



Ecosistema no code 2026.04

Game Engine, CMS Suite, ERP, Content Core, ia Skills.

La primera plataforma no-code para crear, publicar y escalar experiencias 3D de calidad comercial

Sistema integrado de herramientas que elimina la dependencia de tecnicos ySaaS externos,
convirtiendo a partners multidisciplinarios en altamente productivos

1. Kaizen, la Plataforma, TMOG y los Partners

KAIZEN-KINTSUGI · El colectivo

El Colectivo que construye, opera y gobierna el sistema: GameDevs, artistas 2D y 3D, marketers, publicistas, redactores, copys editoriales, software devs... Kaizen define los estandares, valida la calidad y gestiona el roadmap. Los Partners son los miembros del colectivo que lo ejecutan.

LA PLATAFORMA · El ecosistema de herramientas

Conjunto de herramientas que Kaizen construye y evoluciona para que todos los perfiles de Partner sean altamente productivos sin depender de técnicos ni de SaaS externos. Incluye aplicaciones propietarias (Game Engine, ERP) e integraciones open source MIT (CMS Suite) integradas en una interfaz y flujo.

TMOG · The Master of Gaming

El producto B2C y la fuente de monetización del colectivo. Una experiencia 3D navegable con 660+ mundos temáticos (cuenta la historia de los videojuegos meses a mes desde 1971. El usuario web que entra en TMOG navega, interactúa, juega y es gamificado dentro del entorno 3D diseñado por el Game Engine.

LOS PARTNERS · Los creadores del colectivo

Cada Partner tiene acceso al sistema integral y a herramientas especializadas: los GameDevs construyen en el Game Engine, los marketers y copys trabajan desde el CMS, los artistas usan ambas según el formato. Todos están gestionados y gamificados internamente por el ERP.

Kaizen gestiona el Colectivo de Partners que usan la Plataforma según su perfil para crear contenido.

Ese contenido alimenta TMOG, que genera los ingresos que sostienen a Kaizen. Cada iteración mejora los estándares que benefician a todos.

2. Perfiles: quien usa que y para que

El ecosistema está diseñado para que cada perfil tenga sus herramientas optimas sin necesitar al resto. Un marketer no bloquea a un GameDev. Un redactor no depende de un dev para publicar. Cada uno trabaja en su entorno con autonomia total, y los resultados se integran cuando tiene sentido.

Perfiles internos del colectivo (Partners)

PERFIL	HERRAMIENTA PRINCIPAL	QUE CREA	COMO IMPACTA AL SISTEMA
A. GameDev	Game Engine	Mundos 3D, objetos interactivos, mecanicas, NPCs, misiones, experiencias embebidas, logica de gameplay	Produce el contenido central de TMOG. Sus creaciones son el producto que el suscriptor paga.
B. Artista 2D/3D	Game Engine + CMS	Assets 3D, texturas, animaciones, ilustraciones, mangas animados con timeline del CMS, material visual	Alimenta la biblioteca de assets reutilizables en prefabs. Eleva la calidad visual del producto.
C. Marketer / Publicista	CMS Suite	Landings gamificadas, press kits, webs satelite, campanas, contenido para redes, ads, SEO	Genera trafico organico hacia TMOG. Captura comunidad antes, durante y después del lanzamiento del motor.
D. Copy / Editorial	CMS Suite	Articulos SEO, reportajes, analisis historicos, wikis estructuradas, fichas, documentacion	Posiciona el proyecto en buscadores. Genera las wikis que se convierten en skills de IA.
E. Diseñador UX	CMS Suite (Canva + Miro)	Infografias, banners, diagramas de flujo, mapas narrativos, layouts de nivel, diagramas de arquitectura	Produce assets visuales directamente reutilizables en el Game Engine. Facilita diseno narrativo colaborativo.
F. Devs	Todo el ecosistema	Construye y mantiene el Game Engine, CMS, ERP y sus integraciones. No crea contenido: habilita la creacion.	Cada mejora tecnica multiplica la productividad de todos los perfiles anteriores.

PRINCIPIO CLAVE

Ningun perfil bloquea a otro. El marketer publica su landing sin esperar al GameDev. El redactor crea su wiki sin tocar el Game Engine. El artista exporta su ilustracion al sistema sin pedir ayuda al dev. Todos en el mismo ecosistema, sin dependencias cruzadas, sin cuotas de SaaS externos.

Perfiles de consumidor de TMOG

PERFIL	ACCESO EN TMOG	GAMIFICACION (ERP)	FUENTE DE ORIGEN
Visitante	Llega a la web (vía SEO, prensa, redes) pero no puede acceder a nada sin registrarse. Ve la propuesta de valor y se le empuja a registrarse	No deja datos,siendo el objetivo capturar su mail para convertirlo en lead. Sin progreso guardado.	Trailers y publicaciones en redes sociales y foros Búsqueda orgánica, campanas del marketer.

