



Descubre el poder de Joomla! 5



#JoomlaDayES2024

~~Parte 1: Desarrollar una aplicación de lista de deseos para Navidad~~

- ~~1. Planificando el componente~~
- ~~2. Creando las tablas en la base de datos~~
- ~~3. Entendiendo la estructura de carpetas~~
- ~~4. Programando el componente~~

Parte 2: Desarrollando nuestro servicio web

1. Editando nuestros deseos
2. Introducción a los servicios web
3. Enrutando nuestro servicio web
4. ¡A divertirse!

Añadiendo la vista de edición

<https://developingextensionsforjoomla5.com/jdayes2024/live/3-editview>

- Usamos la misma vista para las acciones de **editar** y **agregar**.
- Para los elementos individuales no usamos una consulta, sino la clase `Table`.
- Usamos la clase `Forms` para crear el formulario de manera sencilla.

Referencias



Capítulos 2 y 4

Editando deseos

<https://developingextensionsforjoomla5.com/jdayes2024/live/3-editview>

- Fichero: `src/Controller/DeseoController.php`

```
<?php
namespace Langulero\Component\Aiwfc\Administrator\Controller;
use Joomla\CMS\MVC\Controller\FormController;
\defined('_JEXEC') or die;
class DeseoController extends FormController
{
}
```

Referencias



Capítulo 2

Editando deseos

- Fichero: `src/Model/DeseoModel.php`

```
<?php
namespace Langulero\Component\Aiwfc\Administrator\Model;

use Joomla\CMS\MVC\Model\AdminModel;
use Joomla\CMS\Factory;

\defined('_JEXEC') or die;

class DeseoModel extends AdminModel
{
    public function getForm($data = array(), $loadData = true)
    {
        $form = $this->loadForm(
            'com_aiwfc.deseo',
            'deseo',
            [
                'control' => 'jform',
                'load_data' => $loadData
            ]
        );

        if (empty($form)) {
            return false;
        }

        return $form;
    }

    protected function loadFormData()
    {
        $app = Factory::getApplication();
        $data = $app->getUserState(
            'com_aiwfc.edit.deseo.data',
            []
        );
        if (empty($data)) {
            $data = $this->getItem();
        }
        return $data;
    }
}
```

Referencias



Capítulo 2

Editando deseos

- Fichero: `src/View/Deseo/HtmlView.php`

```
<?php
namespace Languero\Component\Aiwfc\Administrator\View\deseo;

use Joomla\CMS\Factory;
use Joomla\CMS\Toolbar\Toolbar;
use Joomla\CMS\Toolbar\ToolbarHelper;
use Joomla\CMS\MVC\View\GenericDataException;
use Joomla\CMS\MVC\View\HtmlView as BaseHtmlView;

\defined('_JEXEC') or die;

class HtmlView extends BaseHtmlView
{
    public $state;
    public $item;
    public $form;

    public function display($tpl = null): void
    {
        $this->form = $this->get('Form');
        $this->state = $this->get('State');
        $this->item = $this->get('Item');

        if (count($errors = $this->get('Errors'))) {
            throw new GenericDataException(implode('\n', $errors), 500);
        }

        $this->addToolbar();

        parent::display($tpl);
    }

    protected function addToolbar()
    {
        Factory::getApplication()->input->set('hidemainmenu', true);

        $isNew = ($this->item->id == 0);
        $toolbar = Toolbar::getInstance();

        ToolbarHelper::title(($isNew ? 'Add' : 'Edit'));

        if ($isNew) {
            $toolbar->apply('deseo.save');
        } else {
            $toolbar->apply('deseo.apply');
        }
        $toolbar->save('deseo.save');
    }
}
```

```
BAR_CLOSE');
```

Referencias



Capítulo 2

Editando deseos

Añadimos nuestro formulario: `forms/deseo.xml` .

```
<?xml version="1.0" encoding="UTF-8"?>
<form>
  <field name="id" type="hidden" />
  <field name="titulo" type="text" label="Título" required="true" />
  <field name="descripcion" type="editor" label="Descripción" required="false" />
  <field name="creado" type="calendar" default="NOW" label="Creado en" required="false" readonly="true"/>
  <field name="creado_por" type="user" label="Creado por" readonly="true" default="CURRENT" />
</form>
```

References



Capítulos 2 y 4

Editando deseos

Modificamos el layout del listado:

```
tmpl/deseos/default.php .
```

Reemplazando:

```
<div class="item-title">
    <?php echo $item->name; ?>
</div>
```

Por esto:

```
use Joomla\CMS\Router\Route;
```

```
<div class="item-title">
  <a href="<?php echo Route::_('index.php?option=com_aiwfc&view=deseo&layout=edit&id=' . (int) $item->id); ?>">
    <?php echo $task->titulo;?>
  </a>
</div>
```

