

# Digital Experience Platform (DXP)

## Índice

Resumen, .....	2
Digital Experience Platform, DXP, .....	3
Headless CMS, .....	8
ANEXO 1: Arquitectura MACH, .....	15

## Resumen

La evolución de los modelos de relación con los clientes, integrando todos los canales actuales que pueden usarse, han generado un nuevo concepto de gestión llamado Plataforma de Experiencia Digital (DXP).

Un DXP es un software empresarial que busca satisfacer las necesidades de las empresas en su proceso de transformación digital, con el objetivo de dotar de una mejor experiencia de cliente, repercutiendo en un ahorro de costes para la compañía.

Un DXP se basa en una arquitectura llamada MACH que tiene 4 ejes fundamentales:

- **Microservicios:** piezas individuales de funcionalidad empresarial independientes.
- **API-First:** funcionalidad expuesta a través de API.
- **SaaS nativo en Cloud:** software como servicio desarrolla directamente en dicho entorno.
- **Headless:** la experiencia del usuario de frontend está completamente desacoplada de la lógica de backend.

Los beneficios inmediatos cuantitativos y cualitativos de esta aproximación son:

- **Centralización y colaboración.** Contenidos, imagen, prácticas, recursos, ...
- **Mejor time to market**
- **Menor riesgo.** Operativo, imagen, impacto, recursos humanos, etc
- **Ejecuta la mejor estrategia existente en el mercado.** Ajuste inmediato a la estrategia de la empresa
- **Sin actualizaciones o migraciones.** Sin deuda tecnológica.
- **Personalización, innovación.** Implantación inmediata de nuevas tendencias de mercado.
- **Ejecución perfecta.** Sin incidencias

# Digital Experience Platform, DXP

## Introducción

En la década de los 2000 existían soluciones WCM (Web Content Management) y CMS que servían a usuarios y compañías para publicar administrar y entregar contenido en línea de la forma más fácil posible y en muchas ocasiones sin deber tener amplios conocimientos en programación para llevar las ideas a la pantalla. Mas tarde harían su aparición las webs 2.0 y las nuevas tecnologías de dispositivos como smartphones, tablets o relojes inteligentes que constituyeron un gran cambio en cómo se debía disponer y estructurar la información en cada uno de los sitios webs, por no hablar de la infinidad de nuevas opciones que permitían nuevas tecnologías como los GPS u otros sensores asociados a cada dispositivo.

Con la incipiente transformación digital de la sociedad y de las compañías, ha surgido una tercera etapa que ha traído consigo las plataformas de experiencias digitales o DXP.

Las DXP pueden ser un solo producto, aunque lo más normal es que se constituyan en un conjunto de ellos que trabajan de forma coordinada para ofrecer un mejor rendimiento. Las plataformas de experiencia digital otorgan la estructura necesaria para que las empresas digitalicen las operaciones de negocios, ofrezcan experiencias conectadas a los clientes, y reúnan datos procesables sobre los usuarios.



Las plataformas de experiencia digitales son la evolución natural de los CMS, pero con una orientación 100% a crear una experiencia centrada en el cliente para que encuentre el contenido que desea de una forma más rápida y personalizada.

Por tanto, las Plataformas de Experiencia Digitales nacieron de la necesidad y las limitaciones que los CMS tenían en cuanto a gestión de las experiencias de los clientes en profundidad y constituyen la evolución en todo ello.

Los DXP están equipados para manejar contenido y gestionarlo desde todas las plataformas involucradas en la cadena de valor a gran escala como, por ejemplo, sincronizar administrar y enviar contenido a través de verticales como websites, quioscos, anuncios portales de clientes y dispositivos IOT entre otros.

Los CMS proporcionan un control integral del sitio web de las empresas en las que diferentes usuarios pueden acceder y ver el contenido pero sin una vista central del sistema. Un CMS aporta una vista única de todo el contenido reduciendo la capacidad de un mantenimiento específico de un departamento de TI y proporcionando una experiencia totalmente integrada utilizando APIs y otros protocolos de comunicación.

Las mismas tendencias de mercado que han producido la evolución desde los CMS a los DXP generaron que otras aplicaciones evolucionarán en la misma línea:

- **DXP desde CMS:** Enfocados a la hiperpersonalización de los mensajes que buscan generar conocimiento e interés y acelerar el proceso de compra de un bien o servicio dirigiendo ofertas personalizadas según el perfil de los usuarios.
- **DXP desde sitios webs:** para relaciones postventa con el objetivo de mantenerlas a largo plazo, son una herramienta útil en cualquier departamento de atención al cliente para lograr retención y fidelización.
- **DXP desde eCommerce:** usadas por los comercios electrónicos ofreciendo catálogos de productos, inventarios, carritos de compra dinámicos, pasarelas de pago y logística.
- **DXP desde CDP (Customer Data Platform):** Sacan partido a los datos obtenidos de la actividad empresarial para orientar y tomar mejores decisiones en las acciones digitales de la compañía. Cuentan con la capacidad de romper los data silos e incorporar analítica predictiva y experiencias adaptadas a cada usuario.
- **LXP desde LMS:** orientados a crear experiencias formativas integradas y dirigidas a personas
- **DXP desde Herramientas de Marketing:** dónde proveedores de soluciones estándar de marketing, con múltiples herramientas se postulan como soluciones DXP. Este tipo de situaciones suelen tener situaciones complejas comparativamente al ser de alto coste, compleja integración y poco abiertas a terceros.