Registrado	Navega por una zona gratuita acotada, un vertical slice del producto real. Ve mundos pero solo thumbnails y previews desde fuera. No accede a experiencias embebidas. Es una demo envuelta en narrativa, diseñada para convertir.	Misiones de captación: 'Regístrate para desbloquear...'. Misiones básicas, GameCoins limitados, progreso guardado entre sesiones.	Convertido desde CTA del CMS, link en artículo, press kit, campana de marketer.
Suscriptor	Acceso completo. Navega libremente los 660 mundos siguiendo una narrativa. Interactúa con todo, minijuegos, documentales, análisis, coleccionables.	Gamificación completa gestionada por ERP: misiones avanzadas, recompensas, Game Room personalizable, ranking, economía virtual, cobros y facturación.	Principal fuente de ingresos. Sus patrones de consumo condicionan que mundos se producen a continuación.

TMOG ES AUTOSUFICIENTE

El jugador que entra en TMOG navega un videojuego 3D en el navegador. Toda la experiencia, la progresión y la gamificación ocurren en el Game Engine y son gestionadas por el ERP. El CMS no interviene dentro del juego más que para contenidos embebidos. El CMS opera antes, fuera y en paralelo: calienta antes del lanzamiento, amplifica durante y expande después. Pero no es una dependencia del producto, sino un amplificador.

3. La Plataforma: un ecosistema integrado, sin dependencias externas

Es un sistema que proporciona diferentes herramientas de edición para que cada Partner pueda elegir la que más le convenga según el contenido que va a crear. La intención es un ecosistema integrado altamente productivo, sin depender de terceros, sin cuotas en 20 SaaS diferentes, sin saltar entre herramientas distintas, sin esperar a un técnico para publicar o crear. Todo en una única interfaz, con colaboración en tiempo real y diseño visual accesible a perfiles no-code.

La barrera de entrada es prácticamente cero: un marketer puede incorporarse al colectivo y publicar su primera landing el mismo día. Un redactor puede crear y publicar su primera wiki sin saber programar. Un artista puede subir sus ilustraciones y animarlas en el CMS o hacerlas disponibles para el Game Engine sin exportaciones manuales. Eso permite captar decenas o cientos de Partners (y a sus comunidades) porque el coste de incorporarlos es mínimo y el valor que aportan es inmediato.

Herramientas propietarias	Implementaciones open source MIT	Infraestructura comun
Game Engine y ERP: desarrollo propio para resolver lo que ninguna herramienta de mercado resuelve (navegación 3D no code) Control total sobre roadmap, integración y evolución.	CMS Suite construida sobre librerias MIT: Tiptap, dnd-kit, Fabric.js, React Flow. Funcionalidades integradas en semanas, sin licencias y con total control del codigo.	Autenticacion y permisos, sistema de assets con versionado, modelo de documento JSON compartido, API interna y colaboracion en tiempo real via Phoenix Channels. Todo esto es invisible para el Partner.

SUSTITUYE A TODA UNA PILA DE SAAS

Un Partner del colectivo no necesita Webflow (Page Builder), Canva (Visual Editor), Miro (Diagram Editor), Bubble (App Builder), Notion (wikis), Figma (diseño colaborativo), Trello (gestion de tareas), Unity o Unreal (Desarrollo de videojuegos) ni herramientas de automatización (n8n, make).

Todo esta integrado, conectado y disponible en el ecosistema (y sin cuotas). Cada herramienta comparte el mismo sistema de assets, permisos y historial de versiones.

4. Arquitectura: las capas del ecosistema

El ecosistema no es una jerarquía (donde una pieza manda sobre las otras), es una red donde cada capa puede operar de forma independiente y, al mismo tiempo, potenciarse mutuamente. Todas comparten la misma infraestructura y pueden integrar o consumir el contenido de las demás cuando tiene sentido.

El Game Engine puede funcionar solo y TMOG puede ser un producto completo sin necesitar el CMS el cual también puede publicar contenido completamente independiente de TMOG. El ERP gestiona y gamifica cualquiera de los entornos. Esta independencia es una fortaleza (no se ponen todos los huevos en el mismo cesto), y cada capa expande las posibilidades del conjunto.

01 GAME ENGINE

El núcleo del producto. Crea y ejecuta los mundos 3D de TMOG. Runtime optimizado para distribución masiva en navegador.

Motor 3D propio construido para ejecutarse en navegador sin instalación. Carga progresiva por zonas, streaming de contenido y adaptación dinámica de calidad según dispositivo. Diseñado para que un GameDev no-code pueda construir mundos completos sin scripting manual.

