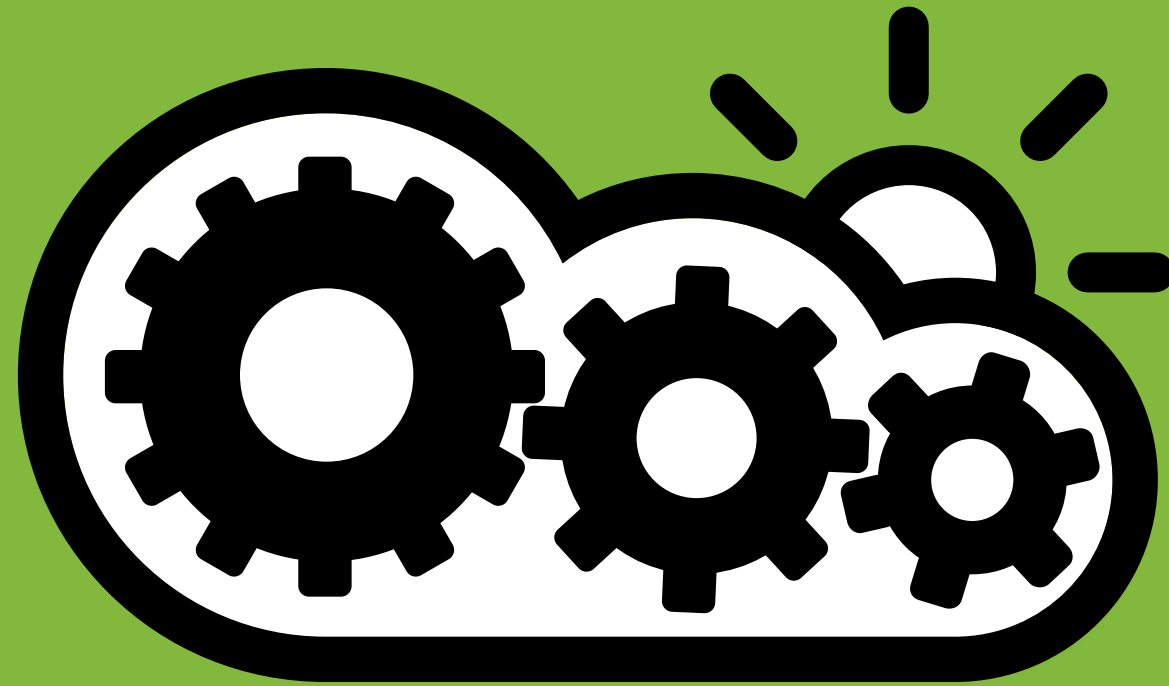


# Cloud para tu juego en una tarde



Ignacio Segura  
SIASAR · Banco Mundial  
nacho@isegura.es

# Este soy yo



Diseñador gráfico de formación.

Me pagué los estudios programando.

Aprendí lo que era la programación viendo a mi hermano con un ZX Spectrum. Yo tenía unos ocho años.

Hablo con fluidez “designspeak” y “codewargon”. Mi nerdscore es inmedible.

Mi último trabajo fue SIASAR, una plataforma de data-mining para el Banco Mundial de la cual dependen varias aplicaciones cliente.

Actualmente estoy en periodo sabático para cuidar de mi hijo, lo cual me deja tiempo para proyectos personales.

nacho@isegura.es

Twitter: @nachenko

LinkedIn: [www.linkedin.com/in/isegura/](http://www.linkedin.com/in/isegura/)

# Mi trabajo consiste en

1 - Jefe de producto / responsable / alguien presenta un problema.

2 - Yo presento una solución.

3 - A veces, esa solución requiere código. Otras veces, diseño + código.

En cualquier caso, mi trabajo es solucionar y anticipar problemas.

Mi conocimiento de programación es sobre todo PHP y Javascript. Ahora estoy introduciéndome en React. En general, me interesa más el frontend que el backend.