## Definición de Plataforma de Experiencia Digital

*Una plataforma de experiencia digital (DXP) es un software empresarial que busca satisfacer las necesidades de las empresas en su proceso de transformación digital, con el objetivo de dotar de una mejor experiencia de cliente, repercutiendo en un ahorro de costes para la compañía.*

Gartner define las plataformas de experiencia digitales como un “conjunto integrado de tecnologías, basado en una plataforma común, que proporciona a una amplia gama de audiencias mediante un acceso coherente, seguro y personalizado a la información y las aplicaciones a través de muchos puntos de contacto digitales”.



Source: Gartner (February 2024)

Gartner

El enfoque Gartner se centra en muy pocos actores, y principalmente grandes empresas originadas desde el Marketing. Por ese motivo en su valoración no aparecen actores referentes del mercado global como Contentful, Prismic o Strapi, entre otros. El único que aparece en su cuadrante es Magnolia.

Un cuadrante equivalente generado por los usuarios en entornos como CB Insights, Crunchbase, Capterra, Pitch Book, saaseworthy.com, muestra resultados con más proveedores y sobre todo más originados desde un enfoque CMS.



El objetivo de las DXP es conseguir que el negocio digital sea capaz de adaptarse lo más rápido posible a las necesidades del cliente y pueda acceder a toda la información necesaria definiendo la mejor estrategia digital para su desarrollo, implementarla en un tiempo récord e incorporar nuevos canales, tendencias o tecnologías de forma inmediata.

La experiencia Digital de un usuario es cada uno de los puntos de contacto que una empresa tiene con el cliente a través de un Customer Journey establecido previamente y por eso, seleccionar una DXP significa adoptar la tecnología idónea que sea capaz de ofrecer las mayores interacciones digitales posibles a través de múltiples canales.

Usar una solución en lugar de varias hará que se optimice el trabajo gracias a su enfoque todo en uno.

#### Características y funcionalidades de un DXP

Las DXP persiguen maximizar los caminos hacia la entrega de experiencias personalizadas estando conectadas entre si y operando a través de diferentes canales de diferentes zonas geográficas y variedad de idiomas. Entre las funcionalidades que deberían tener se encuentran:

- **Gestión de Contenido:** Las interacciones digitales que los potenciales clientes tienen con el sitio/canal ocurren a través del contenido que se es capaz de ofrecer y del interés que se les genera. Descubrir soluciones o servicios mediante información relevante previa a la compra de un producto, interactuar con la marca, conversar con las empresas mediante chat o dejar algún tipo de comentario o valoración.
- **Personalización:** la importancia de comunicarse con los clientes hace imprescindible ser capaz de ofrecer experiencias personalizadas, adaptadas lo máximo posible a las preferencias y gustos de cada uno e incluso que permitan que el cliente se adapte a la empresa.
- **Sitios y aplicaciones:** el contenido es un elemento muy atractivo, pero de nada sirve si no está accesible a través del medio o canal y el momento adecuado para hacerlo visible y útil.
- **Inteligencia contextual y relevancia:** Como un motor de perfil del cliente, traducción de idiomas o integración omnicanal, por ejemplo.
- **e-Commerce y e-Business:** el comercio electrónico es cada vez más importante en los proyectos de transformación digital de las empresas ya que son parte esencial en la facturación y monetización de los

productos y servicios de las empresas. Por ello, soluciones orientadas al comercio, como PCM (Product Content Management) con pagos, facturación, compras o suscripciones deben poder ser integrados en cualquier DXP.

- **Interoperabilidad y extensibilidad:** una buena solución DXP debe contar con la capacidad de interoperar con otros sistemas y servicios empresariales de forma rápida, sostenible y rentable, sin tener que construirlo específicamente para cada integración.
  - **Uso de diferentes herramientas:** Chatbots, aplicaciones móviles, redes sociales, integración entre canales, automatización de procesos o automatización de marketing.

## Capacidades DXP



## Beneficios de la implementación de plataformas de experiencia digital

- Unifica las experiencias del cliente. Proyecta experiencias conectadas a cada uno de tus clientes y consolida una relación con ellos desde todos los puntos de contacto ofreciéndoles una experiencia omnicanal que incremente su fidelidad.
  - Agiliza las operaciones de negocio. Apoya a los equipos y procesos de toda tu organización dando servicio a tus clientes de una forma más eficiente. Aloja múltiples sistemas en una sola plataforma para unificar la experiencia a los clientes.
  - Evoluciona la estrategia digital. Personaliza más rápidamente con una plataforma flexible que ofrece servicios modulares y reutilizables, con una serie de herramientas y APIs abiertas.