Lo que permite crear al GameDev

Mundos 3D navegables con editor visual estilo drag & drop y previsualización en tiempo real
Lógica interactiva sin código: panel visual: EVENTO → CONDICION → ACCION
Máquinas de estado para NPCs, triggers espaciales, hitboxes visuales
Secuencias cinematográficas con el sistema de timeline: animaciones, audio y cámara coordinados
Experiencias embebidas: minijuegos 2D, documentales, análisis activados desde objetos del mundo
Prefabs reutilizables: objetos completos con comportamiento y lógica guardados en la librería
Publicación con URL propia, optimizada para navegador, sin instalación ni configuración técnica

Capacidades técnicas del runtime

Editor de materiales y texturas integrado: color, metalicidad, emisión, sin herramientas externas
Editor de shapes y props: geometrias que se convierten en objetos del mundo directamente
Sistema de cámara dinámico: primera persona, tercera, cámaras fijas por zona, transiciones automáticas
Sistema de persistencia: progreso, puntuaciones, estado del mundo y coleccionables entre sesiones
Colaboración determinista en tiempo real: varios creadores editando el mismo mundo simultáneamente
Optimización para navegador: niveles de detalle según distancia, gestión activa de memoria

02 CMS SUITE

Suite de edición visual no-code para marketers, copys, publicistas y artistas. Independiente del Game Engine. Lo potencia cuando conviene.

El CMS: es una suite completa de herramientas de creacion visual que integra lo que normalmente requeriria 5-10 SaaS diferentes.

Ideal para: marketers, copys, publicistas y diseñadores que no utilizan el Game Engine o que requieren contenidos menos pesados o de menor rigor técnico.

El CMS puede nutrir al Game Engine con contenido embebido cuando tiene sentido, pero el Game Engine funciona perfectamente sin el.

HERRAMIENTA	QUE ES	CASOS DE USO
Page Builder	Constructor visual de paginas por bloques con drag & drop. Basado en Tiptap + dnd-kit + Phoenix.	Articulos SEO, landings gamificadas, fichas de videojuegos, paginas satellite, press kits, documentacion tecnica, wikis-Skills. Publicacion con URL propia y SEO configurable.
Visual Design Editor	Editor grafico tipo Canva integrado. Motor Fabric.js. Diseno vectorial, tipografia, composicion.	Infografias, banners, material de marketing, slides, portadas de articulos, para redes. Los diseños se convierten en assets reutilizables directamente en prefabs.
Diagram Editor	Canvas de diagramas colaborativo en tiempo real. Motor React Flow + Phoenix Channels.	Mapas narrativos, flujos de nivel, arboles de decisión, diagramas de arquitectura, documentación viva. Los GameDevs diseñan la estructura de un mundo antes de construirlo. Varios usuarios editando simultaneamente.
Interaction Engine	Motor de logica visual no-code conceptualmente compartido con el Game Engine. EVENTO → CONDICION → ACCION. Cada sistema usa su propio runtime.	Paginas web interactivas, articulos dinamicos que desbloquean contenido, funnels gamificados, quizzes con progresion. Unifica el modelo mental del Partner.
App Builder	Constructor de mini-apps no-code tipo Bubble/Retool. Bloques, variables, logica visual.	Calculadoras, simuladores, quizzes, dashboards, formularios avanzados, herramientas internas, automatizaciones de proceso tipo n8n. Todo sin codigo.
Timeline de animacion	Herramienta de secuenciacion visual para animaciones 2D, mangas animados y contenido narrativo con movimiento.	Los artistas pueden crear contenido animado complejo sin herramientas externas: Animaciones 2D para campañas de marketing, mangas animados, tutoriales animados, presentaciones cinemáticas.

EL CMS COMO MULTIPLICADOR

El Game Engine por si mismo puede producir TMOG El CMS aporta multiplicacion: el crecimiento organico del contenido del GameDev (SEO, comunidad, prensa) mas la amplificacion del marketer desde el CMS. Ademas, abre un universo de posibilidades de contenido que seria mas caro o lento producir en el Game Engine: articulos de prensa, landings comerciales, contenido para redes, press kits. Todo eso se produce fuera y puede o no conectarse con TMOG segun convenga.

03 ERP

Cerebro operativo transversal. Gestiona a los Partners internamente y gamifica la experiencia del jugador en TMOG.

Es la capa que conecta a los Partners del Colectivo con el producto, y a los suscriptores con la experiencia. Diferentes contextos, un mismo motor de logica.

Gestion interna del colectivo Gestion de proyectos: colaborativo en tiempo real, hitos, tareas, dependencias Roles y permisos ACL granular: quien puede crear, editar o publicar que Métricas de productividad individual y de equipo Gamificación interna: misiones de producción. Automatizaciones no-code: triggers, condiciones y acciones para flujos operativos Puede priorizar que mundos se producen según demanda y métricas de consumo	Experiencia del jugador en TMOG Toda la gamificación del jugador ocurre aqui: misiones, logros, desbloqueos, Game Room GameCoins y economia virtual: obtencion, uso, recompensas Progreso persistente: el estado del jugador se guarda y recupera entre sesiones Control de acceso por tipo de suscripción: que puede ver y hacer cada perfil Métricas de comportamiento: zonas de mayor interacción, contenido mas consumido, retención	Monetizacion y negocio Facturación recurrente automatizada: Stripe-Paypal, Gestion de planes de membresia: creacion, modificacion, cupones, migraciones Campanas completas: segmentacion, A/B testing, tracking y reporting de ROI Sincronización en tiempo real con el CMS vía Phoenix Channels Dashboard de métricas: conversión, retención, ingresos, comportamiento
---	--	---

04 CONTENT CORE

Fuente unica de verdad. Centraliza datos de juegos, plataformas, historia y activos de marketing para que todos los sistemas los consuman sin duplicar.

