

SIMPLIFICACION DE FORMULARIOS DE ACTIVIDADES

Inmaculada Montero



@inmaelearning



inmaelearning



get_context()

¿Qué nos pasa?

- Demasiadas configuraciones
- Campos no utilizados
- Usuarios no expertos
- Tiempos de creación
- Accesibilidad

Formulario complejo

➤ Agregando un nuevo URL a Tema 1

▶ Expandir todo

General

Nombre



URL externa



Seleccione un enlace...

Descripción



Muestra la descripción en la página del curso



▶ Apariencia

▶ Parámetros

▶ Ajustes comunes del módulo

▶ Restricciones de acceso

▶ Finalización de actividad

▶ Marcas

Guardar cambios y regresar al curso

Guardar cambios y mostrar

Cancelar

Formulario sencillo

➤ Agregando un nuevo URL a Tema 1 ⓘ

▼ General

Nombre



URL externa



Descripción



Gestionar ficheros



Muestra la descripción en la página del curso ⓘ

Guardar cambios y regresar al curso

Guardar cambios y mostrar

Cancelar

En este formulario hay campos obligatorios ⓘ .

Ventajas

Sencillo de instalar

Sencillo de desinstalar

Muy configurable

Facilita la usabilidad al usuario

Inconvenientes

Tabla nueva

Un módulo nuevo por tipo de actividad

Alteración de la navegación

Muchas configuraciones previas

Desarrollo

- Estructura de módulo Demo:

- lang
 - en
 - demo.php
- db
 - access.php
 - install.php
 - install.xml
- pix
 - Icon.png
 - Icon.svg
- index.php
- mod_form.php
- settings.php
- version.php



Base de datos

```
<?php
function xmldb_demo_install() {
    global $DB;
}

```

Función de
instalación sin
contenido

install.php
install.xml

Tabla con 3 campos básicos: id,
course, name

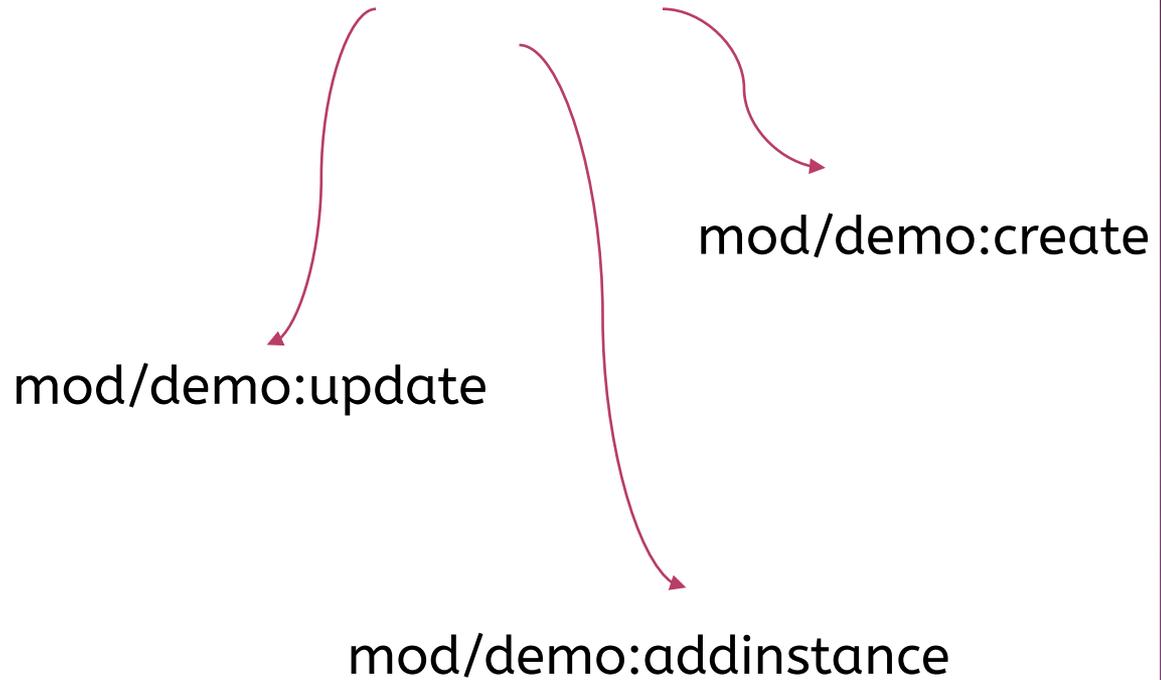
```
<?xml version="1.0" encoding="UTF-8" ?>
<XMLDB PATH="mod/demo/db" VERSION="20120122" COMMENT="XMLDB file for Moodle mod/demo"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="../../../../lib/xmldb/xmldb.xsd"
>
  <TABLES>
    <TABLE NAME="demo" COMMENT="Defines demo activities">
      <FIELDS>
        <FIELD NAME="id" TYPE="int" LENGTH="10" NOTNULL="true" SEQUENCE="true"/>
        <FIELD NAME="course" TYPE="int" LENGTH="10" NOTNULL="true" DEFAULT="0" SEQUENCE="false"/>
        <FIELD NAME="name" TYPE="char" LENGTH="255" NOTNULL="true" SEQUENCE="false"/>
      </FIELDS>
      <KEYS>
        <KEY NAME="primary" TYPE="primary" FIELDS="id"/>
      </KEYS>
      <INDEXES>
        <INDEX NAME="course" UNIQUE="false" FIELDS="course"/>
      </INDEXES>
    </TABLE>
  </TABLES>
</XMLDB>