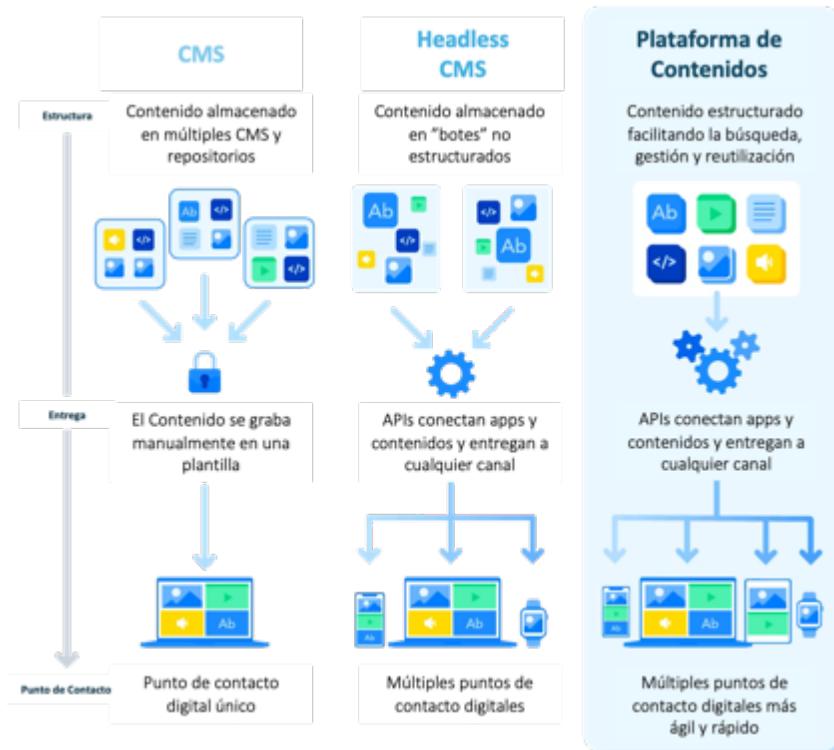
## Headless CMS

### Introducción

La arquitectura tradicional del sistema de gestión de contenidos (CMS) alguna vez fue el estándar para el desarrollo web, pero a medida que las necesidades comerciales y las expectativas de los clientes se volvieron más complejas, la tecnología evolucionó hacia un estándar más común utilizado hoy en día que se conoce como Headless CMS o Sistema de Gestión de Contenido sin cabeza. A continuación, se explica el porqué de este término.

**Un Headless CMS es un sistema de gestión de contenidos que separa la capa de presentación (dónde se presenta el contenido) del backend (dónde se gestiona el contenido).**

Permite administrar contenido en un solo lugar y poder implementar ese contenido en cualquier canal digital elegido. Separar el frontend del backend libera los contenidos de cómo van a ser presentados, lo que facilita a los especialistas en marketing administrar dichos contenidos de forma independiente y a los creadores hacerlo más rápido, automatizar cambios y administrar lo digital a escala. En un CMS tradicional, el contenido está integrado con el código y enterrado en silos, lo que hace que su reutilización (y la creación de experiencias digitales modernas) sea casi imposible.



### Los orígenes del Headless CMS

Para comprender qué es una solución CMS sin cabeza, es útil observar primero el sistema de gestión de contenido tradicional y para qué fue diseñado. Los CMS tradicionales existen desde los primeros días del desarrollo web. Plataformas como Wordpress, Drupal y Sitecore fueron diseñadas para almacenar y presentar elementos de contenido como texto, imágenes y videos específicamente en sitios web.

El enfoque tradicional de CMS para gestionar contenido pone todo en un gran grupo: contenido, imágenes, HTML y CSS. Esto hacía imposible reutilizar el contenido porque estaba mezclado con código.

A medida que los canales y dispositivos digitales han evolucionado, ha surgido la necesidad de soluciones más flexibles. Ahora, las empresas están desarrollando sitios web, aplicaciones móviles, pantallas digitales, interfaces conversacionales y más. Mientras tanto, el CMS tradicional no logró seguir el ritmo. ¿Por qué? Porque un CMS

organiza el contenido en marcos orientados a páginas web, lo que hace imposible que el mismo contenido se ajuste a otras plataformas o software digitales.

La siguiente tabla sirve como referencia rápida para de cómo un CMS tradicional se compara con su evolución, el Headless CMS, cuando se trata de funciones críticas necesarias para realizar el trabajo correctamente.

