



WORDCAMP
BILBAO
2025

JORGE MONCLÚS

Conectando tu WordPress a un CRM

Sala 2 - 18:10

#WCBILBAO25 #WCBILBAO25 #WCBILBAO25 #W

Tu WordPress puede ser un gran aliado de tu CRM, surtiéndole de mucha información, como nuevos contactos y leadscorings. Y esto también funciona al revés, proveyendo a tu WordPress con información sobre el contacto que le ayude a vender. Pero primero debes:

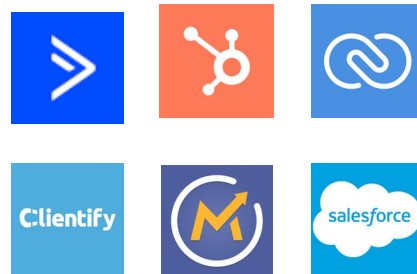
APRENDER A CONECTARLOS



En esta ponencia veremos las diferentes formas que permiten que se comuniquen entre sí estos dos elementos fundamentales de tu estrategia de marketing con ejemplos sencillos de algunos de los CRM más populares, como:

- ActiveCampaign
- HubSpot
- ZohoCRM
- Clientify
- Mautic
- Salesforce

Trataremos de ver formas de conexión que impliquen programación y ejemplos que no, para ver todas las posibilidades.



- Sí o sí tu CRM o, por lo menos el plan que escojas, **debe tener una API** y cuanto más completa mejor. Con buena documentación y a poder ser con alguna librería ya desarrollada en el lenguaje que vayas a usar.
- Sabed si **vuestro hosting puede conectarse a otros servidores**. No es muy normal, pero a veces están capados. Es bastante común con instituciones y grandes empresas preocupadas con la ciberseguridad. Hablad con el hosting para que meta la IP del CRM en algún tipo de lista blanca.
- Tratad de saber si tienen **servidores europeos**. Como están las cosas puede ahorraros muchos problemas si en el futuro hay que migrar todo a sistemas en Europa.
- Una búsqueda rápida os puede dar ideas de si vais a tener **comunidad, software libre o de pago y empresas desarrollando para ese CRM** a las que pedir ayuda.

Muchos CRM dan periodos de **prueba gratuitos de 15 días** o similar. No es ninguna perdida de tiempo **pedir esas pruebas y hacer algunos tests básicos**.

Migrar el día de mañana de un CRM a otro porque no cumple todas nuestras expectativas va a llevarte más tiempo que hacer unas pruebas básicas con el periodo de prueba.

La forma principal de conectar tu WordPress y tu CRM es un código de tracking. Suele ser meter un código parecido a Analytics en todas las páginas de la web. Hay muchas formas para meter estos códigos.

- Directa en **header.php** del tema.
- Con el hook «**wp_header**» en el `functions.php`.
- Hay **temas que te permiten meter código JavaScript** en las cabeceras web.
- Muchos CRM tienen **plugins** con integraciones básicas del código de tracking.
- Si ya tenéis **Google Tag Manager** podéis configurarlo como una etiqueta más.

Obligaciones legales: Muchos de estos códigos de tracking dejan cookies, así que tendrás que montarlo dentro de tu sistema de gestión de cookies y dar información en el aviso de cookies.

¿Cómo trabajan estos códigos? 🤖

Estos códigos detectan e identifican a los usuarios a partir de los clics en emails que hayan sido enviados desde el CRM.

Una vez clican, son identificados por el código de tracking y este deja una cookie que nos permite rastrearlo por la web en ese equipo.

Sin ese primer clic en una newsletter y otro tipo de email no hay identificación y no se recoge información. Un usuario de nuestro CRM, si entra directamente en nuestra web desde un ordenador sin la cookie, no será detectado.

