

CONVIERTE TU WEB EN UN MUNDO 3D

Tu mascota camina por un escenario y cada edificio abre uno de tus servicios. **Un único prompt:** lo copias, rellenas tus datos y se lo das a la IA.

SE RECORRE ANDANDO



Así quedó la de Octonove

● EL EJEMPLO

ESTO ES LO QUE VAS A CREAR



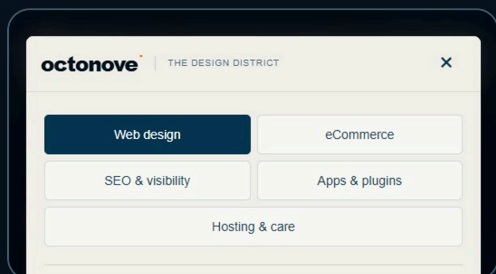
Tu mascota, caminando

Tocas el suelo y va hasta allí, sin atravesar edificios



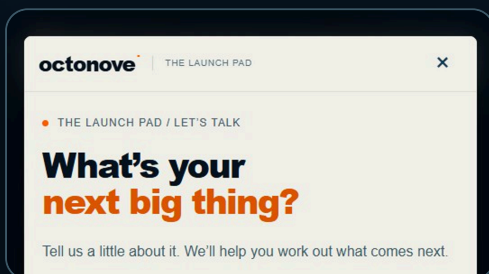
Cada edificio, un servicio

Con su etiqueta numerada encima



Paneles que venden

Titular, ventajas, llamada a la acción y trabajos



Contacto a un toque

Un formulario que abre tu WhatsApp o tu email

Portada que vende

Vista rápida sin 3D

Móvil táctil

SEO incluido

Un solo archivo

Recórrela en vivo (está en inglés): octonove-odyssey.web.app

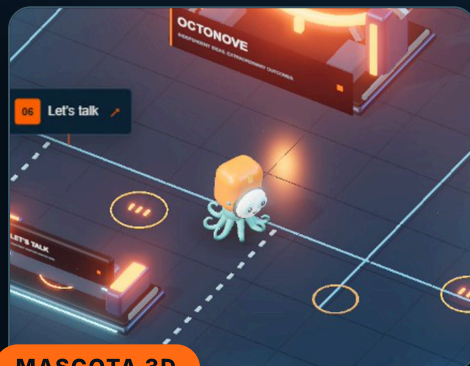
● LO MÁS IMPORTANTE

TU MASCOTA Y TUS SERVICIOS

El pulpo es el logo de Octonove convertido en 3D. La IA hace lo mismo con **tu** logo, o crea la mascota que le describas, y convierte cada uno de tus servicios en un distrito del mundo.



la IA lo convierte en...



LO HEMOS PROBADO

Dos negocios inventados. Una sola respuesta de la IA para cada uno, sin tocar una línea.



Huellas Clínica Veterinaria

6 distritos · mascota: perro salchicha con bata



Radios Taller de Bicis

5 distritos · mascota: sacada del logo

TRES PASOS, UN SOLO PROMPT

01 COPIA EL PROMPT

Está en las páginas 6 a 13. Pégallo en tus notas para rellenarlo con calma.

02 RELLENA TUS DATOS

Solo el bloque 1: lo que va entre corchetes, marcado en naranja. Cambia el corchete entero, con su ejemplo, por tu dato.

Marca: **[NOMBRE DE TU MARCA]**



Marca: Radios Taller de Bicis

03 ADJUNTA Y ENVÍA

Pégalo en Claude, ChatGPT o Gemini, añade tu logo y tu material con el botón de adjuntar, y envíalo.

 logo.png

 fotos.jpg

 servicios.pdf



Actúa como un estudio de desarrollo creativo de primer nivel...



 aquí adjuntas tu material

¿Se ha cortado a mitad? Escríbele **continúa** y seguirá justo donde lo dejó.

● JUNTO AL PROMPT

DALE CONTEXTO

El prompt le dice a la IA qué construir. Lo que adjuntas o le indicas le cuenta cómo es **tu** negocio. Cuanto más real sea, menos se inventa.

IMPRESINDIBLE



Tu logo

Mejor en PNG sin fondo. De aquí salen la mascota y los colores.