	CMS Tradicional	Headless CMS
Hosting	On Premise	Cloud
Cultura de desarrollo	Foco en proyecto	Foco en producto
Modelo de Contenidos	Creados para una página concreta	“Bloque” para varios usos/salidas
Equipos soportados	Limitados	Sin límite
Orientación	Específica (one-to-one)	Retulización (one-to-many)
Workflow	Cascada	Agile
Actualizaciones	Calendarizadas	Continuas
Backend	Monolítico, todo en uno	Microservicios, best-in-class
Inversión	Frontales de usuario de alto coste	Pilotos rápidos
Dependencia/Deuda Técnica	Inherente al sistema	Gestionada

## Cómo funciona una solución Headless CMS

El contenido alojado en un Headless CMS se entrega a través de API para una visualización perfecta en cualquier sitio, dispositivo u otro punto de contacto digital. Esto hace que el contenido de un Headless CMS sea infinitamente reutilizable, sin importar la experiencia omnicanal del cliente que se utilice hoy en día o los canales que surjan en el futuro. Esto es diferente de Wordpress y otros CMS monolíticos que combinan estrechamente el frontend con el backend, manteniéndote atrapado en cómo se puede (y no se puede) mostrar el contenido.

Sólo hay que pensar en una arquitectura Headless y API como esta: el trabajo principal de un Headless CMS es almacenar y administrar el contenido. Realmente no importa lo que se quiera hacer con ese contenido. La función principal de las plataformas de visualización, como un sitio web o una aplicación móvil, es presentar contenido a las personas. Realmente no les importa cómo se almacena o gestiona ese contenido. Las API son los puntos de conexión mágicos que permiten que estos sistemas backend (por ejemplo, el Headless CMS) y sistemas frontend (por ejemplo, un sitio web) se comuniquen de la manera específica que un equipo digital desea.



## Headless CMS y CMS desacoplado

A medida se conocen diferentes tipos de soluciones de gestión de contenidos, aparece otro término: "CMS desacoplado" que puede confundirse con un Headless.

Como su nombre indica, la característica definitoria de un CMS desacoplado es que el backend y el frontend están separados. Esta separación es similar a un Headless CMS, sin embargo, un CMS desacoplado viene con una cabeza, pero usarlo es completamente opcional. Algunos usan los términos indistintamente, pero no son idénticos. La principal diferencia es que un Headless CMS no incluye ninguna capa de presentación, sino que permite a los desarrolladores decidir cómo les gustaría mostrar el contenido.

Esto suele realizarse a través de marcos JSON interactivos como React o Vue.js o generadores de sitios estáticos como Gatsby.

## Mejores prácticas para configurar soluciones Headless CMS

Si bien una solución de software Headless CMS permite implementar contenido en cualquier capa de presentación, no resuelve un problema subyacente: no da estructura al contenido. Mientras el contenido no esté estructurado, no puede reutilizarse fácilmente en diferentes plataformas y canales. Contenido estructurado es un término general que se refiere al contenido que se divide en pequeños bloques de construcción, se organiza de forma predecible y se clasifica con metadatos.

Un ejemplo para ilustrar cómo se suele utilizar el contenido no estructurado, podría ser una página web. El contenido no estructurado combina todo el contenido y el código que crean esa página web. Esto suele hacerse en un editor WYSIWYG, que significa "lo que ves es lo que obtienes". Los creadores de contenido digital suelen estar familiarizados con los WYSIWYG, ya que les brinda la posibilidad de realizar ediciones de contenido en el backend. Esto crea páginas web perfectamente buenas, sin embargo, el contenido se augea a ese formato. Lo que ves es lo que obtienes... y nada más.

Un enfoque de contenido estructurado separa los diversos elementos de esa página en distintos componentes, como autor, título, cuerpo, imagen, descripción de la imagen, definiciones, información de comercio electrónico, precios de productos, términos y condiciones, entre otros. Todos estos componentes aún se pueden ensamblar para crear la misma página web, pero también se pueden volver a ensamblar para crear varias iteraciones de la página web, personalizarlas para una audiencia distinta, reorganizarlas para una campaña específica o recortarlas para una experiencia móvil.

La forma en que este contenido estructurado se genera en el backend es a través de algo llamado modelo de contenido. Un modelo de contenido define y organiza todos los diferentes tipos de contenido que utiliza una organización. En el ejemplo anterior, todos los diferentes elementos enumerados se consideran tipos de contenido distintos (título del autor, imagen, título, texto del cuerpo, etc.). Los modelos de contenido se componen de estos

tipos de contenido y pueden reutilizarlos para adaptarlos a las necesidades únicas de cada organización, de modo que los creadores de contenido no se quedan atrapados con las plantillas de páginas demasiado prescriptivas de un CMS tradicional.

Esta cultura de gestión en un Headless CMS, aún fácil de explicar y comprender, tiene unos retos importantes para su óptima implementación, máximo aprovechamiento y rendimiento.