References



Capítulo 2

Editando deseos

Añadimos el layout de edición: `tmpl/deseo/edit.php`

```
<?php
use Joomla\CMS\HTML\HTMLHelper;
use Joomla\CMS\Router\Route;

\defined('_JEXEC') or die;

HTMLHelper::_('behavior.formvalidator');
HTMLHelper::_('behavior.keepalive');

?>
<form action="<?php echo Route::_('index.php?option=com_aiwfc&view=deseo&layout=edit&id=' . (int) $this->item->id); ?>"
method="post" name="adminForm" id="deseo-form" class="form-validate">
  <div>
    <div class="row">
      <div class="col-md-9">
        <div class="row">
          <div class="col-md-6">
            <?php echo $this->form->renderField('id'); ?>
            <?php echo $this->form->renderField('titulo'); ?>
            <?php echo $this->form->renderField('descripcion'); ?>
            <?php echo $this->form->renderField('creado'); ?>
            <?php echo $this->form->renderField('creado_por'); ?>
          </div>
        </div>
      </div>
    </div>
    <input type="hidden" name="task" value="">
    <?php echo HTMLHelper::_('form.token'); ?>
  </form>
```

References



Capítulo 2

Introducción a los servicios web

- Forma de hablar entre máquinas
- Sin HTML, responde en JSON, XML...
- Difícil de probar en un navegador
- 4 verbos: POST, GET, PATCH, DELETE
- Sin sesión de usuario

References



Capítulo 6

Usando los servicios web en Joomla!

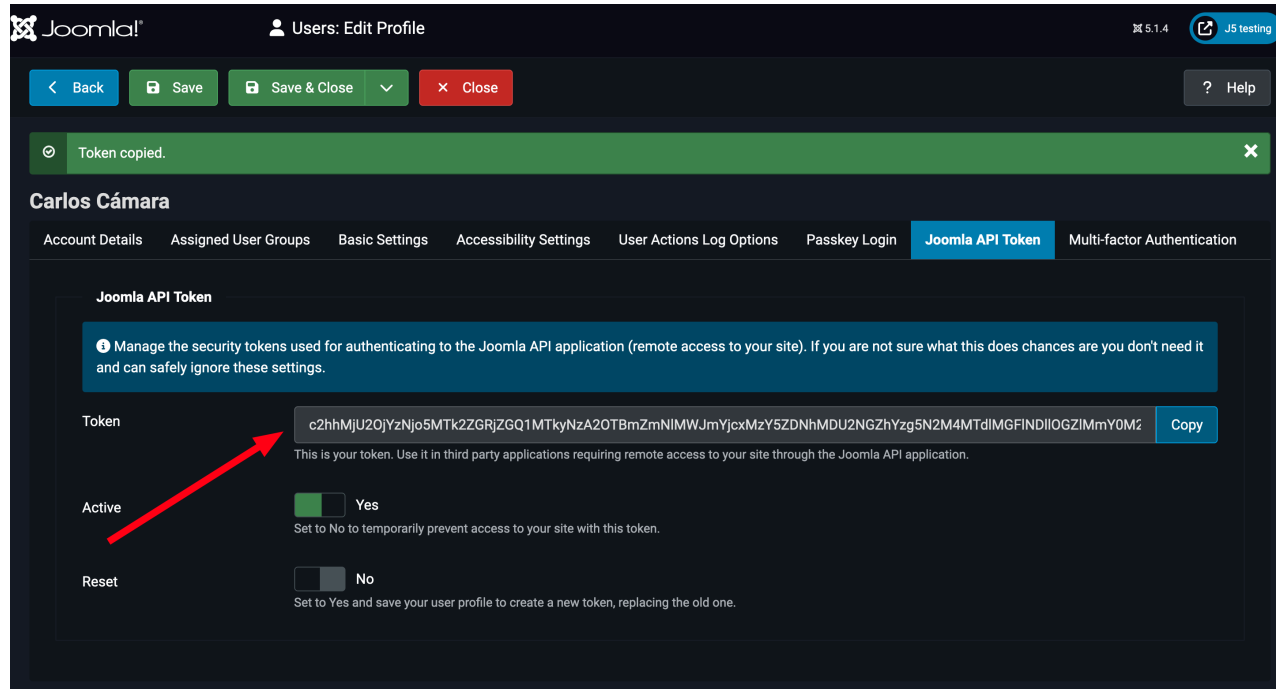
- POSTMAN, HOPPSCOTCH, PHP scripts, cURL, plugins de tu IDE...
- Necesitas crear un token para tu usuario de Joomla en cualquier caso.

References



Capítulo 6

Creando el token de API



References



Capítulo 6

Creando un punto en entrada para nuestro servicio web

<https://developingextensionsforjoomla5.com/jdayes2024/live/4-web-services-plugin>

- Las rutas se añaden con plugins del tipo `webservices` .
- Vamos a desarrollar el plugin y lo instalaremos con el método **Descubrir** . que usamos con el componente.