# He encontrado un problema

Queremos que nuestro juego guarde cosas online: records de puntuación, logros, hojas de personaje, objetos únicos, valoraciones de los propios jugadores sobre cuáles son los mejores escenarios, etc etc.

Pero para nosotros, eso es un problema enorme. Esa tecnología es completamente diferente a lo que usamos diariamente. Somos un estudio pequeño, no podemos desviar ni tiempo ni dinero a ese problema.

El conocimiento que tienen en su área de especialización no les ayuda en este problema. **El conocimiento que adquieran resolviendo ese problema por sí mismos no les ayudará en el desarrollo del juego. La solución no será óptima.**

Necesitan una solución que les permita no desviarse de su objetivo principal.

Ni siquiera Riot Games intenta ocuparse de todos los aspectos de su tecnología con su propio personal. Para algunos aspectos relacionados con servicios online y web, contratan empresas externas, como por ejemplo X-Team, para los cuales yo trabajé dos años (pero no en el equipo de Riot).



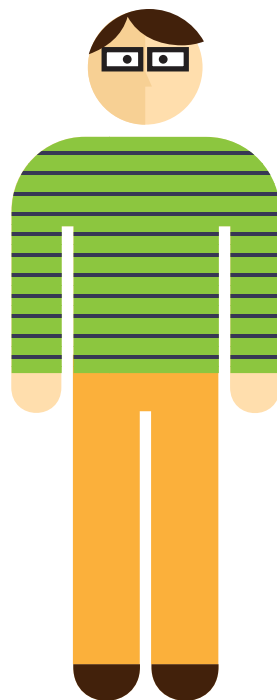
# Para el cual tengo una solución

Necesitáis una cosa que por arte de magia os proporcione un almacén online, y os proporcione métodos para gestionar los datos.

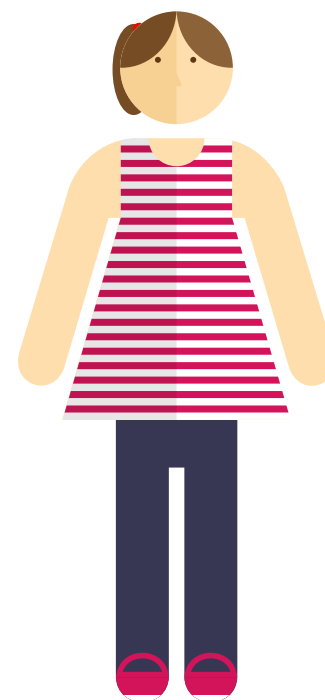
Traducido a nerd, una DB, un middleware y una API pública.



Qué salao



Verdad que sí



Es necesario encontrar una herramienta que proporcione:

- Una base de datos.
- Gestión de la información y la estructura de esa base de datos.
- Métodos para leer y escribir en esa base de datos.
- Documentación sobre su uso, incluso para nuestros casos particulares.
- SEGURIDAD básica.

Todo ello con un panel de control visual, que nos ahorre tocar un código en el que no estamos especializados.

A ser posible, pública, sin coste y con el futuro asegurado.



Drupal<sup>TM</sup>

# Qué es Drupal 8 y qué problema resuelve



Drupal es un gestor de contenidos Open Source. Los gestores de contenidos suelen tener dos funciones:

1. Dar a editores de contenido una herramienta sencilla para publicar sus textos, imágenes y vídeos.
2. Generar una web a partir de ese contenido.

A partir de la versión 8, Drupal relegó la creación de webs a un segundo plano y puso como prioridad el contenido:

1. Ponérselo más fácil a los editores.
2. Ser capaz de entregar y recibir datos para cualquier dispositivo capaz de conectarse a Internet.