RETOS		
CMS	vs.	DXP
Impulsar la colaboración con los equipos de IT		Impulsar la adopción en toda la organización
Mantener el contenido actualizado y al día		Asegurar que el contenido sea de fácil acceso, coherente y con una fuente única y confiable.
Empoderar a los propietarios de sitios web		Empoderar a los autores de contenido
Gobernanza de activos gráficos, propiedad, regulación, control interno y cumplimiento		Construir un sistema de diseño que promueva la agilidad y el cumplimiento.
Ofrecer una experiencia digital al cliente		Proporcionar a los clientes una experiencia digital significativa, fluida y coherente
Agregar nueva funcionalidad a medida que el negocio crece		Aprovechar una funcionalidad sólida para impulsar los objetivos comerciales

## Beneficios del Headless CMS

Definido el Headless CMS, cómo funciona y las formas en que es una mejor solución para la gestión de contenidos que un CMS tradicional, se puede detectar rápidamente el valor que puede aportar su implementación en la organización.

Los elementos que se mencionan en los siguientes puntos son los más evidentes, y fácilmente se aprecian importantes impactos cuantitativos (recursos, inversión, tiempo,...) y cualitativos (imagen, marca, unificación, pruebas,...).

### Unificación de todo tipo de contenido en un único punto

Al hacer que el contenido sea infinitamente reutilizable, el contenido estructurado de un Headless CMS elimina procesos manuales como copiar y pegar, duplicar, ubicar en diferentes silos por localización, negocio, geografía, etcétera.

Esto hace que la edición sea mucho más sencilla: cambiar un texto o imagen en un solo lugar provoca que ese cambio se aplique en cualquier lugar donde se encuentre distribuido. Con todo el contenido almacenado en un centro de contenido único, los editores de contenido pueden aplicar fácilmente el **principio COPE: crear una vez, publicar en todas partes** (*Create Once, Publish Everywhere*). Otra forma de aplicar este principio es editar una vez y actualizar en todas partes.

### Habilita flujos de trabajo colaborativos

Al separar el frontend (o capa de presentación) del backend y organizar el contenido con modelos de contenido reutilizables, un Headless CMS permite a los editores y desarrolladores de contenido trabajar en paralelo.

Los editores de contenido pueden actualizar ágilmente el contenido en todos los canales sin el apoyo de los desarrolladores, lo que los libera para abordar trabajos más estratégicamente importantes. Esto ofrece una ventaja competitiva para las empresas que desean aumentar su velocidad de comercialización y aprovechar al máximo el tiempo de sus desarrolladores.

## Aumentar la reutilización y la escalabilidad.

El Headless CMS hace que la reutilización del contenido sea muy sencilla. La reutilización del contenido es un componente clave para aprovechar al máximo los recursos invertidos en la creación de contenido. Cuando todo el contenido es accesible para su uso en cualquier terminal digital, los equipos digitales pueden reutilizar el contenido en todos los dispositivos y canales. También permite que el contenido se amplíe en diferentes regiones, casos de uso y campañas. El contenido se puede optimizar para diversas experiencias de usuario mediante la integración de herramientas de personalización y localización para garantizar que el contenido correcto llegue a la audiencia adecuada en el momento adecuado.

## Características clave de la arquitectura Headless

La arquitectura Headless describe un enfoque modular para crear aplicaciones. En este enfoque, el frontend de la aplicación, la parte que dicta la experiencia del usuario, es decir, lo que el usuario final puede ver e interactuar, está separado del backend, el lado del servidor de la aplicación que dicta la estructura, define la lógica y almacena datos.

Con este desacoplamiento, el contenido, los medios y cualquier otra cosa alojada en el backend no se limita a un solo dispositivo, diseño o canal. Más bien, se puede entregar a tantas interfaces o "cabezas" diferentes como sea necesario. Con esta arquitectura, es posible desarrollar una aplicación web y luego enviar la misma lógica de negocios y funciones a otro canal (por ejemplo, una aplicación móvil) donde los usuarios pueden disfrutarla mientras viajan sin exceso de esfuerzo de desarrollo.

Esta definición inspira un *déjà vu*, debida a que es un enfoque similar al de otros marcos relacionados, como la arquitectura modular de microservicio, API-First, nativa de la nube y Headless (MACH, ver anexo 1). Todos estos términos describen una única arquitectura modular, similar a la arquitectura Headless, que prefiere el uso de elementos de tecnología a las grandes suites cerradas.

Las principales características, aunque no las únicas, de una arquitectura Headless son las siguientes.

### 1. Utiliza API para conectar el front-end y el backend.

Si bien el frontend y el backend no están estrechamente acoplados en las arquitecturas headless, sí necesitan comunicarse entre sí, y ahí es donde entran las API. Las interfaces de programación de aplicaciones (API) son intermediarios de software que permiten la comunicación entre aplicaciones. En el ámbito de la arquitectura sin cabeza, los desarrolladores configuran API específicas, a menudo API REST, para pasar datos sin procesar o contenido desde los servicios de lógica empresarial backend a las capas de presentación frontend.

