marketful fulfillment	Documentación del archivo: admin-product-edit-form.php
	Autor: Mauricio Alcala
	Fecha: 8 de agosto de 2018
	Versión: 1.02
	Autoriza: Adolfo Yanes

Descripción general:

Archivo PHP que se manda a llamar desde el plugin de Marketful, en el cual se pueden modificar la información de venta de un producto.

Funciones PHP

```
/*********************************************************************************
* @Script
* A la variable $product_id se le asigna lo resultante de usar la @funcion isset
* que verifica que se hay posteado alguna variable llamada 'product_id' tiene
* algún dato en caso contrario $product_id tomara el valor ""
*/
$product_id = isset($_REQUEST['product_id']) ? $_REQUEST['product_id'] : "";

*********************************************************************************/


/*********************************************************************************
* @Script
* El siguiente IF valida que la variable $product_id NO este vacía en cuyo caso
* primeramente creara un @objeto llamado $productObject que tomara todos los valores que devuelve el
* @método MKF_ProductEntry.
* Después con las @funciones get_post() y get_post_meta() se obtienen los registros relacionados con
* ese id.
* Después se asignan los valores dentro del array $meta[0] para mostrarlos posteriormente.
*
* En caso de que la validación no se haya cumplido, se valida si $ml_categories NO esta vacía en
* cuyo caso con la @función Json_decode entrega el resultado de @string codificado para utilizarlo
* dentro del código PHP
*/
if (!empty($product_id))
{
$productObject = MKF_ProductEntry::GetInstance();
$custom_fields = get_post($product_id); $meta = get_post_meta($product_id);
$post_title = $custom_fields->post_title;
$post_sku = $meta['_sku'][0];
$all_mlmeta = $productObject->get_ml_metadata($product_id)[0]['data'];
$ml_title = $all_mlmeta[0]->title;
$ml_store = $all_mlmeta[0]->store_recall;
$ml_stock = $all_mlmeta[0]->inventary;
$ml_sts = $all_mlmeta[0]->status;
$ml_exp = $all_mlmeta[0]->exposicion;
$ml_wtime = $all_mlmeta[0]->time_warranty;
$ml_categories = $all_mlmeta[0]->categories;
$categories = $productObject->get_ml_categories();
if (!empty($ml_categories))
{
$ml_obj_cat = json_decode($ml_categories, true);
}
}
*********************************************************************************/
```

```
/*********************************************************************************
* @Script
*
* Creamos un foreach para recorrer todo el resultado de la query que retorna la @función de Json
* Se hace un echo al id de categoría para asignarlo como valor de la opción.
* Se hace un echo al nombre de la categoría para mostrarlo
*/
<?php
foreach (json_decode($categories->getBody(), true) as $key =>
$category) : ?>
<?php echo $category['id']; ?>
<?php echo $category['name'];
endforeach;
?>
*********************************************************************************/
```