References



Capítulo 6

Creando el plugin de servicio web

Fichero de manifiesto:

plugins/webservices/aiwfc/aiwfc.xml :

```
<?xml version="1.0" encoding="UTF-8"?>
<extension type="plugin" group="webservices" method="upgrade">
  <name>plg_webservices_aiwfc</name>
  <author><![CDATA[Carlos Cámara]]></author>
  <authorEmail>carlos@hepta.es</authorEmail>
  <authorUrl>https://developingextensionsforjoomla5.com</authorUrl>
  <creationDate>2024-04-12</creationDate>
  <copyright>(C) 2024 Langulero SL.</copyright>
  <license>GNU General Public License version 2 or later; see LICENSE.txt</license>
  <version>0.1.0</version>
  <description><![CDATA[Plugin para enrutar el web service de la lista de deseos para Joomla!]]></description>
  <namespace path="src">Langulero\Plugin\WebServices\Aiwfc</namespace>
  <files>
    <folder plugin="aiwfc">services</folder>
    <folder>src</folder>
  </files>
</extension>
```

References



Capítulo 6

Fichero:

Creando el plugin de servicio web

plugins/webservices/aiwfc/services/provider.php :

```
<?php

use Joomla\CMS\Extension\PluginInterface;
use Joomla\CMS\Factory;
use Joomla\CMS\Plugin\PluginHelper;
use Joomla\DI\Container;
use Joomla\DI\ServiceProviderInterface;
use Joomla\Event\DispatcherInterface;
use Langulero\Plugin\WebServices\Aiwfc\Extension\Aiwfc;

\defined('_JEXEC') or die;

return new class () implements ServiceProviderInterface {
    public function register(Container $container): void
    {
        $container->set(
            PluginInterface::class,
            function (Container $container) {
                $plugin = new Aiwfc(
                    $container->get(DispatcherInterface::class),
                    (array) PluginHelper::getPlugin('webservices', 'aiwfc')
                );
                $plugin->setApplication(Factory::getApplication());

                return $plugin;
            }
        );
    }
};
```

References



Capítulo 6

Fichero:

Creando el plugin de servicio web

plugins/webservices/aiwfc/src/Extension/Aiwfc.php :

```
<?php
namespace Langulero\Plugin\WebServices\Aiwfc\Extension;
use Joomla\CMS\Plugin\CMSPlugin;
use Joomla\Router\Route;
\defined('_JEXEC') or die;
final class Aiwfc extends CMSPlugin
{
    public function onBeforeApiRoute(&$router)
    {
        $route = new Route(
            ['GET'],
            'v1/aiwfc/deseos/:id',
            'deseos.displayItem',
            ['id' => '(\d+)'],
            ['component' => 'com_aiwfc']
        );
        $router->addRoute($route);
        $router->createCRUDRoutes(
            'v1/aiwfc/deseos',
            'deseos.displayList',
            ['component' => 'com_aiwfc']
        );
    }
}
```

References



Capítulo 6

- Instalamos el plugin usando la función de **descubrir**

Gestionando las peticiones en nuestro componente

<https://developingextensionsforjoomla5.com/jdayes2024/live/5-api>

- En Joomla!, creamos el fichero

```
api/components/com_aiwfc/src/Controller/DeseosController.php :
```

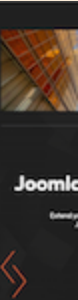
```
<?php

namespace Langulero\Component\Aiwfc\Api\Controller;

use Joomla\CMS\MVC\Controller\ApiController;

\defined('_JEXEC') or die;

class DeseosController extends ApiController
{
    protected $contentType = 'Deseos';
    protected $default_view = 'Deseos';
}
```



Creando el plugin de servicio web

Fichero:

api/components/com_aiwfc/src/View/JsonapiView.php :

```
<?php
namespace Langulero\Component\Aiwfc\Api\View\Deseos;
use Joomla\CMS\MVC\View\JsonApiView as BaseApiView;

class JsonapiView extends BaseApiView
{
    protected $fieldsToRenderList = [
        'id',
        'titulo',
        'description',
        'creado'
    ];
    protected $fieldsToRenderItem = [
        'id',
        'titulo',
        'descripcion'
    ];
}
```

References



Capítulo 6

Preparamos el manifiesto del componente

Modificamos el manifiesto para añadir la API:

```
<api>
  <files folder="api/components/com_aiwfc">
    <folder>src</folder>
  </files>
</api>
```

References



Capítulo 6

Reinstalando el componente

- Copiamos las carpetas a un lugar seguro:
 - `administrator/components/com_aiwfc`
 - `api/components/com_aiwfc`
- Desinstalamos el componente ⚠
- Copiamos de nuevo los ficheros
- Instalamos con **Descubrir**

Probando el Servicio Web

Fichero de pruebas HoppScotch:

<https://developingextensionsforjoomla5.com/jdayes2024/live/main/Hoppscotch>

En brazos de gigantes

- Libro online *Joomla Extension Development* by Nicholas Dionysopoulos
 - <https://www.dionysopoulos.me/book.html>
- Libro de Astrid: *Joomla 4 – Developing Extensions: Step by step to an working Joomla extension*
 - Disponible en Amazon
 - Disponible en Web Archive
- Documentación Joomla! para programadores
 - <https://manual.joomla.org>



¡GRACIAS!




Carlos Cámara

Desarrollador Joomla!

 carlos@hepta.es

 www.hepta.es

 Almería