El Content Core alimenta a todos los sistemas. Es la capa que garantiza que el mismo dato no existe dos veces y que cada sistema lo interpreta según su contexto.

Contenidos de juego e historia 300.000+ videojuegos: titulo, fecha, plataforma, genero, descripción, relaciones entre titulos 660+ meses historicos: contexto cultural, lanzamientos clave, eventos del sector Hardware y plataformas: fichas tecnicas, fabricantes, generaciones Las fichas de los 300.000 videojuegos tienen mas sentido en el Content Core que en el Game Engine: son datos estructurados (formulario de ficha) que el GameDev simplemente referencia al construir su mundo 3D, sin tener que reintroducirlos. El CMS puede mostrarlos como articulo SEO; el ERP puede crear misiones vinculadas a ellos.	Activos de marketing y produccion Disenos visuales del CMS: infografias, banners, ilustraciones disponibles para todos los sistemas Diagramas de flujo y mapas narrativos reutilizables entre proyectos Landings y press kits versionados Libreria centralizada de assets 3D, audio y UI compartida entre CMS y Game Engine sin exportaciones Versionado y actualizacion global: si se mejora un asset, el cambio se propaga a todas sus instancias
---	---

El Content Core responde a la pregunta: donde deberian estar las fichas de los 300.000 videojuegos. La respuesta es: en el Content Core, como datos estructurados independientes. El Game Engine los interpreta como objetos 3D interactivos. El CMS los muestra como articulos o fichas SEO. El ERP los usa como nodos de mision o progreso. El GameDev simplemente referencia la ficha al colocar el objeto en el mundo, sin introducir datos manualmente.

05 IA CON SKILLS

Amplificador del sistema. Convierte el conocimiento estructurado del colectivo en capacidades ejecutables. El cerebro digital compartido.

La IA amplifica el sistema. Sin wikis estructuradas, sin prefabs validados y sin Content Core poblado, no hay automatización real. La calidad de las wikis determina directamente la capacidad del sistema. La IA es tan buena como el conocimiento que la alimenta.

Wiki → Skill: el mecanismo central

Una wiki ejecutable tiene una plantilla fija de 7 campos: que es, para que sirve (inputs y outputs tipados), pasos secuenciales, reglas verificables, límites y ejemplo completo. Cuando una wiki cumple ese estándar, el sistema la convierte automáticamente en una skill ejecutable sin intervención técnica.

Cada skill puede ser invocada por otras skills: el sistema crece de forma multiplicativa. A más wikis de calidad, más tareas puede ejecutar el sistema de forma autónoma. Las wikis son otro activo que ningún competidor puede replicar rápidamente: solo se construyen con uso real

El cerebro digital compartido del colectivo

Cada Partner que crea una wiki contribuye al cerebro digital del colectivo. El conocimiento acumulado sobre cómo producir un mundo de 1987, cómo escribir el análisis de un juego de NES o cómo estructurar una landing de captación, queda disponible para todos los perfiles y para los agentes futuros.

En Beta: asistencia en creación, validación de wikis, generación de assets básicos
En Gamma-Delta: pipeline wiki → skill activo, generación automática de mundos.
En Omega-Zeta: agentes autónomos que coordinan producción completa a partir de un objetivo

5. Por qué no lanzar primero solo el Game Engine

TMOG puede funcionar sin el CMS. El Game Engine puede producir un producto completo y monetizable con el ERP. Sin embargo, todo fue ideado mucho antes para el CMS

LA RESPUESTA DIRECTA

Lanzar solo el Game Engine es lanzar un producto funcional pero limitado en su crecimiento, genera la experiencia y la monetización. El CMS multiplica el alcance, diversifica los contenidos y acelera la producción. Cada día sin CMS activo es un día sin crecimiento orgánico, sin base de wikis para la IA y sin que los perfiles no-GameDev (marketers, creadores editoriales, mangakas...) puedan contribuir. Y esos días no se recuperan.

1. El crecimiento organico no empieza el día del lanzamiento (empieza meses antes)

El SEO tarda en posicionarse. Si el CMS no está activo antes del lanzamiento del motor, TMOG llega al mercado sin tráfico orgánico extra.

Con el CMS activo semanas antes, el proyecto llega con linkbuilding en medios e influencers, posicionamiento real, comunidad construida y early adopters esperando.

2. Los perfiles no-GameDev no tienen herramientas sin el CMS

Los marketers, copys, publicistas y diseñadores del colectivo no pueden contribuir sin el CMS y el 100% de la producción de contenido recaería en los GameDevs, que tienen que hacer cosas que no son su core: escribir artículos, crear landings, generar material de prensa. El CMS libera a los GameDevs para que se enfoquen en construir mundos.