```

Permisos

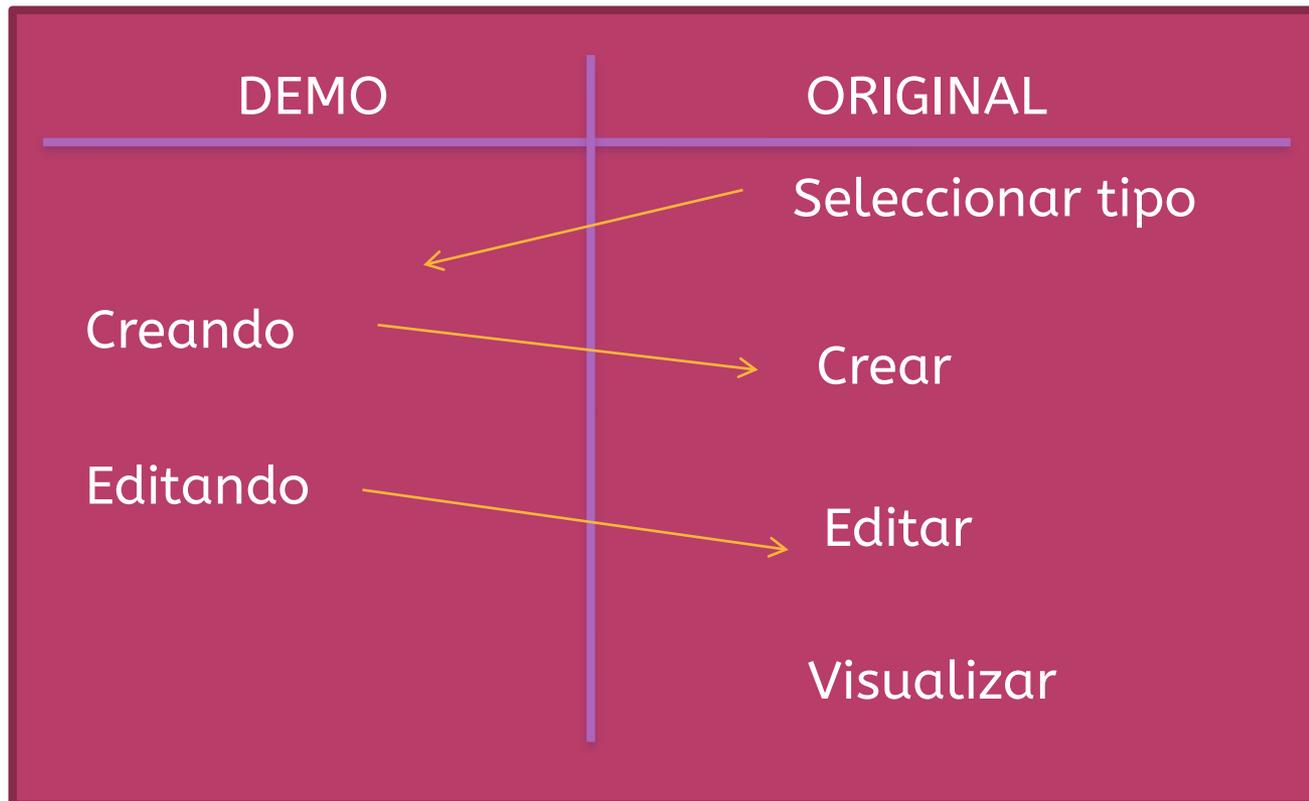
```
$capabilities = array(  
    'mod/demo:addinstance' => array(  
        'riskbitmask' => RISK_XSS,  
        'captype' => 'write',  
        'contextlevel' => CONTEXT_COURSE,  
        'archetypes' => array(  
            'editingteacher' => CAP_ALLOW,  
            'manager' => CAP_ALLOW  
        ),  
        'clonepermissionsfrom' => 'moodle/course:manageactivities'  
    ),  
    'mod/demo:create' => array(  
        'riskbitmask' => RISK_XSS,  
        'captype' => 'write',  
        'contextlevel' => CONTEXT_MODULE,  
        'archetypes' => array(  
            'teacher' => CAP_ALLOW,  
            'editingteacher' => CAP_ALLOW  
        ),  
    ),  
    'mod/demo:update' => array(  
        'riskbitmask' => RISK_XSS,  
        'captype' => 'write',  
        'contextlevel' => CONTEXT_MODULE,  
        'archetypes' => array(  
            'teacher' => CAP_ALLOW,  
            'editingteacher' => CAP_ALLOW  
        ),  
    ),  
);
```