Tu web actual

Pon su dirección en el bloque 1: la IA saca de ahí servicios, textos y fotos.



Imágenes

Del local, del equipo o de tus trabajos. Irán en la galería de cada distrito.



Servicios y precios

Un documento o una lista. Cada servicio será un distrito del mundo.



Opiniones reales

3 o 4 reseñas de Google con el nombre del cliente.



Lo que necesites

Capturas de proyectos, tu carta, un catálogo o webs que te gusten.

¿Sin web?

Pon la de un negocio de tu sector como ejemplo: la IA la usará solo como guía de estructura, sin copiar nada.

EL PROMPT

Lo único que cambias es lo que va en **naranja**.

Actúa como un estudio de desarrollo creativo de primer nivel: director de arte, desarrollador experto en Three.js y juegos web, y copywriter de conversión trabajando juntos. Vas a crear la web de mi negocio: una landing page de una sola página convertida en un mundo 3D isométrico explorable, donde mi mascota camina entre distritos y cada distrito es uno de mis servicios. Referencia de estilo, no de contenido: <https://octonove-odyssey.web.app>. Tiene que parecer un pequeño videojuego premiado, pero vender como una buena landing.

1. MI NEGOCIO

- Marca: **[NOMBRE DE TU MARCA]**
- A qué me dedico: **[QUÉ HACES, EN UNA FRASE. Ej.: taller de bicicletas y tienda online en Sevilla]**
- Mi cliente ideal: **[A QUIÉN VENDES. Ej.: ciclistas urbanos que quieren su bici lista en 24 horas]**
- Qué quiero que haga quien entre: **[LA ACCIÓN PRINCIPAL. Ej.: pedir presupuesto por WhatsApp]**
- Web de la que sacar la información: **[URL DE TU WEB ACTUAL, O DE UNA WEB DE EJEMPLO SI AÚN NO TIENES]**
- Mi mascota: **[DESCRÍBELA, O ESCRIBE "INVÉNTALA A PARTIR DE MI LOGO". Ej.: un erizo naranja con casco de ciclista y mochila]**
- Colores de marca: **[TUS COLORES, O ESCRIBE "SÁCALOS DE MI LOGO"]**
- Tono: **[CÓMO LE HABLAS A TUS CLIENTES. Ej.: cercano, directo y con algo de humor]**
- Mis servicios o productos, entre 4 y 6 (serán los distritos): **[LISTA. Ej.: Reparación exprés · Tienda de bicis · Alquiler · Rutas guiadas · Mantenimiento para empresas]**
- Contacto: WhatsApp **[TU NÚMERO CON PREFIJO. Ej.: 34600000000]** · Email **[TU EMAIL]** · Dirección **[TU DIRECCIÓN, O BORRA ESTO SI NO TIENES LOCAL]**
- Idioma de la web: **[IDIOMA]**

Junto a este mensaje te adjunto: **[LO QUE ADJUNTAS. Ej.: mi logo, 8 fotos de trabajos, mi lista de precios, 3 opiniones reales y capturas de mi web actual]**. Úsalo todo: mi logo manda sobre la mascota y los colores; mis fotos y mis textos mandan sobre el contenido.

2. EL CONTENIDO

- Si puedes navegar, entra en la web que te he indicado y saca de ahí servicios, textos, precios, trabajos, opiniones, contacto e imágenes (con su URL completa).
- Si esa web no es mía sino un ejemplo, úsala solo como referencia de estructura: nunca copies sus textos, su marca ni sus imágenes.
- Si no puedes navegar, trabaja con lo que te adjunto y con lo que sabes de mi sector.
- No inventes opiniones, cifras, premios ni nombres de clientes. Donde falte un dato real, escribe un ejemplo creíble y márcalo con (EJEMPLO: CAMBIAR).
- Escribe para vender: titulares que digan qué gana el cliente, frases cortas y una llamada a la acción en cada panel.
- Todo vive en UN único array STATIONS (un objeto por servicio más uno de contacto) con: id, número, título, nombre de zona, titular, texto, 3 ventajas, 2 trabajos con imagen, posición, punto approach, color e icono. Geometría, etiquetas, botones, paneles y caminos se generan desde ahí: ninguna coordenada se escribe dos veces.