¿Qué permiten estos códigos solo con meterlos? < ☐ : ☷

Este simple código os da mucho potencial, ya que nos permite saber las páginas que visita. Sabemos perfectamente que páginas ha visitado y con eso podemos:

- Enviarle correos si visita asiduamente una URL interesante.
- Meter puntos en un LeadScoring si visita contenidos categorizados de un determinado tema.
- Sacar contenidos personalizados al visitar ciertos contenidos.

El código de tracking (▮_▮)

Caso práctico (°Δ°)

Todos los enlaces en LinkedIn tienen unos parámetros «UTM» que les marca como que vienen de LinkedIn. Si el código detecta un usuario que visita una página tiene esos parámetros «UTM», le mete una etiqueta «usuario-linkedin».

Esa etiqueta la usamos luego para tener listados de emails para crear audiencias en LinkedIn y para mandar campañas para promocionar LinkedIn, pidiéndoles que den me gusta y compartan.



```
1  
2  
3 https://www.spri.eus/teknologia-komunikazioa/  
4 ceit-zentro-teknologikoak-300-enplegu-/?  
5 utm\_source=linkedin&utm\_medium=social&  
6 utm\_campaign=linkedinLS  
7  
8
```

La forma más común de conexión entre WordPress y Un CRM es **mediante formularios**. Esta conexión se puede hacer mediante diferentes formas.

- Incrustados
- Conectados con un plugin
- Conectados con un desarrollo propio
- Mixtos



The screenshot shows a WordPress interface for managing a form titled "Formulario Inscripción". At the top right is a blue button labeled "Acciones". Below the title, there are two buttons: "publicada" (highlighted in green) and "borrador". A statistics section displays the following data:

Metric	Value
Impresiones	362
Conversiones	43
Contactos creados	23
Empresas creadas	28

Below the statistics is a tabbed interface with three tabs: "Vista previa" (selected), "Publicación", and "Conversion Funnel". The "Vista previa" tab shows a form titled "INSCRIPCIÓN JORNADA / EMAN IZENA JARDUNALDIA". The form fields are:

- Nombre *
Tu nombre
- Apellidos *
Tus apellidos
- Empresa *
Tu empresa
- Email *
Tu correo electrónico

At the bottom of the form is a red button labeled "Quiero inscribirme al curso / Izena eman nahi dut".

La mayoría de CRM dan códigos para incrustar formularios en tu web. Son códigos muy sencillos de usar, generas el formulario y cortas y pegas el código que te dan en tu web, en páginas, widgets, ...

Ventajas:

- Muy fácil de crear e integrar y exigen pocos conocimientos y gratis.
- La conexión es directa, no debería haber fallos.
- Ellos se encargan de meter la cookie de rastreo o similares para poder seguir al usuario.

Desventajas:

- Cuando quieres meter campos especiales que no están en el CRM, te obliga a crear campos personalizados en el CRM para guardar esa información. A la larga creas montones de campos que casi ni se usan.
- No tienen buena integración en el diseño de tu WordPress y suelen quedar un poco pegote.
- No permiten conectar a terceros. Los formularios incrustados se conectan difícilmente con elementos de terceros. Tengo una fundación de una empresa grande que todos los contactos recogidos con los formularios deben registrarse en el CRM que usan ellos (Clientify) y con el de la empresa matriz (Salesforce). Los formularios incrustados de Clientify no permiten hacer una conexión también a Salesforce cumpliendo los requerimientos que Salesforce exige.
- Si cae el CRM caen los formularios de tu web. No suelen fallar, pero puede pasar.

Conectados con un plugin ^ˆ○ˆ○^ˆ

Como siempre en temas de WordPress, siempre ha habido alguien antes con tu mismo problema y que ha creado un plugin para solucionarlo, así que plugin conectores de CRM hay unos cuantos. Son más o menos amplios y permiten más o menos opciones. Pero todos hacen más o menos las mismas funciones. Meter los datos del usuario como contacto o empresa en tu CRM.

Luego los habrá que dan elementos condicionales, lanzar automatizaciones, etc. pero la esencia es la misma.