Toda la información, módulos, distribuciones, temas, perfiles de instalación, documentación oficial, paquetes de idiomas e incluso usuarios y empresas contribuidoras puede encontrarse en [drupal.org](https://drupal.org)

Mi perfil en [drupal.org](https://drupal.org) es [www.drupal.org/u/nachenko](https://www.drupal.org/u/nachenko)



# Cosas que Drupal 8 puede hacer por ti

Sin programar una línea de código



- Tablas de records globales y por países para juegos de speed run.
- Gestión de objetos coleccionables y únicos para RPGs.
- Grabaciones de partidas.
- Hojas de personaje asociadas a un usuario.
- Gestionar que un usuario tenga varios personajes.
- Gestión de traducciones de los textos de tu juego.
- Guardado de partidas en la nube.
- Inventario de logros y medallas por usuario.
- Autenticación de los logros del usuario. Si un logro existe en local, pero no en remoto, el jugador ha hackeado el juego.
- Centro de documentación con histórico de versiones de los textos
- Publicación de noticias relacionadas con tu juego.
- Un entorno colaborativo para la redacción y organización de textos, diálogos, etc que luego se puede exportar a XML o JSON.
- Un backend unificado para tu juego y para el sitio web de éste.


El único tipo de uso que no recomiendo es aquel que requiera respuesta inmediata, como un chat, debido a la arquitectura de PHP. Para ese caso particular, intentaría integrar una tecnología ya existente o construiría algo en NodeJS, que es una tecnología con la que ya he tenido contacto.






# Qué hace falta para poner Drupal 8 en marcha


Drupal 8 está escrito en PHP, que es soportado por prácticamente la totalidad de las compañías de hosting. La tecnología es la misma que usan Wordpress o Joomla. Hay varias maneras de instalarlo, pero la versión resumida es: cualquiera que haya instalado alguna vez Wordpress sabe instalar Drupal 8.



Drupal 8 soporta Apache y Nginx. Funciona con bases de datos MySQL, PostgreSQL y MariaDB. Puede funcionar sobre PHP 5.6, pero se recomienda como mínimo 7.1. La diferencia de velocidad entre ambas versiones es considerable. La Memoria mínima para funcionar son 64 MB pero se recomiendan 128, o 256 si el sitio va a procesar muchas imágenes.



# Drupal viene personalizado de serie



Drupal 8 se expande mediante la instalación y configuración de módulos adicionales. Sin embargo, para los usos más comunes existen “distribuciones”: paquetes “todo en uno” que instalan y configuran Drupal 8 y módulos adicionales para servir a un propósito concreto. Hay distribuciones para comercio electrónico, intranets, agencias gubernamentales, ligas deportivas, periódicos online... y para proyectos en los que sólo necesitamos un almacén de datos, una API pública y las menos complicaciones posibles porque en realidad no hemos visto Drupal en la vida.

La distribución que necesitamos es ContentaCMS.



Existen más distribuciones similares. Otra muy conocida es Reservoir, pero el proyecto no ha tenido ninguna actividad en los últimos meses en GitHub. Aunque no hay comunicado oficial al respecto, el proyecto parece abandonado.

# Lo que vosotros tenéis que poner



Además de un sitio donde guardar vuestro Drupal 8, la adaptación de vuestra aplicación sigue corriendo de vuestra cuenta.


Necesitaréis:

- Ser capaces de realizar peticiones HTTP(S).
- Ser capaces de entregar y recibir información en formato JSON.

La cuenta de GitHub de ContentaCMS está llena de proyectos de ejemplo en Angular, React, Vue, Gatsby, Elm, Ionic... También hay un contenedor Docker.

También hay documentación con ejemplos de uso de la JSON API:  
[www.drupal.org/docs/8/modules/json-api](http://www.drupal.org/docs/8/modules/json-api)

Drupal 8 corre en prácticamente cualquier hosting mínimamente profesional del mundo capaz de ejecutar PHP 7. También funciona en PHP 5, pero la diferencia de velocidad es abismal.



**“A little less  
conversation,  
a little more  
action, please”**

**Elvis Presley**



# Ejemplo: record de Speedrun

Supongamos que estamos haciendo un juego de speedrun y nos interesa que los records sean públicos.