3. LA MASCOTA 3D

- Diseñala a partir de mi logo o de mi descripción: toma de 3 a 5 rasgos (colores HEX, forma de la cabeza, ojos, accesorio) y traduce cada uno a una pieza concreta, para que quien conozca la marca la reconozca.
- Modelala con código combinando esferas escaladas, cajas redondeadas (RoundedBoxGeometry, o BoxGeometry si da problemas), cilindros, toros y tubos sobre curvas. Sin modelos externos ni GLTF.
- Receta de juguete: una cabeza o cuerpo grande que se reconozca por su silueta, ojos grandes con brillo y un accesorio de tamaño generoso. Altura total de unas 2 unidades en un mundo de unas 36×33 .
- Jerarquía root → body → head, mirando hacia +Z. La navegación mueve y gira root; la animación solo toca el resto.
- Al caminar: rebote suave, ligera inclinación hacia delante, balanceo lateral y extremidades con desfase, al ritmo de la velocidad real. En reposo: respira, mira a los lados y parpadea cada 4 a 6 segundos.

- Gira hacia donde va por el camino más corto y de forma suave. Lleva una sombra de contacto falsa (plano con degradado radial) que la sigue.

4. EL MUNDO

- Base: isla flotante de 36×33 unidades, rectangular con las esquinas achaflanadas 3 unidades (el suelo que se pisa es ese contorno menos 1 unidad de margen), con canto grueso, losas escalonadas debajo, juntas de pavimento, tiras de luz en los bordes y calles guía hacia cada distrito. El suelo de la isla, en un tono medio de mi paleta (luminosidad entre el 30 % y el 45 %), para que destaque claramente sobre el fondo oscuro.
- Un distrito por servicio más el de contacto, colocados en manzanas (reparto exacto más abajo). Plantilla común (plataforma del color del distrito, pantalla con una maqueta de mi trabajo y cartel con su nombre) más una instalación con objetos que evoquen ese servicio (Ej.: para un taller, una bici gigante en su soporte). Cada distrito tiene una plataforma de 8×7 que se puede pisar y reúne entre 5 y 9 piezas de alturas distintas (un volumen alto tipo edificio o torre de 4 a 7 unidades de altura, la pantalla grande de unas 3×2 elevada del suelo, los objetos del servicio a escala de 1,5 a 3 unidades y detalles con luz emisiva), para que se vea como un escenario y no como una losa plana. Al menos 3 distritos con algo animado.
- Núcleo central: una pieza icónica de la marca que flota y gira despacio entre anillos luminosos, en una base de 5×5 como mucho. Es el punto de orientación.
- Reparto que garantiza que se puede caminar a todas partes: una cuadrícula de 3×3 manzanas con centros en $x = -11, 0, 11$ y $z = -10, 0, 10$. La manzana central es la plaza del núcleo. Los distritos ocupan las otras ocho (el de contacto, en la del centro del lado que ve la cámara al empezar) y las que sobren son parques bajos sin obstáculos. La caja de obstáculo de cada distrito rodea solo sus volúmenes y, ya hinchada, mide como mucho $7,5 \times 6,5$, así que entre manzanas siempre quedan calles libres. La mascota empieza en la calle entre el núcleo y el distrito de contacto, nunca dentro de una caja.
- Entorno lejano: islas con edificios de semilla fija dibujados con InstancedMesh, drones en órbita, partículas y luces que pulsan en mis colores.
- Paleta: fondo oscuro derivado de mi color principal y acentos con mis colores.
- Luz y niebla con estos valores exactos, porque con otros el mundo sale negro o desaparece:
 - `Renderer` con `toneMapping ACESFilmicToneMapping, toneMappingExposure 1.2` y `outputColorSpace SRGBColorSpace`.

- Niebla: new THREE.Fog(color del fondo, 76, 160). Tiene que empezar MÁS LEJOS que la cámara (que está a 65 del foco): nunca una niebla que empiece antes ni FogExp2, o la isla queda tapada.
- HemisphereLight (cielo claro, suelo oscuro de mi paleta, 1.7).
- DirectionalLight principal cálida de intensidad 2.7 en (-16, 28, 20), con sombra de 2048 (1024 en móvil) y cámara de sombra de ± 29 .
- DirectionalLight de contraluz de mi color secundario, intensidad 1.8, en (15, 18, -25).
- PointLight de mi color principal bajo el núcleo, intensidad 48, distancia 18 y decay 2, y otra de relleno de intensidad 25 y distancia 24. En three.js actual las luces puntuales usan unidades físicas: con intensidades de 1 o 2 no se ve nada.
- Pantallas y carteles con CanvasTexture dibujada en código y MeshBasicMaterial con toneMapped en false. Nunca cargues imágenes de otra web como textura 3D: dan errores de CORS.