access.php
Capabilities



*index

Simplemente contiene el *original* -> INCLUDE



mod/demo/lib

```
defined('MOODLE_INTERNAL') || die();

/**
 * Given an object containing all the necessary data,
 * (defined by the form in mod.html) this function
 * will create a new instance and return the id number
 * of the new instance.
 *
 * @param object $instance An object from the form in mod.html
 * @return int The id of the newly inserted url record
 */
function demo_add_instance($url) {
    return url_add_instance($url);
}

/**
 * Given an object containing all the necessary data,
 * (defined by the form in mod.html) this function
 * will update an existing instance with new data.
 *
 * @param object $instance An object from the form in mod.html
 * @return boolean Success/Fail
 */
function demo_update_instance($url) {
    return url_update_instance($url);
}

/**
 * Function to be run periodically according to the moodle cron
 * This function searches for things that need to be done, such
 * as sending out mail, toggling flags etc ...
 *
 * @uses $CFG
 * @return boolean
 * @todo Finish documenting this function
 */
function demo_cron() {
    return url_cron();
}
```

Funciones básicas:

- Añadir actividad
- Actualizar actividad

LIB DEMO

require_once

LIB
Original

settings

Dashboard / Site administration / Plugins / Activity modules / DEMO

DEMO

Frame height
demo | framesize

Default: 130

When a web page or an uploaded file is displayed within a frame, this value is the height (in pixels) of the top frame (which contains the navigation).

Password
demo | secretphrase

[Click to enter text](#)  

This secret phrase is used to produce encrypted code value that can be sent to some servers as a parameter. The encrypted code is produced by an md5 value of the current user IP address concatenated with your secret phrase. ie code = md5(IP.secretphrase). Please note that this is not reliable because IP address may change and is often shared by different computers.

Include role names in
parameters
demo | rolesinparams

Default: No

Enable if you want to include localized role names in list of available parameter variables.

Available display options

Specifies default height of popup windows.

Availability
demo | visible

Show on course page 

Default: Show on course page

If the availability is set to 'Show on course page', the activity or resource is available to students (subject to any access restrictions which may be set).

If the availability is set to 'Hide from students', the activity or resource is only available to users with permission to view hidden activities (by default, users with the role of teacher or non-editing teacher).

Save changes

mod_form

```
class mod_original_mod_form extends moodleform_mod {}
```



```
class mod_demo_mod_form extends mod_original_mod_form {}
```

- *definition:*
 - Campos predefinidos ocultos
 - Campos visibles limitador (Ej: Título, descripción, URL...)
 - Campos visibles

mod_form

- *standard_coursemodule_elements*
- validation
- add_completion_rules
- data_preprocessing
- completion_rule_enabled
- standard_grading_coursemodule_elements
- add_action_buttons