Ventajas:

- **El tiempo de desarrollo y configuración es muy bajo**, por no decir mínimo. Instalar, activar y pocos minutos tenemos un formulario conectado a nuestro CRM.
- **Precio asequible.**

Desventajas:

- **Puedes hacer lo que te deje el plugin.** ni más ni menos. Al principio sirven, pero según usas más tu CRM se suelen quedar cortos.

Conectados con un desarrollo propio ()

La opción de desarrollar nuestro propio sistema es una buena decisión, pero no está al alcance de todos.

Si los formularios están desarrollados con algún plugin como Gravity Forms o Contact Form 7, asegurarnos que el plugin de formularios que estéis usando tenga «hooks» que os permitan meter fácilmente vuestro desarrollo dentro del plugin de formulario.

Por ejemplo, Contact Form 7 tiene un «hook» llamado «mail_sent» que permite meter tu código tras terminar todo el proceso.

Ventajas:

- **Podemos meter toda la personalización que queramos** y hacer que interactúen muchos plugins de tu WordPress. Nos permiten depurar mucho mejor los datos del formulario.
- También **permite interactuar varios Plugins**. Hay plugins que conectan CF7 con varios CRM y otros plugins que conectan WooCommerce con tu CRM, pero si queremos que nos envíen el contenido del carrito de compra de WooCommerce cuando rellena el formulario de consulta creado con CF7 no hay plugin que lo conecte todo.
- **Permite montar varias conexiones**. No es común, pero puede darse algunas veces que tengas que hacer varias conexiones a diferentes CRM y puede haber colisiones entre diferentes plugins, cada uno de que tira contra un CRM distinto.

Conectados con un desarrollo propio ☐ (☐) ☐

Desventajas:

- **Supone un mayor coste y tiempo de implementación.** Aunque los costes son cada vez más baratos debido a que puedes aprovechar el código ya creado.

Consejos:

- No tenemos que partir de cero, podemos trabajar sobre plugins ya existentes que hacen la conexión y sobre ellos hacer nuestro desarrollo propio.
- Los CRM no suelen funcionar al momento, acumulan tareas y una cola las procesa de forma que puede haber un desfase de minutos entre un disparador y que se ejecute las acciones asociadas.
- Si los formularios tienen que enviar algún email al momento con algún dato o confirmación debe hacerlo vuestro desarrollo y el plugin que uséis. Si no es importante que el envío sea inmediato puede hacerlo a través del CRM de forma que se registre aperturas, clics, etc. y puedas montar flujos y embudos.

Otra forma bastante común de conectar tu web en WordPress con tu CRM es crear un editor de perfil en tu web para que las propias usuarias puedan editar sus datos, sus preferencias y sus boletines de forma que ellas directamente nos den la información que luego vamos a usar.

Algunos ejemplos:

- [SPRI](#)
- [AED](#)

Consideraciones

- **No son desarrollos complicados**, son casi siempre tema de POST/GET de los datos y montarlos en un formulario. Se pueden complicar cuando metemos fotos o documentos como CV en PDF, por ejemplo.
- **No debemos almacenar contraseñas en nuestro CRM**, lo suyo es usar el login/registro de WordPress o generar algún tipo de «hash» temporal que se envíe al correo, así la gestión de seguridad la hace el correo.

¿Cómo tratar los datos?

Lo importante en esos desarrollos es **definir bien qué elementos va a tener el formulario y cómo van a poder rellenar**. Datos muy relevantes como pueden ser el tamaño de tu empresa, puede que no interese dejarse de mano de las clientes que editan su perfil.

La mejor opción es **cerrar lo más posible la entrada de datos**. Si podéis cerrar las opciones con desplegabes, checkboxes, etc. mucho mejor que textos libres. Pensad que una ciudad puede escribirse de muchas formas: Galdakano, Galdacano, Galdakao, etc.