3. La base de wikis que escala la IA no se puede construir después

Las wikis son el origen de las skills. Sin un entorno adecuado para construirlas desde el inicio, el conocimiento queda en formatos inconsistentes o simplemente no se genera. Cada semana sin wikis activas es capacidad autónoma futura que no se acumula. No se puede añadir el CMS 6 meses después y tener la misma base que si se hubiera empezado desde el día uno.

4. La producción es más rápida con el CMS activo desde el inicio

El CMS no compite con el Game Engine: lo descarga. Los diagramas de nivel, las wikis de referencia, los assets visuales, el contenido editorial de apoyo..., todo eso se produce mejor y más rápido fuera del motor. Sin el CMS, los GameDevs tienen que hacerlo dentro del Game Engine o pedir ayuda a devs para cada pieza de comunicación. Con el CMS, cada perfil trabaja en su entorno óptimo.

5. El CMS se construye en paralelo, no retrasa el motor

Implementado con librerías MIT open source por un especialista en react y apoyo puntual del Backend. Apenas consume recursos del equipo del Game Engine.

LA DECISION

Decidir si el proyecto crece desde el primer día con todos sus perfiles activos o si arranca con solo una parte del equipo contribuyendo. El Game Engine genera la experiencia y la monetización. El CMS hace que esa experiencia llegue a más gente, con más velocidad y con más herramientas.

6. Que aporta el CMS Suite: casos de uso concretos

El CMS Suite y el Game Engine son herramientas independientes que pueden funcionar por separado y que se potencian mutuamente cuando conviene.

Adquisicion y marketing: trafico hacia TMOG y mas alla

Press kits dinamicos El marketer construye un press kit que consume datos reales del sistema: metricas actualizadas, capturas del mundo 3D, assets generados por el artista. Se distribuye a medios especializados, blogs de retro y canales de YouTube. Genera linkbuilding hacia TMOG sin trabajo técnico.	Webs satelite por era historica Una landing optimizada con una temática específica posiciona en búsqueda orgánica y dirige al mundo 3D relacionado. El usuario no se queda en la web satélite, la cual existe para capturar trafico que llega a TMOG con intención específica.	Campanas gamificadas Un quiz sobre historia de los videojuegos publicado en el CMS capta al usuario, registra su progreso en el ERP y ese progreso desbloquea contenido dentro de TMOG. La captación se convierte en una extensión de la experiencia del juego.
--	--	---

Produccion: el CMS descarga al Game Engine y acelera a todos los perfiles

Diseño y narrativa previos al 3D El Diagram Editor permite al GameDev diseñar el mapa del nivel, los flujos narrativos y la arquitectura de interacciones antes de abrir el Game Engine. Detecta errores conceptuales antes de construir. Reduce retrabajo, acelera entregas y en colaboración.	Assets reutilizables sin fricción El artista diseña en el Visual Editor y el resultado entra directamente en la librería de assets del ecosistema. El GameDev lo usa en el motor sin exportar ni convertir. El diseñador no necesita saber nada del Game Engine para contribuir al producto.	Wikis como base de producción El redactor crea wikis ejecutables que se convierten en skills de IA. No son documentación: son capacidad operativa futura. Un proceso bien definido en una wiki permite al sistema ejecutarlo automáticamente. A más wikis, el colectivo es más eficiente.
---	--	---

Contenido editorial: lo que el Game Engine no procesa bien

El Game Engine no está tan optimizado para texto largo, artículos estructurados o contenido SEO. El CMS cubre exactamente ese espacio:

Artículos de análisis histórico de videojuegos: escritura larga, referencias, SEO Reportajes sobre developers, estudios y lanzamientos históricos Fichas técnicas de hardware y plataformas para posicionamiento en búsqueda Contenido para redes sociales: hilos, posts, carruseles, virales	Documentación técnica del proyecto para nuevos Partners Guías de estilo y estándares de producción disponibles para todo el colectivo Newsletters y comunicación con la comunidad sin herramientas externas Todo puede embeber contenido del Game Engine para enriquecer la experiencia
--	--

El crecimiento de TMOG tiene dos vectores: el contenido del Game Engine (experiencias 3D de alta calidad que retienen al suscriptor) y la amplificación del CMS (SEO, prensa, comunidad, redes que traen nuevos usuarios). Solos, cada uno crece de forma lineal. Juntos, se multiplican. El crecimiento orgánico generado por el contenido del GameDev mas la amplificación del marketer desde el CMS crean un efecto compuesto que ninguno de los dos puede alcanzar por separado.

7. Como se conectan las piezas del ecosistema

El potencial de la plataforma no está en una única herramienta, sino en la capacidad de conectarlas todas. Cada conexión elimina una capa de trabajo manual y abre posibilidades que no existirían de forma aislada:

7.1 El Interaction Engine: el lenguaje comun de comportamiento

No es una pieza del sistema: es la logica que lo conecta y lo hace operar como uno solo. El Game Engine y el CMS usan el mismo modelo conceptual de definición de comportamiento: EVENTO → CONDICION → ACCION. Cada sistema usa su propio runtime (WASM en el Game Engine, JS en el CMS). No se comparte código ni ejecución, sino la forma de definir comportamiento.