5. CÁMARA Y MOVIMIENTO

- OrthographicCamera con frustum $\pm \text{zoom} \times \text{aspecto}$ en horizontal y $\pm \text{zoom}$ en vertical, near 0.1, far 250, en órbita a distancia 65 del foco, giro inicial 0.55 rad y elevación 0.68 rad (limitada entre 0.38 y 1.16). updateProjectionMatrix tras cada cambio; no uses camera.zoom.
- Zoom base: el mayor entre 22 y 27 dividido por el aspecto, limitado entre 11 y 42, de forma que al empezar la isla llene entre el 80 % y el 90 % del ancho del área del mundo. ResizeObserver sobre el contenedor (no window.resize) y renderer.setSize con el tercer parámetro en false.
- Clic o toque: si el puntero se movió 7 px o menos, raycast contra las pantallas (va a ese distrito) y, si no acierta, contra el suelo (camina a ese punto). Si se movió más, orbita. setPointerCapture al arrastrar.
- Zoom con rueda (listener con passive false y preventDefault) y con pellizco. touch-action none solo en el lienzo. Botones de acercar, alejar, recentrar y pausar.
- Teclado: WASD y flechas mueven respecto a la cámara, deslizándose contra obstáculos; E abre el distrito cercano.
- La cámara sigue a la mascota solo al 30 % de su posición y con interpolación exponencial. Mascota a unas 5.5 unidades por segundo, con dt limitado a 0.05.
- A* sobre rejilla de paso 0.35 a 0.5. Obstáculos: cajas AABB en UNA sola constante que usan la geometría y el A*, hinchadas con holgura de 0.6 a 0.85. Ocho vecinos sin diagonales que corten esquinas, heurística euclídea, segmento libre con prueba de slabs y suavizado saltando al punto visible más lejano.

- El destino de cada distrito es su punto approach, libre delante del edificio. Calcúlalo en código, no a mano, trabajando con celdas de la rejilla y no con coordenadas sueltas: desde la celda del centro del distrito, avanza celda a celda hacia el centro del mundo hasta la primera celda libre, avanza una celda más y guarda como approach el centro exacto de esa celda (si lo dejas pegado a la caja, al redondearlo a la rejilla cae dentro y no se llega nunca). Si la celda de origen o la de destino está ocupada, usa la celda libre más cercana, y el límite de iteraciones del A* nunca es menor que el número de celdas de la rejilla. Fuera de la isla o sin ruta: mensaje amable, nunca una excepción. Al arrancar, recorre la rejilla en anchura desde la posición inicial de la mascota y comprueba con `console.assert` que llega al approach de todos los distritos.

6. ETIQUETAS Y PANELES

- Sobre cada distrito, un button HTML con su número sobre el color del distrito, el título y una flecha, dentro de una capa con `position absolute` e `inset 0` en el mismo wrapper que el lienzo.
- Cada frame: `project(camera)` y paso a píxeles con el tamaño del lienzo, no de la ventana: $x = (\text{ndc.x} \times 0.5 + 0.5) \times \text{ancho}$, $y = (-\text{ndc.y} \times 0.5 + 0.5) \times \text{alto}$. Oculta la etiqueta fuera de cámara. Mueve con `transform translate` y mide solo al cambiar el ancho.
- Sin solapes ni salidas: márgenes de 10 px a los lados, 52 arriba y 18 abajo. Ordena por altura y prueba desplazamientos de 0, ± 52 y ± 104 px en vertical por 0, ± 50 y ± 100 en horizontal; quédate con el que no solape (margen 7 px). Si aun así se pisan, cuadrícula compacta centrada.
- Etiquetas y botones marcan los distritos visitados, con contador "n / total".
- Al llegar se abre el panel. Mientras camina, un mensaje dice a dónde va, con un botón "Abrir ya". Un clic en otro sitio cambia la ruta.
- UN único dialog nativo con `showModal()`; mientras está abierto se pausan el render y la navegación. Cierre con X, Escape o clic en el fondo solo si el pointerdown empezó fuera. Foco al título al abrir y de vuelta al elemento que lo abrió al cerrar.
- Panel de servicio: pestañas entre servicios, zona y número, titular en 2 líneas, texto, 3 ventajas, botón de la acción principal que abre contacto con ese servicio elegido y galería de 2 trabajos. Cada imagen, creada con `createElement` y un listener de error, se cambia por un bloque del color del distrito con el título: nunca un icono roto. Nunca escribas un atributo `onerror` dentro de una cadena de HTML: las comillas anidadas rompen el script entero.