Muchas
simplificaciones



Código común a
todas actividades



Refactor en plugin
general

Idioma

- Añadir al lang el string de los permisos

```
$string['modulename'] = 'DEMO';  
$string['modulename_link'] = 'mod/url/view';  
$string['modulenameplural'] = 'DEMOS';  
$string['pluginname'] = 'DEMO';  
$string['demo:addinstance'] = 'Add a new URL DEMO resource';  
$string['demo:create'] = 'Create URL DEMO';  
$string['demo:update'] = 'Update URL DEMO';
```

- Nombre del módulo
- Nombre del plugin
- Link de view se mantiene
- Cadenas de configuraciones nuevas
- Cadenas de permisos nuevos
- Cadenas de configuraciones viejas (opcional)

version^{*}

- Añadir **dependencias** del módulo de actividad o recurso original y el plugin general (si procede).

```
/**
 * Demo module version information
 */
*
* @package      mod_demo
* @author      Inma Montero (@inmalearning)
* @license     http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
*/

defined('MOODLE_INTERNAL') || die();

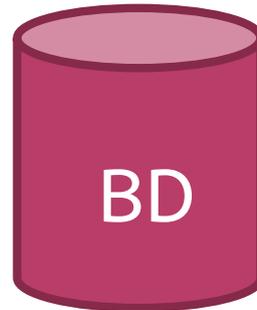
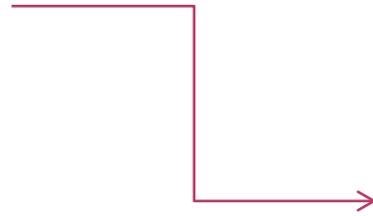
$plugin->component = 'mod_demo'; // Full name of the plugin (used for diagnostics).
$plugin->version   = 2018061700; // The current module version (Date: YYYYMMDDXX).
$plugin->requires = 2017110800; // Requires this Moodle version.
$plugin->cron      = 60;
$plugin->release  = '1.0 (Build 2018061700)';
$plugin->dependencies = array('mod_url' => 2017111300);

$plugin->dependencies = array('mod_url' => 2017111300);
$plugin->dependencies = array('mod_url' => 2017111300);
$plugin->dependencies = array('mod_url' => 2017111300);
$plugin->dependencies = array('mod_url' => 2017111300);
```

```
$plugin->dependencies = array('mod_original' => 2018010100);
```


Backup / Restore

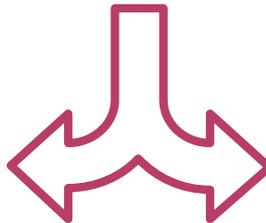
Crear actividad
simplificada



mdl_original

No guarda datos finales en BD ni en Moodledata

No
backup

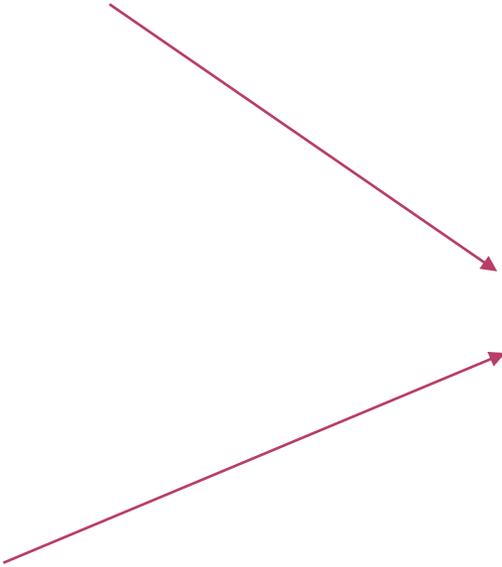


No
restore

Activación

Permiso mod/demo:addinstance -> CAP_PROHIBIT

Permiso mod/demo:create -> CAP_ALLOW



No aparece en el listado

Deshabilitado (No acceso
fácil a configuración)

Posibilidades

Formularios distintos por roles

Campos generales visibles	
Campos ocultos profesor	Campos visibles editor
Campos generales ocultos	

Posibilidades

- Settings personalizadas:
 - *Distintas en función del rol:*
 - Valor profesor → True
 - Valor editor → False
 - *Distintas en función del curso:*
 - Nota curso 1 → 10
 - Nota curso 2 → 100
- Settings de plugin adicional → *elegir activaciones*
- Añadir funcionalidades a Moodle para que los cursos pudieran tener una *opción de configuración* para habilitar o no los formularios simples.

¡GRACIAS!!



Inmaculada Montero



@inmaelearning



inmaelearning