Resultado práctico: **el Partner aprende una vez y opera en todo el sistema sin cambiar de modelo mental.**

Cada mejora en el modelo de eventos impacta automaticamente en todos los entornos. La IA puede operar con lógica transversal entre sistemas.

En el Game Engine

Player entra en zona → trigger activa → NPC reacciona

Objeto interactuado → maquina de estado cambia → timeline se ejecuta

Misión completada → ERP asigna recompensa → estado del jugador actualizado

En el CMS (mismo modelo, distinto runtime)

Scroll al 50% + usuario no suscrito → mostrar CTA personalizado

Click en bloque + ha leído sección previa → desbloquear contenido adicional

Publicar artículo → generar assets + notificar al ERP + trackear distribución

7.2 Game Engine ↔ CMS: contenido embebido bidireccional

Ambas herramientas pueden enriquecerse mutuamente cuando conviene. Ninguna depende de la otra.

El Game Engine puede abrir contenido del CMS embebido: al interactuar con un objeto en el mundo 3D, se puede activar un artículo largo, un diagrama narrativo o una landing específica del CMS en pantalla completa

El CMS puede embeber experiencias del Game Engine: una landing de captación puede incluir una demo o un fragmento jugable del mundo 3D para que el visitante entienda lo que se encontrara o directamente llevarlo a una escena para fomentar la suscripción

Los assets creados en el CMS (infografías, banners, ilustraciones) entran directamente en la librería del Game Engine sin exportaciones manuales

Los mapas y diagramas del Diagram Editor sirven de referencia visual al GameDev mientras construye en el motor

7.3 Game Engine ↔ ERP: estado, progreso y monetización

El Game Engine genera eventos; el ERP los interpreta: misión completada, zona explorada, objeto desbloqueado

El ERP devuelve el estado: que puede ver y hacer el jugador según su suscripción y progreso

Persistencia completa entre sesiones: el jugador vuelve exactamente donde lo dejo

El ERP puede priorizar que mundos producir según el comportamiento de los suscriptores en el motor

7.4 CMS ↔ ERP: contenido conectado a objetivos y datos

Las misiones del ERP pueden reflejarse en contenido del CMS: una misión de onboarding puede incluir la lectura de un artículo del CMS como condición
El consumo de contenido CMS (lectura, interacción) puede generar eventos en el ERP: completar la lectura de un artículo puede asignar GameCoins o activar una misión
Los cambios estratégicos en el ERP (nuevas campanas, objetivos) se propagan al CMS sin intervención manual via Phoenix Channels
Las métricas del CMS (tráfico, conversión, engagement por articulo) se consolidan en los dashboards del ERP

7.5 Content Core → todos los sistemas

Los assets del Content Core son accesibles para todos los sistemas sin duplicar ni desincronizar
Cada videojuego es un proyecto completo con mecánicas, assets y análisis que se estructuran en el Content Core. El Game Engine convierte los contenidos de estos proyectos en el entorno 3D, el CMS en articulo o ficha SEO, el ERP en nodo de misión o coleccionable
Los contenidos no se precargan, sino que se generan según el contexto del mes que explora el jugador
Actualización global: si se mejora un prefab o un asset en el Content Core, el cambio se propaga a todas sus instancias en todos los sistemas

7.6 Content Core ↔ Prefabs del Game Engine

El Content Core alimenta directamente el sistema de prefabs: una sala con juegos de NES se crea una vez y se reutiliza en todos los mundos de esa era y posteriores.
Los prefabs pueden estar vinculados a contenido dinámico del Core: al colocar un arcade de Pac-Man, el prefab sabe automáticamente que datos mostrar cuando el jugador interactúa
Si se mejora el diseño de un prefab, el cambio se aplica a todas sus instancias en todos los mundos publicados

7.7 Asset Pipeline: libreria centralizada compartida

Una librería única de assets (3D, visuales, audio, UI) accesible desde cualquier herramienta del ecosistema
Lo que el artista crea en el CMS Visual Editor esta disponible en el Game Engine sin exportar ni convertir
Lo que el GameDev produce como asset reutilizable en el motor esta disponible en el CMS si se quiere usar en marketing
Versionado completo: cualquier versión anterior de cualquier asset es recuperable

7.8 ERP ↔ Produccion interna (Kanban gamificado)

El Partner activa su tablero desde el ERP y vincula tareas a elementos reales del sistema: mundos, misiones, prefabs, articulos
La productividad se mide automaticamente: métricas de tiempo, eficiencia y contribución sin introducción manual de datos
El ERP puede sugerir que mundos priorizar según métricas de consumo y demanda de los suscriptores
La gamificación interna (tareas, hitos, KPI's) incentiva la actividad continua

7.9 Publicacion ↔ Distribucion: el pipeline completo

Publicar un mundo en el Game Engine puede activar automaticamente la generación de contenido SEO asociado en el CMS
El marketer opera sin depender del equipo de producto: publica campanas, actualiza landings y distribuye en redes desde el CMS autónomamente
Cada experiencia publicada tiene una capa de adquisición asociada que puede activarse o no según la estrategia