### 2. Permite la presencia omnicanal

La razón principal por la que las empresas adoptan la arquitectura Headless durante la transformación digital es porque les permite crear y gestionar más fácilmente experiencias digitales multicanal. Con el Internet de las cosas (IoT) en constante expansión para incluir aplicaciones web, aplicaciones móviles, tecnología portátil, asistentes de voz, AR/VR, la Inteligencia Artificial y lo que venga después, el único reinado digital de la aplicación web ha terminado oficialmente y, con él, la preferencia para arquitecturas integradas únicas.

Los sistemas de gestión de contenidos tradicionales, como Drupal, WordPress, Adobe Experience Manager y Sitecore, que se crearon para un solo caso de uso (que admiten sitios web estrechamente acoplados) tienen dificultades para adaptarse a estos nuevos canales. Con texto, activos digitales, CSS, HTML, JavaScript y otros elementos de la experiencia y mensajería digital de una marca diseñados para un único punto de contacto (por ejemplo, una publicación de blog, una página de inicio, una página de inicio), la oportunidad de reutilizar el contenido en nuevos dispositivos y canales se convierte en un reto.

Si bien se puede introducir otro CMS para cada nuevo canal digital, esto significa más herramientas para administrar, más gastos administrativos para los desarrolladores y creadores de contenido y un mayor compromiso financiero.

### 3. Se integra fácilmente con otras herramientas

La razón secundaria por la que las empresas seleccionan una arquitectura headless es que se integra bien con otras herramientas. Con la opción de introducir tecnologías de código abierto, de terceros y personalizadas o incluso una

suite completa, los equipos pueden crear un conjunto tecnológico de elementos que aborde las necesidades de proyectos o preferencias presentes y futuros.

Cada herramienta integrada, que se denomina microservicios, como bloques de construcción que se pueden agregar, eliminar o reconfigurar para desarrollar cualquier experiencia digital o característica deseada. La escalabilidad de la arquitectura headless admite elementos de tecnología preparados para el futuro que permiten a los equipos digitales integrar las mejores herramientas de su clase hoy con la flexibilidad de aumentar dicho conjunto de tecnología según sea necesario en el futuro.

Las arquitecturas monolíticas, por otro lado, requieren que los equipos esperen a que su único proveedor de soluciones desarrolle nuevas capacidades o cree soluciones alternativas para la funcionalidad deseada, lo que genera una deuda tecnológica innecesaria.

#### **4. Se adapta a diversos lenguajes y marcos**

Con libertad en las herramientas, las arquitecturas Headless permiten a los equipos de desarrollo diversificar los lenguajes y marcos que utilizan. Este es un gran salto con respecto a los monolitos, como un CMS tradicional, donde el proveedor de software decide qué lenguajes y marcos deben utilizar los desarrolladores. Las soluciones de software monolíticas ofrecen una variedad de capacidades a expensas de la funcionalidad y la flexibilidad.

Con más libertad y flexibilidad en lenguajes y marcos, los miembros del equipo que trabajan con arquitecturas Headless se vuelven más productivos e incluso pueden experimentar una mayor satisfacción con su trabajo, una ventaja para la retención de los administradores y la imagen única para los usuarios del sistema.

El lenguaje y el marco de elección se pueden adaptar a un proyecto determinado, el ecosistema y las habilidades del desarrollador. Los desarrolladores de frontend pueden mezclar y combinar el uso de marcos de frontend populares, como React, Angular, Vue.js, Svelte o Preact, mientras que los desarrolladores de backend pueden recurrir a marcos propios de backend como JavaScript, Python, PHP, Java o Ruby.

#### **5. Acelera la productividad mediante flujos de trabajo ágiles**

Las arquitecturas Headless reducen el tiempo de comercialización en más de un sentido. Además de admitir la diversidad en marcos y lenguajes, un enfoque sin cabeza admite flujos de trabajo ágiles en lugar de los tradicionales flujos de trabajo en cascada comunes en las arquitecturas monolíticas.

Cada microservicio dentro de una arquitectura Headless suele ser administrado por equipos pequeños y dedicados que pueden trabajar de forma autónoma. Esto permite a los equipos iterar e implementar nuevas funciones según sea necesario sin las interdependencias comunes a la arquitectura monolítica estrechamente acoplada.

Con los flujos de trabajo digitales tradicionales, las etapas de un proyecto se completan de forma independiente. Los creativos pueden generar y editar contenido mientras los desarrolladores modifican el código sin que un esfuerzo detenga (o dependa) del otro.

Con procesos de desarrollo, publicación y optimización tan ágiles, las marcas pueden actuar en función de los comentarios de los usuarios finales más rápidamente y crear experiencias mejoradas para los clientes en tiempo real.

#### **6. Admite contenido estructurado y modular**

La arquitectura Headless no solo mejora los flujos de trabajo y las capacidades de los desarrolladores, sino que también mejora los de los editores, los especialistas en marketing y cualquier otra persona que cree, actualice y administre contenido al admitir contenido modularizado.