Funciones JQuery / Ajax

```
/*********************************************************************************
* - @Función  JQuery/Ajax: getCategory(@string, @string, @currentlevel)
* Se valida que tree tenga valor de 'father' Y el valor del tag con id category_aux no sea
* igual al valor de categ.value, en el tag id category_aux seteamos el valor de categ.value,
* después retiramos el tag class subcat.
*
* En caso contrario, buscamos los tags con class syi-category-tree__column. Validamos que index sea
* mayor que el nivel actual En tal caso removemos el ítem A numItems le asignamos el valor de los
* items menos uno veamos una @función Ajax que sea del tipo GET con la url con terminación en
* categ.value.
*
* Se crea una @función que recibe el @parámetro data, después Se crea una @función que recibe el
* @parámetro data creamos una variable que contiene varios div para mostrar las categorías. Se
* buscan todos los tags hijos de categorías y se les aplica la siguiente @función se muestra al
* final del tag class syi-category-tree__column: el contenido de Ndiv en caso contrario solo se
* muestra un option con un objeto Se agrega el valor de Ndiv al final del tag de la class yi-
* categorytree__column se ajusta la vista de #demo con scrollleft Se agrega un boton para enviar
*/
function getCategory(categ, tree, currentLevel)
{
if (tree == "father" && jQuery("#category_aux").val() != categ.value)
{
jQuery("#category_aux").val(categ.value);
jQuery(".subcat").remove();
}
else
{
jQuery.each(jQuery('.syi-category-tree__column.subcat'), function(index, item)
{
if (parseInt(index) >= parseInt(currentLevel))
{
jQuery(item).remove();
}
});
}
var numItems = jQuery('.syi-category-tree__column').length-1;
jQuery.ajax(
{
type: "GET",
url: "https://api.mercadolibre.com/categories/"+categ.value,
success: function(data)
{
if (data.children_categories.length>0)
{
```

```
Ndiv = '<div class="ui-box syi-category-tree__column cat subcat">';
Ndiv += '<div data-index="1" class="syi-category-tree__container ">';
Ndiv += '<select class="syi-category-tree__selector selected"
id="level_'+numItems+'" title="Elige una categoría" size="20" onChange="getCategory(this,
\'child\', ' + numItems +');" name="ml_categories[child][]">';
jQuery.each(data.children_categories,function(i,obj)
{
/* Se agregan ma div */
Ndiv +='<option class="syi-category-tree__option"
value='+obj.id+'>'+obj.name+'</option>';
});
Ndiv += '</select>';
Ndiv += '</div>';
Ndiv += '</div>';
jQuery(".syi-category-tree__column:last").after(Ndiv);
}
else
{
Ndiv +='<option class="syi-category-tree__option"
value='+obj.id+'>'+obj.name+'</option>';
}
Ndiv += '</select>';
Ndiv += '</div>';
Ndiv += '</div>';
jQuery(".syi-category-tree__column:last").after(Ndiv);
jQuery('#demo').scrollLeft(2000);
add_submit_button();
}
});
} **********************************************************************************************/


/**********************************************************************************
* @Función add_submit_button() JQuery
* Se crea una @funcion llamada add_submit_button() que agrega
* los valores de nuevas categorias aen el ultimo elemento de
* .syi-category-tree__column:last con Jquery
* Se agrega e Ndiv al final de la clase syi-catogory-tree__column
**/
function add_submit_button()
{
var Ndiv = '<div class="ui-box syi-category-tree__column subcat">';
Ndiv += '<div class="syi-category-tree__box-msg ">';
Ndiv += '<svg class="ui-icon " xmlns="http://www.w3.org/2000/svg" width="70" height="70"
viewBox="0 0 52 52">';
Ndiv += '<path fill="#468847" d="M42.9910714,20.2901786 C42.9910714,19.6651754
42.7901806,19.1517877 42.3883929,18.75 L39.3415179,15.7366071 C38.9174086,15.3124979
38.4151815,15.1004464 37.8348214,15.1004464 C37.2544614,15.1004464 36.7522343,15.3124979
36.328125,15.7366071 L22.6674107,29.3638393 L15.1004464,21.796875 C14.6763372,21.3727657
14.17411,21.1607143 13.59375,21.1607143 C13.01339,21.1607143 12.5111628,21.3727657
12.0870536,21.796875 L9.04017857,24.8102679 C8.63839085,25.2120556 8.4375,25.7254433
8.4375,26.3504464 C8.4375,26.953128 8.63839085,27.4553551 9.04017857,27.8571429
L21.1607143,39.9776786 C21.5848235,40.4017878 22.0870507,40.6138393 22.6674107,40.6138393
C23.2700923,40.6138393 23.78348,40.4017878 24.2075893,39.9776786 L42.3883929,21.796875
C42.7901806,21.3950873 42.9910714,20.8928602 42.9910714,20.2901786 L42.9910714,20.2901786 Z
M51.4285714,25.7142857 C51.4285714,30.3794876 50.2790294,34.6818999 47.9799107,38.6216518
C45.6807921,42.5614036 42.5614036,45.6807921 38.6216518,47.9799107 C34.6818999,50.2790294
30.3794876,51.4285714 25.7142857,51.4285714 C21.0490838,51.4285714 16.7466715,50.2790294
12.8069196,47.9799107 C8.8671678,45.6807921 5.74777935,42.5614036 3.44866071,38.6216518
C1.14954208,34.6818999 0,30.3794876 0,25.7142857 C0,21.0490838 1.14954208,16.7466715
3.44866071,12.8069196 C5.74777935,8.8671678 8.8671678,5.74777935 12.8069196,3.44866071
C16.7466715,1.14954208 21.0490838,0 25.7142857,0 C30.3794876,0 34.6818999,1.14954208
38.6216518,3.44866071 C42.5614036,5.74777935 45.6807921,8.8671678 47.9799107,12.8069196
C50.2790294,16.7466715 51.4285714,21.0490838 51.4285714,25.7142857 L51.4285714,25.7142857
```

```
Z"></path>';
Ndiv += '</svg><span class="ui-box-msg__title">¡Listo!</span>';
Ndiv += '<div class="syi-action-button">';
Ndiv += '<input type="submit" value="Continuar" class="syi-action-button__primary ui-btn ">';
Ndiv += '</div>';
Ndiv += '</div>';
Ndiv += '</div>';
jQuery(".syi-category-tree__column:last").after(Ndiv);
} ********************************************************************************/
```

Funciones JQuery/ Ajax/ PHP

```
/*********************************************************************************
* @ Jquery
* Creamos la @funcion add_categories_object que recibe 4 @parámetros, posteriormente Se definen dos
* variables myLevel y myName que toma el valor del nivel y la categoria.
*/
function add_categories_object(level, ml_url, cat_name, flag_to_add)
{
/* */
var myLevel = parseInt(level) + 1;
var myName = cat_name;

/*********************************************
* @función JQuery Ajax
* Creamos la funcion ajax de @metodo GET
* La @función en caso de que tenga éxito, requiere un valor llamado data, se crea una validación en
* caso de que flag_to_add sea verdadera.
* Se crea una variable llamada Ndiv para imprimir las categorías y se agregan valores a Ndiv.
* Posteriormente se crea una @función Jquery que recorre todas las categorías hijas de data y Se
* agrega una opción por cada categoría hija.
* Se agrega el valor de Ndiv al final de la última categoría, después Validamos si el nivel actual
* es mayor que uno, en caso de que asi sea, se agrega al tag con id = level:(mylevel-1).val(myname)
*
*/
jQuery.ajax({
type: "GET",
url: ml_url,
async: false,
success: function(data)
{
if (flag_to_add)
{
var Ndiv = '<div class="ui-box syi-category-tree__column cat subcat">';
Ndiv += '<div data-index="1" class="syi-category-tree__container ">';
Ndiv += '<select class="syi-category-tree__selector selected" id="level_'+ myLevel
+'" title="Elige una categoría" size="20" onChange="getCategory(this, \'child\', ' + myLevel + ');"
name="ml_categories[child][]">';
jQuery.each(data.children_categories,function(i,obj)
{
Ndiv +='<option class="syi-category-tree__option"
value='+obj.id+'>'+obj.name+'</option>';
});
Ndiv += '</select>';
Ndiv += '</div>';
Ndiv += '</div>';
jQuery(".syi-category-tree__column:last").after(Ndiv);
}
if (myLevel > 1)
{
jQuery("#level_" + (myLevel - 1)).val(myName);
}
```

```
}
});
}
*****************************************************************************/
```