Os diría que **si tenéis dudas con ciertos campos, los duplicáis** y luego ver como funcionan las usuarias, tomáis una decisión de mantenerlo separado o fusionáis. Siguiendo el ejemplo anterior del tamaño de empresa, podemos tener el campo del CRM «tamaño_empresa» que usa marketing para sus segmentos y creamos otro para «tamaño_empresa_perfil» para el perfil de usuaria.

> Rellenado al 76%

El perfil es bueno.
Trata de mejorarlo para recibir los contenidos que mejor se ajustan a tus preferencias.

MIS INTERESES

Recibir solo aquello que es importante para ti. Ni más ni menos.

☐ Ciberseguridad
☐ Digitalización

☐ Emprendimiento
☐ Financiación

☐ I+D
☐ Infraestructuras

☒ Innovación
☐ Internacionalización

☐ Invertir en Euskadi
☐ Sostenibilidad Medioambiental

☐ Me interesan los contenidos de todas estas temáticas.

MIS DATOS

Email

monclaus.jorge@gmail.com

Provincia *

Bizkaia

Nombre *

Jorge

Nombre de empresa

Enutt S.L.

Apellidos *

Monclaus

Sector

Energía

Teléfono

123456789

¿En qué idioma quieres recibirnos?

☒ Castellano
☒ Euskera
☒ Seleccionar/Deseleccionar ambos

¿CON CUAL DE ESTOS PERFILES DE EMPRESA TE IDENTIFICAS?

☐ Personas emprendedoras
☐ Autónomos/as

☐ Micropyme (1-10)
☐ Pequeña empresa (10-50)

☐ Mediana empresa (50-250)
☐ Gran empresa (+250)

☐ Agentes de la RVCTI
☐ Asociaciones

☐ Estudiantes
☐ Inversores

☐ Agentes intermedios
☐ Me interesan los contenidos de todas estas temáticas.

¿Cuántas más sobre ti para poder ofrecerte las ayudas, eventos y servicios de los que te puedes beneficiar?

☐ Guardar

10 de Mayo, 2025

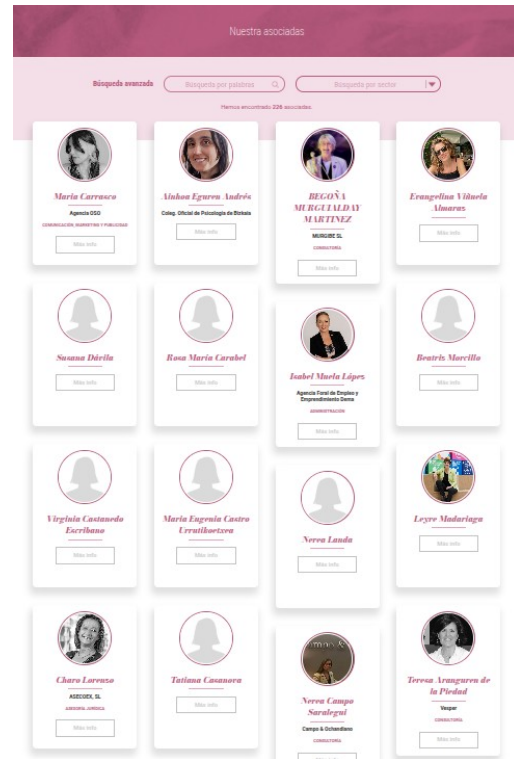
Sacando información del CRM para mostrarlo en WordPress <:3)~~~

Lo mismo que tu WordPress alimenta a tu CRM, habrá veces en que el CRM va a alimentar nuestra web. Este tipo de conexiones van a necesitar desarrollos propios y acceso a API. Las funcionalidades pueden muchas, aunque normalmente solo extraen información.

Caso práctico 1: Tenemos una [asociación](#) que tiene etiquetada a todas sus socias. La web muestra las fichas de las asociadas con sus datos más importantes. Esa información se saca del CRM y cuando se modifica en el CRM se modifica en la web.