- Panel de contacto: nombre, email o teléfono, servicio (los míos más "Quiero consejo") y mensaje, con validación y errores junto a cada campo. Al enviar, resumen y aviso de que aún no se ha enviado nada, con tres botones: abrir WhatsApp (<https://wa.me/> con mi número y el texto con encodeURIComponent), preparar email (mailto con asunto y cuerpo) y copiar. Ningún fetch a servidores inventados. El borrador sobrevive a cerrar el panel.

7. PORTADA Y NAVEGACIÓN ALTERNATIVA

- La portada vende desde el primer segundo y el mundo nunca la tapa. En escritorio, columna izquierda de entre el 40 % y el 46 %: frase de gancho, titular grande en 3 líneas con la última en mi color, subtítulo, botón "Explorar el mundo" y enlace a la acción principal. El mundo va detrás y a la derecha, desde el 29 % del ancho, fundido con la página con una máscara de degradado.
- Cabecera con logo y botón fijo de la acción principal. Barra inferior con un botón por distrito y "Vista rápida": rejilla de tarjetas HTML sin 3D que abren los mismos paneles.
- Pista breve de controles, distinta en táctil.

8. MÓVIL

- Por debajo de 700 px: menú hamburguesa, portada a ancho completo y el mundo como franja en el flujo, al 100 % de ancho y unos 415 px de alto. "Explorar el mundo" hace scroll hasta ella.
- Botones de al menos 44 px, distritos en rejilla de 2 o 3 columnas y etiquetas de 145 px de ancho como máximo.
- Paneles casi a pantalla completa (ancho menos 20 px y 94 % del alto), en 1 columna, con campos a 16 px para que el iPhone no haga zoom.

9. REQUISITOS TÉCNICOS

- UN ÚNICO archivo index.html con todo dentro, que funcione con solo abrirlo y se pueda subir a cualquier hosting.
- three@0.170.0 como módulo, con un importmap que asigne "three" a build/three.module.js y "three/addons/" a examples/jsm/, las dos rutas dentro de <https://cdn.jsdelivr.net/npm/three@0.170.0/> (escribelas completas). Nada de three/examples/js ni scripts UMD. Sin otras librerías; si alguna fuera imprescindible, solo de cdnjs o jsDelivr con versión fija.
- Bloom opcional (UnrealBloomPass con umbral alto) detrás de una constante.

- Todo el JavaScript en un único script type="module", con las funciones en el nivel superior (nunca dentro de un if ni de otra función). Sin WebGL siguen existiendo y no hacen nada.
- Para ocultar y mostrar elementos usa el atributo hidden y añade al CSS una regla que oculte siempre todo lo que lleve ese atributo, con display none e !important; si no, una regla display del propio elemento lo deja visible (un menú móvil «oculto» acaba tapando la web entera).
- La llamada que lo arranca todo va en la última línea del script, después de declarar todas las const y let. Si va antes, el navegador lanza «Cannot access ... before initialization» y el mundo no carga.
- Imágenes con las URL completas de mi web; las adjuntas, con rutas assets/nombre.jpg. El logo, como SVG en línea.
- Rendimiento: pixel ratio máximo 2, sombra actualizada cada 5 frames, render en pausa con panel abierto, en vista rápida o con la pestaña oculta. Si 3 tandas seguidas de 1,5 s van por debajo de 48 fps, baja la calidad una vez (pixel ratio 1, sin bloom, sombra menor).
- Accesibilidad: lienzo con role application y aria-label con los controles, foco visible y aria-live en los mensajes.
- Reduce-motion: prefers-reduced-motion con listener. Si está activo o se pulsa pausa, se paran las animaciones del mundo y de reposo, pero la mascota sigue caminando cuando se le pide.
- Sin WebGL: render en try/catch y listener de webglcontextlost. Si falla, oculta el mundo, activa la vista rápida y avisa: la web sigue entera.
- SEO: title, meta description, Open Graph, JSON-LD de mi tipo de negocio y un noscript con servicios y contacto.
- Cero errores en la consola. Como mucho unas 1.800 líneas: si no cabe, recorta el entorno lejano, nunca funciones.
- Antes de entregar, repasa tres cosas: que con la cámara a 65 del foco, la niebla empezando en 76 y las luces anteriores la isla se ve entera, iluminada y con el suelo más claro que el fondo; que la mascota puede llegar andando a todos los distritos desde donde empieza; y que ninguna función usa una const o let declarada más abajo antes de que exista.