7.10 Wikis / IA ↔ todos los sistemas

Las wikis del CMS alimentan skills que pueden asistir la creación en el Game Engine, el contenido en el CMS y las decisiones en el ERP

Cada iteración del Partner genera conocimiento estructurado que beneficia a todos: el sistema escala en capacidad de ejecución, no solo en contenido

La IA puede operar con lógica transversal gracias al modelo de comportamiento común del Interaction Engine

EL POTENCIAL DE LA CONEXION

La plataforma no es la suma de sus herramientas: es la red de conexiones entre ellas. Cada conexión elimina una fricción, reduce dependencias entre perfiles y abre posibilidades que no existirían de forma aislada. Un activo creado por un artista en el CMS puede aparecer en el mundo 3D del GameDev, ser usado en la campana del marketer y ser referenciado por la wiki del redactor. Todo en el mismo ecosistema, sin fricciones ni costes de integración.

8. El flujo completo del Partner no-code

PASO	HERRAMIENTA	LO QUE HACE EL PARTNER
1. Planificacion	ERP + Diagram Editor	Define el objetivo del mundo y diseña el mapa narrativo. Valida coherencia histórica con el Content Core antes de construir.
2. Gestion del proyecto	ERP · Kanban	Descompone el mundo en tareas y fases. Define hitos, prioridades y dependencias. Gestiona colaboración si hay mas Partners. Conecta planificación con ejecución real y habilita la medición de productividad.
3. Base de conocimiento	CMS · Page Builder	Crea wikis estructuradas (reutilizable y futura skill de IA). Genera contenido SEO de apoyo: articulo sobre la época del mundo, ficha de juegos clave, landing de captación. Este contenido no sustituye al del Game Engine, lo complementa y posiciona.
4. Assets y prefabs	Content Core + libreria	Selecciona assets y prefabs de época de la librería validada. Reutiliza componentes existentes (arcades, tiendas, interfaces). Crea nuevos si es necesario y los integra en la librería común. Este paso es el que hace viable el ritmo de producción.
5. Diseno visual y narrativo	CMS · Diagram + Canva	El artista crea assets visuales que entran directamente en la librería del Game Engine. El diseñador mapea la estructura narrativa del mundo en el Diagram Editor. Estos recursos guían la construcción en el motor.
6. Construccion 3D	Game Engine	Construye el mundo con el editor visual no code usando prefabs validados (estilo Mario Maker). Coloca entornos, objetos y NPCs. Previsualiza en tiempo real alternando entre crear y jugar.
7. Logica e interactividad	Interaction Engine	Define comportamientos, reglas, estados y triggers con el modelo visual común (EVENTO → CONDICION → ACCION). Sin scripting. Las reglas son reutilizables entre sistemas y pueden ser ejecutadas por la IA.
8. Integracion de contenido	Game Engine + CMS opcional	El contenido principal vive en el Game Engine construido con prefabs. El CMS interviene cuando aporta valor concreto: activar un articulo largo desde un objeto del mundo, mostrar un diagrama narrativo, embeber una demo jugable en una landing.
9. Progresion y misiones	ERP	Configura objetivos, recompensas, GameCoins y desbloques. Define como evoluciona la experiencia del jugador y que condiciones de acceso aplican según tipo de suscripción.
10. Publicacion	Game Engine + ERP	Publica el mundo con URL compartible. Control de permisos según tipo de usuario. Carga progresiva desde cualquier navegador sin instalación.
11. Promocion y distribucion	CMS + ERP	El marketer activa campanas, actualiza el press kit, publica en redes y genera contenido SEO de la época. Opera de forma completamente autónoma sin bloquear al GameDev. El crecimiento orgánico del Game Engine se multiplica con la amplificación.
12. Metricas y optimizacion	ERP	Analiza comportamiento real: zonas de mayor interaccion, contenido mas consumido, retención por época. Los datos condicionan la producción del siguiente mundo y la estrategia de marketing.
13. Estandarizacion	CMS + Content Core	Documenta lo aprendido como wiki ejecutable. Convierte soluciones en prefabs reutilizables. Cada iteración reduce el tiempo y coste del siguiente mundo. El sistema mejora solo con cada mundo publicado.

9. El valor diferencial que nadie puede igualar

La ventaja competitiva reside en la combinación: un producto sin equivalente en el mercado, construido sobre una plataforma que ningún competidor tiene, impulsado por un colectivo multidisciplinar que mejora el sistema con cada iteración.

Frente a los competidores reales de TMOG

Los competidores no son motores de videojuegos ni herramientas de creación. Son los canales con los que compartimos audiencia: YouTube, revistas y blogs especializados en historia de videojuegos. Tienen el público; nosotros ofrecemos una experiencia sin igual que no pueden replicar.