En lugar de tener contenido y medios relacionados vinculados a un diseño específico, el contenido modular o estructurado, divide los elementos en pequeños bloques de construcción que se pueden editar y reutilizar fácilmente en otros canales o para respaldar nuevos contenidos y experiencias.

Además de eliminar el copiado y pegado excesivo (y la posibilidad de que surjan errores con dichas acciones manuales), el contenido estructurado es más accesible para los rastreadores web y se presta a la personalización y traducción, mejorando el SEO y los mercados en los que se infiltran las empresas que adoptan esta estructura de contenido.

Para implementar contenido estructurado, se debe combinar una arquitectura Headless con un sistema de gestión de contenido Headless CMS.

## 7. Admite el comercio modularizado

Si bien las arquitecturas Headless pueden proporcionar un marco para las necesidades digitales de las marcas en todas las industrias, se están utilizando en el comercio minorista hasta tal punto que el comercio electrónico Headless, también llamado comercio Headless o comercio modular, es ahora un software y un marco en sí mismo.

No existen diferencias significativas entre el comercio electrónico sin cabeza y la arquitectura sin cabeza, excepto que la primera es más específica e incluye una plataforma de comercio electrónico como Adobe Commerce (Magento), Shopify, BigCommerce, WooCommerce y Squarespace, entre otras. La popularidad del comercio electrónico Headless se puede ver en el volumen de minoristas y mercados líderes que utilizan este concepto, que incluyen a Amazon, Costa Coffee, Etsy, Nike, Notion, Shiseido, Target, West Elm entre otros.

## ANEXO 1: Arquitectura MACH

La arquitectura MACH es el conjunto de principios tecnológicos que está detrás de las mejores plataformas tecnológicas actuales. El acrónimo significa basado en **Microservicios**, primero **API-First**, nativo **Cloud** y **Headless**:

- **Microservicios:** piezas individuales de funcionalidad empresarial que se desarrollan, implementan y gestionan de forma independiente.
- **API-First:** toda la funcionalidad se expone a través de API, lo que permite unir dos o más aplicaciones o servicios.
- **SaaS nativo en Cloud:** software como servicio que aprovecha todas las capacidades de la nube y se desarrolla directamente en dicho entorno, más allá del almacenamiento y el alojamiento, incluido el escalamiento elástico de recursos de alta disponibilidad. La funcionalidad se actualiza automáticamente, eliminando la necesidad de gestión de actualizaciones. No es meramente una adaptación desde on-premise a nube.
- **Headless:** la experiencia del usuario de frontend está completamente desacoplada de la lógica de backend, lo que permite una total libertad de diseño para crear la interfaz de usuario y para conectarse a otros canales y dispositivos, es decir, aplicaciones existentes, IoT, A/R, máquinas expendedoras, sensores, etc.

MACH ganó rápidamente popularidad por la forma en que ayuda a las empresas. Las tecnologías MACH respaldan una empresa modular, lo que significa que cada componente es conectable, escalable, reemplazable y puede mejorarse continuamente. La arquitectura MACH brinda a las empresas la libertad de elegir entre las mejores herramientas del mercado y mantener una estructura que facilite agregar, reemplazar o eliminar esas herramientas en el futuro.

### ¿Cuáles son los beneficios de la arquitectura MACH?

Pasar de una tecnología monolítica o basada en suite a una arquitectura MACH brinda la libertad de elegir entre las mejores herramientas del mercado actual y proporciona una estructura que facilita agregar, reemplazar o eliminar tecnologías en el futuro. En pocas palabras, la arquitectura MACH permite romper los ciclos de cambio de plataforma o actualización finalmente.

Además de evitar otro caso de estar esposado por tecnología obsoleta y la incapacidad de innovar y evolucionar, hay cuatro beneficios adicionales de la arquitectura MACH:

- **Velocidad mejorada con menor riesgo:** con esta arquitectura ágil, se obtiene una ruta sensiblemente más rápida hacia el MVP (producto mínimo viable) y, por lo tanto, hacia el lanzamiento. Para el marketing, comunicación, los equipos de ventas, nuevos negocios, desarrollo de procesos y en general cualquier departamento de la empresa, se puede implementar rápidamente prototipos que ayuden a probar conceptos clave antes de invertir en implementaciones a gran escala. La creación rápida de prototipos también ayuda a evitar los tediosos procesos de RFP, ahorrando recursos, tiempo e inversión.
- **Ejecuta la mejor estrategia existente en el mercado:** la arquitectura MACH permite aprovechar la mejor tecnología disponible. Ya no es necesario conformarse con complementos que no son los mejores que vienen con los paquetes de software de los grandes proveedores. Debido a su naturaleza modular, MACH también puede ayudar a preservar la funcionalidad existente en la que ha invertido y con la que está satisfecho.
- **Sin actualizaciones:** ya no es necesario ocuparse por actualizaciones disruptivas que parecen proyectos de cambio de plataforma en sí mismas. Con la arquitectura MACH, todos los lanzamientos son automáticos y sin interrupciones.
- **Personalización, innovación y ejecución perfecta:** el mercado valora la importancia de poder realizar cambios rápidamente a medida que cambian las necesidades. Priorizar la innovación significa priorizar la iteración. La capacidad de cambiar e innovar constantemente en la experiencia del cliente es un pilar clave de la arquitectura MACH.