10. SI ALGO CHOCA, ESTE ES EL ORDEN DE PRIORIDAD

1. Que funcione sin errores. 2. Que la portada venda y el contacto funcione. 3. Que se use bien en el móvil. 4. Que la mascota camine sin atravesar edificios. 5. Todo lo demás.

11. CÓMO ME LO ENTREGAS

1. Primero, en 10 líneas como máximo: cómo es la mascota (piezas y colores), la paleta con sus códigos HEX y qué instalación tiene cada distrito.
2. Después, el archivo index.html completo, sin "..." ni trozos por rellenar. Si no te cabe en una respuesta, para al terminar una función y, cuando te escriba "continúa", sigue exactamente desde ese punto.
3. Al final: la lista de textos marcados con (EJEMPLO: CAMBIAR), las imágenes que tengo que poner en la carpeta assets/ y cómo subir la web a un hosting con mi propio dominio (por ejemplo, Hostinger: Administrador de archivos, carpeta public_html).

No me hagas preguntas antes de empezar: si falta algún dato, decide tú con criterio y déjalo marcado.

TRUCOS QUE MARCAN LA DIFERENCIA

01 Usa el modelo más potente que tengas

Un mundo 3D pide mucho código. En Claude, ChatGPT o Gemini, elige su modelo más avanzado.

02 Mírala en la vista previa de la IA

Claude la muestra como artefacto, también en su plan gratis; en Gemini, activa Canvas antes de enviar el prompt. En la vista previa no salen tus fotos: es normal.

03 Pide cambios de uno en uno

«Haz la mascota más grande», «añade un distrito de ofertas». En el mismo chat y pidiendo solo el trozo que cambia.

04 Busca **(EJEMPLO: CAMBIAR)**

Son los textos que la IA no tenía. Cámbialos por datos reales antes de publicar.

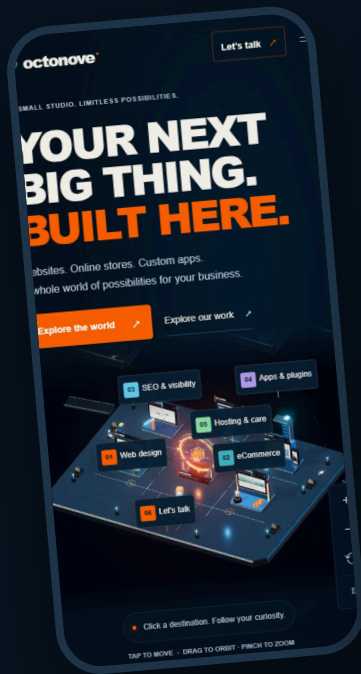
05 Publícala con tu dominio

Sube el index.html y la carpeta assets a un hosting como **Hostinger**: con el plan Premium contratado por 12 meses o más, el dominio es gratis el primer año. (Enlace de afiliado · sin coste extra para ti)

● Y AHORA, A POR ELLA

¿YA TIENES TU MUNDO 3D?

Súbelo a tus stories y etiqueta a [@simplificaconia](#). Nos encantará recorrerlo.



TAMBIÉN EN EL MÓVIL

Más prompts, guías paso a paso y herramientas de IA gratis en simplificaconia.com

Web de ejemplo diseñada por Octonove · octonove.com