Caso práctico 2: Se desarrolló un plugin que creaba un mini-sistema de contenido publicitario personalizado. Cada contacto tenía una serie de «LeadScorings» basados en sus intereses. En el admin se podían crear contenidos muy sencillos (titular, foto y texto) y asignarle un «LeadScoring» y un máximo y un mínimo. Si un contacto identificado con puntos en ese interés dentro de los parámetros establecidos, visitaba la web veía esos contenidos en determinadas partes de la web como el «sidebar» o el «footer». No eran muy intrusivos porque eran publicidad interna, pero lo interesante es que estaba bastante personalizada.

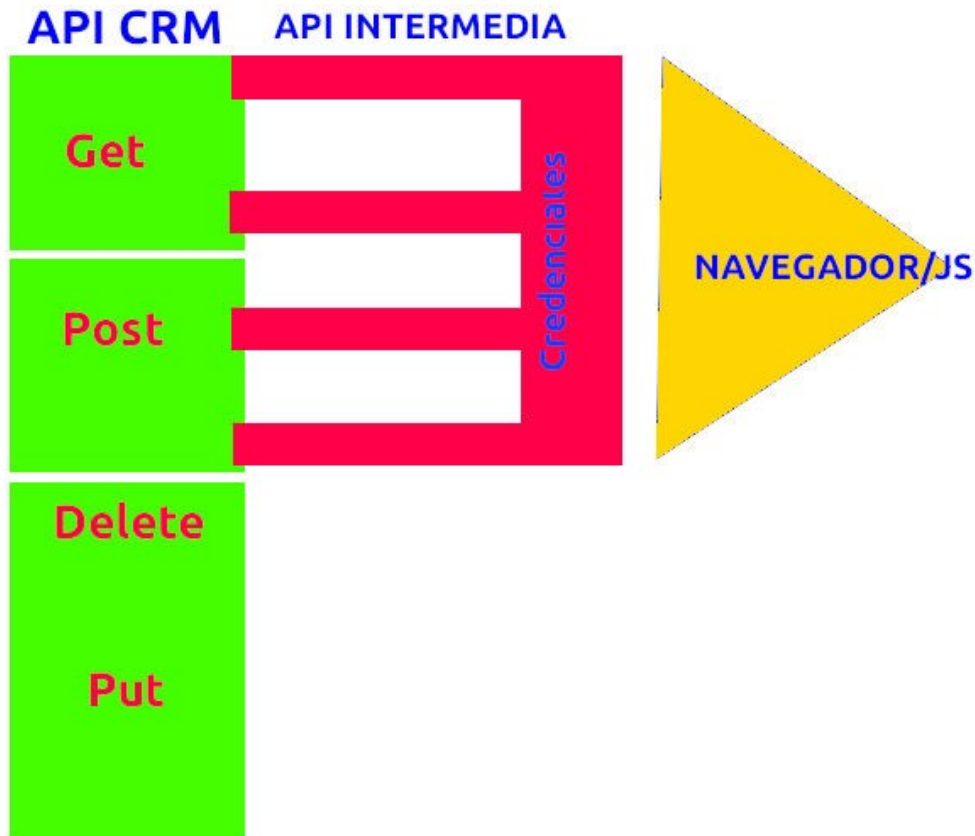
Consejo: Si no es importante que los datos estén actualizados al minuto trata de cachearlos lo máximo posible. El listado de socios del caso práctico puede pedirse una vez al día y tirar de ese cacheo hasta el día siguiente



Conexión directa a la API o crear un middleware c[]

Si vas a trabajar de servidor a servidor, no hay problemas de atacar directamente a la API del CRM, pero **cuando se trabaja desde el navegador**, por ejemplo con eventos de JavaScript, **las credenciales de la API pueden quedar expuestas**.

Para evitar esto **se puede crear una API intermedia** a la que ataque el navegador y que solo ofrezca los «endpoints» que nos interesen de la API del CRM.



Conexión directa a la API o crear un middleware c[]

Usando la API-REST de WordPress |[•---•]|

Con la API-REST de WordPress es muy fácil montar una API intermedia, ya que la propia API-Rest de WordPress te da un buen framework de desarrollo y este desarrollo solo tiene que recoger los mismos datos que la API del CRM y reenviárselos usando las credenciales.