Primero vamos a crear un usuario de pruebas...  
[video: User creation and login.mp4]

Vamos a crear un tipo de contenido “record” y vamos a crear un primer registro usando la web.  
[video: Creating Record type and one record.mp4]

Ahora vamos a leer un record, guardar uno desde la API y leer la lista de records de nuevo. Pero no sé cómo.

**La API de Drupal 8  
se documenta sola 😎.**

# Leemos. Guardamos.

Leemos todos los records siguiendo la documentación generada automáticamente.

[video: Getting all records.mp4]

Ahora, creamos un nuevo record.

[video: adding one record.mp4]

Podéis encontrar documentación completa “how-to” en:

[www.drupal.org/docs/8/modules/json-api](http://www.drupal.org/docs/8/modules/json-api)



# Cambiamos de opinión



Supongamos que queremos que nuestro registro de records guarde más cosas. Por ejemplo, queremos que guarde también información sobre la innecesaria destrucción causada en el escenario.

- Número de personas pequeñas espachurradas por objetos grandes.
- Personas golpeadas con otras personas.
- Strikes, como en los bolos, pero con cosas que no son bolos.
- Casas reacondicionadas como solares.
- Vacas puestas en órbita geosincrónica.

Y lo queremos hacer a lo Quake III Arena, con medallas.  
[video: Creating new fields.mp4 - minuto 8:10]

	<b>Accuracy:</b> Kind of like a sharp shooter medal, this is awarded when your percentage of hits-to-shots is more than 50%.
	<b>Excellent:</b> If you make two frags within two seconds, you'll receive this medal and a resounding "excellent" by the announcer.
	<b>Frag:</b> You'll see this medal if you're lucky enough to get to a frag count of 100.
	<b>Gauntlet:</b> Since it's difficult to actually frag someone with the gauntlet, you'll receive this medal if you humiliate someone by doing just that.
	<b>Impressive:</b> Everyone knows that the railgun requires the most skill. That's why this medal is awarded for making two consecutive hits with it.
	<b>Perfect:</b> You'll receive this medal if you've managed the almost-impossible task of winning an arena without getting killed once yourself.



# Datos que Drupal 8 sabe manejar

Con el core o añadiendo algún módulo contribuido por la comunidad, Drupal sabe manejar los siguientes tipos de datos:



Números enteros y flotantes  
Texto  
Fechas  
Timestamps  
Referencias a usuarios u otros contenidos  
Tags  
Enlaces externos  
Emails

Teléfonos  
Países  
Monedas  
Unidades de medida física  
Imágenes  
Vídeo  
Audio  
Otros ficheros  
Puntos de geolocalización



La lista no es exhaustiva. La he rellenado de memoria.

Yo he contribuido al de medidas físicas para Drupal 7, por cierto :D

# Anti-cheating



Drupal es un software libre utilizado por muchas organizaciones públicas y privadas, por lo que algunas de ellas gestionan información confidencial o datos bancarios. A causa de esto, hay mucho esfuerzo para hacer de Drupal una herramienta segura.

Sin embargo, estas estrategias de seguridad no tienen nada que ver con una seguridad “anti-cheating”.

Para evitar que la gente intente hacer llamadas manipuladas a la API, se puede, por ejemplo, añadir un campo que contenga algún tipo de checksum o similar sobre los datos recibidos, que validamos en el servidor.

Por desgracia, esta estrategia sí que requiere código a medida. No es difícil si se sabe por dónde empezar. Es un tema más apropiado para un taller de tipo “Drupal and Beers”. Wink wink nudge nudge.

Providing a custom validation constraint:

<https://www.drupal.org/docs/8/api/entity-validation-api/providing-a-custom-validation-constraint>

# Me voy a quedar tirado?

TL;DR: No.



