hooks, $hook, $component, $callback, $priority, $accepted_args ) {

    $hooks[] = array(
        'hook'          => $hook,
        'component'     => $component,
        'callback'      => $callback,
        'priority'      => $priority,
        'accepted_args' => $accepted_args
    );

    return $hooks;

}
/*****/

// Register the filters and actions with WordPress
/**
 * @función run ()
 *
 * Registra los filtros y acciones con Woerdpres, utilizando un
 * foreach para recorrer todos loos posibles
 * filtros y acciones.
 */
public function run() {

    foreach ( $this->filters as $hook ) {
        add_filter( $hook['hook'], array( $hook['component'], $hook['callback'] ),
$hook['priority'], $hook['accepted_args'] );
    }

    foreach ( $this->actions as $hook ) {
        add_action( $hook['hook'], array( $hook['component'], $hook['callback'] ),
$hook['priority'], $hook['accepted_args'] );
    }

}
/*****/

```