El principal trabajo es decidir que endpoints de la API del CRM quiero usar y necesito y si es seguro poder usarlo en abierto.

Consejos y consideraciones:

- **Mejor POST que GET.** Aunque parezca al revés es más seguro los POST que los GET. Que metan datos es malo, pero que los saquen es un problemón.
- **Capa extra de seguridad.** Podemos meter restricciones de seguridad que la API del CRM no tenga. Si hay un número inusual de peticiones desde la IP puedes bajar esa IP.
- **Estadísticas.** Puede ser interesante unos logs de consumo de la API intermedia de los luego sacar estadísticas para la gente de marketing.

- **Cread un campo de fecha de última modificación.** Los CRM suelen guardar esta información, pero no discriminan entre conexiones por API y directamente por web. Con este campo personalizado tendréis control de qué contactos habéis tocado desde WordPress y cuáles han sido tocados de otras formas.
- **Usad las notas.** Muchos CRM te permiten crear notas de cada contacto, usad esas notas para guardar información variada, por ejemplo, el contenido de las áreas de texto de un formulario. Ponéis un título explicativo y luego el texto que escribió la persona que relleno el formulario. Siempre que se pueda, **no montéis campos específicos para ese tipo de cosas.**
- **Si puedes hacerse con una automatización del CRM, mejor que programándolo.** Al conectar se puede hacer que el script que desarrolléis haga todas las tareas, como meter etiquetas, apuntar a listas, meter puntos en un LeadScoring, etc. O simplemente que lance un disparador de una automatización y esta ejecute todo lo anterior. Piensa que **cambiar la programación solo podrá hacerlo una desarrolladora y cambiar una automatización con una interfaz gráfica, puede hacerlo cualquier persona de marketing con unos conocimientos básicos.**
- **Usad los sistemas de almacenamiento de ficheros de PDF, DOC, ODT, etc. que ofrecen los CRM.** Es muy tentador subir un fichero a tu web y usar ese enlace en todas partes. Pero no da gran información. Subiéndolo al sistema de archivos del CRM puedes saber quién y cuándo lo ha descargado si está entre tus contactos y si no, como, mínimo te dará unas buenas estadísticas de descargas.

Soy un **desarrollador web con más de 20 años de experiencia**, muchos de ellos trabajando con WordPress. De hecho, mis primeros proyectos fueron en B2Evo, el origen de WordPress.

Trabajo en Eñutt Comunicación desde hace 5 años, principalmente desarrollando en temas de CRM para clientes como SPRI, IHOBE, BEAZ o Gaztenpresa.

Mis enlaces:

- <https://bsky.app/profile/gwannon.com>
- <https://github.com/gwannon>
- <https://codepen.io/gwannon>
- <https://www.linkedin.com/in/jorgemonclus/>

Enlaces

- **Ponencia:** Todo el código fuente de la ponencia para que puedas usarlo como quieras. <https://github.com/gwannon/WordCampBilbao2025>

Plugins

- **FormsCRM:** Plugin gratuito que permite hacer una conexión entre varios plugins de formularios como «Contact Form 7» o «WP Forms» y CRM como «Holded» o «Clientify». <https://es.wordpress.org/plugins/formscrm/>
- **Add-on de Gravity Forms para Active Campaign:** Add-on para Gravity Forms que conecta tus formularios con Active Campaign y tiene bastantes opciones de conexión. <https://www.activecampaign.com/apps/gravity-forms-integration>

Librerías

- **PHPClientifyAPI:** Clase de PHP para manejar los contactos del CRM de Clientify. <https://github.com/gwannon/PHPClientifyAPI>
- **PHPActiveCampaignAPI:** Clase de PHP para manejar contactos del CRM a ActiveCampaign <https://github.com/gwannon/PHPActiveCampaignAPI>