DIMENSION	YouTube / revistas / blogs	TMOG
Formato de consumo	Pasivo: ver, leer, escuchar	Activo: explorar, interactuar, jugar, descubrir
Experiencia 3D	No disponible	660+ mundos navegables por era con escenarios históricos fieles
Mecanicas de videojuego	No	Minijuegos originales, misiones, coleccionables, progresión real
Granularidad histórica	Por decadas o por titulo	Mes a mes desde octubre de 1971
Coste de replicación	Sin barrera tecnica	Motor 3D + plataforma no-code + colectivo multidisciplinar + wikis acumuladas + prefabs validados. Anos de ventaja estructural.

La ventaja acumulativa que no se puede copiar

La barrera más alta no es tecnológica: es operativa. No solo es difícil construir la tecnología o acumular los miles de contenidos e idear los mundos 3D, sino que todavía es más difícil coordinar contenido 3D, SEO, gamificación, producción editorial e IA en un sistema coherente con un colectivo multidisciplinar.

Cada mundo publicado hace que el siguiente sea más rápido y barato:

- Los prefabs de ese mundo entran en la librería y reducen el coste del siguiente
- Las wikis generadas se convierten en skills que automatizan partes del pipeline
- El ERP aprende de los patrones de consumo y orienta la producción
- La base de conocimiento del colectivo es mas robusta con cada iteración

La ventaja no se copia, se construye. Y cada día de operación amplifica la distancia. Ningún competidor puede replicar esta combinación de tecnología, procesos y acumulación de conocimiento en el tiempo que el mercado necesita. Esa es nuestra ventaja operativa.

El ciclo autosostenible

La Plataforma capta Partners con barrera de entrada cero

→ Perfiles multidisciplinares se incorporan y contribuyen en su área: GameDevs crean mundos, marketers generan tráfico, redactores posicionan en SEO, artistas elevan la calidad visual.

Los Partners producen con autonomía, sin bloquearse entre si

→ Mas mundos, más contenido editorial, mas campanas — todo en paralelo sin dependencias cruzadas.

El CMS y el SEO construyen comunidad antes del lanzamiento del motor

→ TMOG llega con tráfico orgánico, early adopters y cobertura de medios ya establecidos.

TMOG retiene con gamificación continua y experiencias que nadie mas puede ofrecer

→ Ingresos recurrentes que sostienen a Kaizen y permiten escalar la plataforma y el colectivo.

Las wikis y prefabs acumulan el conocimiento del colectivo

→ Cada mundo publicado es mas barato y rapido que el anterior. El sistema mejora solo.

10. Resumen ejecutivo

ELEMENTO	ROL EN EL ECOSISTEMA
TMOG	El producto y fuente de monetización. El jugador navega un videojuego 3D en el navegador. Gamificado 100% por el Game Engine y el ERP.
Game Engine	El núcleo tecnológico con el que se crea el producto. Runtime 3D optimizado para distribución masiva en navegador. Crea y ejecuta los mundos 3D.
CMS Suite	Suite de edicion visual no-code para perfiles no-GameDev: marketers, copys, publicistas, diseñadores. Puede operar de forma completamente independiente del Game Engine. Multiplica el alcance de TMOG y permite que el colectivo completo sea productivo sin depender de los devs.
Interaction Engine	La logica que conecta el sistema y lo hace operar como uno solo. Modelo común (EVENTO → CONDICION → ACCION) en todos los entornos. No comparte código, sino que comparte la forma de definir comportamiento. El Partner aprende una vez y opera en todo el ecosistema.
ERP	Cerebro operativo transversal. Gamificación universal para jugadores (TMOG) y Partners (colectivo). Facturación, métricas y gestión interna. Puede priorizar producción según datos reales. Conectado en tiempo real con todos los sistemas.
Content Core	Fuente única de verdad. Centraliza datos de juegos, historia, activos de marketing y producción. Un mismo dato (videojuego, prefab, asset) se interpreta de forma distinta en cada sistema sin duplicar. Las fichas de los 300.000 videojuegos viven aquí, no en el motor: el GameDev las referencia, no las reintroduce.
Prefabs + Assets	Sistema estructural transversal. Lo que hace viable producir 660 mundos con coherencia entre Partners distintos. Construcción estandarizada, reutilización acumulativa, actualización global. Conectado al Content Core.
IA con Skills	Amplificador del sistema y cerebro digital compartido del colectivo. Sin wikis y prefabs no hay automatizacion real. A mas base estructurada, mas capacidad autonoma. Evolucion a partir de asistencia en Beta hasta agentes autonomos en Omega-Zeta.

El Game Engine es el nucleo. TMOG es el producto que lo monetiza.

La Plataforma es el ecosistema que permite que un colectivo multidisciplinar produzca con autonomía, calidad y consistencia.

Interaction Engine + Content Core: una única lógica de comportamiento y un único dato estructurado que fluye por todo el sistema.

Ningún competidor puede replicar esta combinación de tecnología, procesos y acumulación de conocimiento en el tiempo que el mercado necesita.

Esa es nuestra ventaja operativa.

KΛIZEN

MEDIA CONTENT