Respuesta larga:

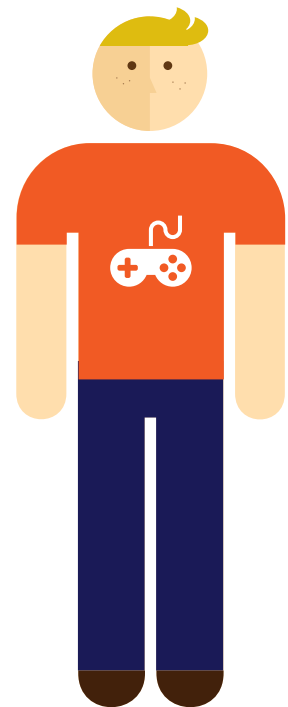






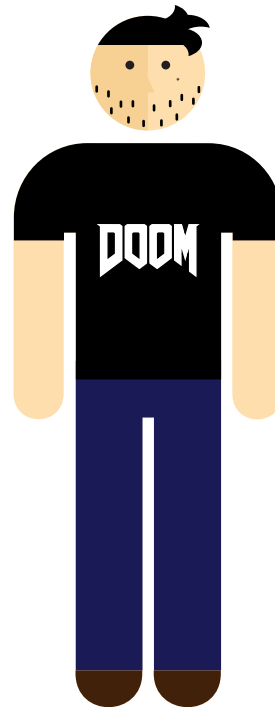


**¡No me  
lo creo!**

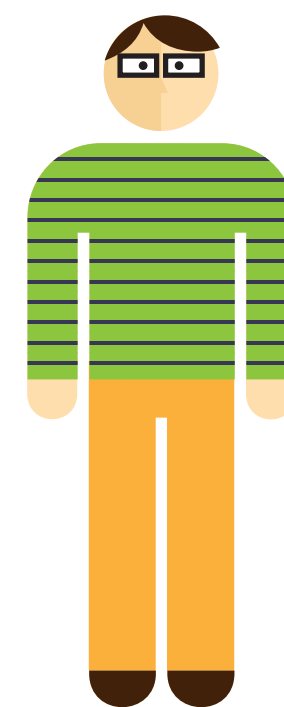


Tú mismo con tu mecanismo, digooooo...  
ni falta que hace. Lo puedes probar en  
**[demojam.niknak.es](https://demojam.niknak.es)**  
user/pass: **admin / banana**

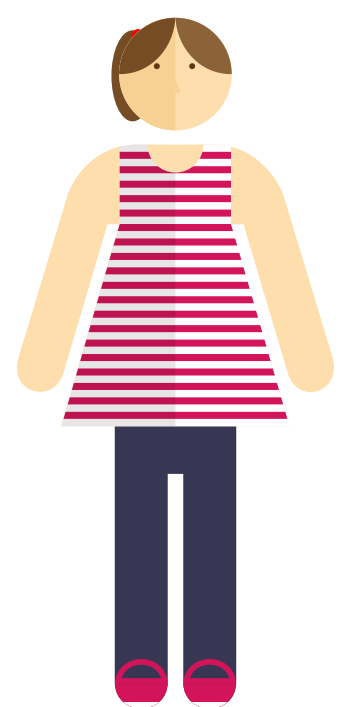
Y puedes bajarte el proyecto y las  
diapositivas en Github:  
**[github.com/ignaciosegura/demojam](https://github.com/ignaciosegura/demojam)**



¿Ves como es  
mu salao?



Mu salao, sí.





nacho@isegura.es

### **Cosas random**

Twitter: @nachenko

### **Mis cosas con corbata**

LinkedIn: [www.linkedin.com/in/isegura](http://www.linkedin.com/in/isegura)

### **Mis cosas pinta pega colorear**

<https://www.behance.net/isegura>

**Cosas random largas como: empresarios  
lloricas, Starcraft, ganar dinero contando  
mentiras, juegos de Spectrum, niños más  
listos que los adultos...**

Medium: [medium.com/@isegura](http://medium.com/@isegura)