### Cómo evaluar la arquitectura MACH

A medida que el término “arquitectura MACH” gana fuerza en el mercado, puede resultar difícil determinar qué herramientas son verdaderamente modernas y alineadas con ese modelo. Además, adoptar la arquitectura MACH puede ser tan simple como reemplazar el frontend del elemento de administración de contenido o comercio existente con un marco Headless o tan complejo como adoptar la arquitectura MACH por completo como una nueva estrategia de tecnología empresarial.

De cualquier manera, comprender qué buscar es importante para asegurarse de que esta decisión ayude a romper el ciclo de cambio de plataforma.

Factores a considerar al evaluar una arquitectura MACH potencial, es decir, un proveedor ya sea de un Microservicio o una Suite:

**Microservicios:**

1. ¿Cuenta con verdaderos microservicios? Puede desarrollar, actualizar y escalar servicios de plataforma independientes y específicos con un impacto y riesgo mínimos para la implementación general.
2. ¿Se pueden ejecutar implementaciones por fases? Los prototipos ligeros y los MVP que ofrecen un enfoque iterativo para la implementación se están volviendo comunes. Se debe poder agregar y reemplazar servicios gradualmente con el tiempo y con capacidad, finalmente, de marcha atrás inmediata sin afectar a los usuarios.
3. ¿Apoya un enfoque de mejor calidad? Agrega o reemplaza sistemas y servicios de forma independiente. Las tecnologías elegidas, tanto inversiones en tecnología existentes como requisitos emergentes, se realizan a medida que cambian las necesidades.

**API-First:**

4. ¿Está construido con API desde cero o ha adoptado una estrategia de API integrada? Por definición, los sistemas API-First son completamente abiertos. Las tecnologías creadas con API desde cero brindan un control total de toda la plataforma.
5. ¿Se puede acceder a documentación de calidad? Un repositorio de conocimientos completo, con capacidad de búsqueda y de autoayuda que ofrece documentación técnica de calidad que demuestra las capacidades y recursos de la plataforma para ayudar en el desarrollo de soluciones es la piedra angular de las plataformas API-First.
6. ¿Cómo se completan las integraciones? Las API y los webhooks ayudan a facilitar la interacción con otros sistemas. A menudo, hay disponibles conectores o integraciones prediseñadas para numerosas tecnologías comunes.

*Nota: Un webhook es un mensaje automatizado que se envía a una aplicación externa cuando ocurre un evento. A nivel más técnico, un webhook es una devolución de llamada HTTP o una petición HTTP POST generada por la notificación de un evento. Esta petición HTTP se registra como URL de webhook y se utiliza para almacenar datos en formato JSON. En ese momento, puedes gestionar el webhook para que disponga de la información como se deseé.*

**Nativo Cloud:**

7. ¿Ofrece escalabilidad ilimitada? La infraestructura de la nube proporciona capacidades de escalamiento avanzadas para satisfacer los crecientes niveles de demanda en el tiempo.
8. ¿El software se entrega como servicio (SaaS)? No hay ninguna opción local disponible. El proveedor proporciona un Acuerdo de Nivel de Servicio (SLA) específico.
9. ¿Las actualizaciones y mejoras se realizan mediante entrega continua sin cambios importantes? Las actualizaciones en toda la plataforma se aplican automáticamente sin esfuerzo del cliente, tiempo de inactividad, costes de licencia u otras tarifas. Las mejoras de funcionalidad e infraestructura aparecen continuamente y son comunicadas de forma adecuada al administrador.

**Headless:**

10. ¿Se puede “ver” la interfaz administrativa o del comprador sin desarrollo? Las verdaderas plataformas Headless vienen sin ninguna interfaz frontal existente. Es posible que encuentre componentes, interfaces o plantillas prediseñadas que pueda aprovechar para llegar más rápido a la interfaz de usuario que deseé.

11. ¿Se puede desarrollar e implementar la experiencia del usuario de forma libre y flexible? Debido a que la tecnología Headless no dicta una experiencia de usuario frontend, el desarrollo de la interfaz de usuario, así como la implementación, dependen completamente del desarrollador. Las soluciones pueden utilizar cualquier elemento de tecnología, lenguaje de programación y marco de interfaz de usuario.

Las tecnologías creadas con la arquitectura MACH proporcionan los componentes básicos necesarios no solo para transformar digitalmente una empresa, sino también para evolucionar continuamente hacia el futuro. La arquitectura MACH es el modelo tecnológico adecuado para cualquier entidad que busca preparar su negocio para el futuro.