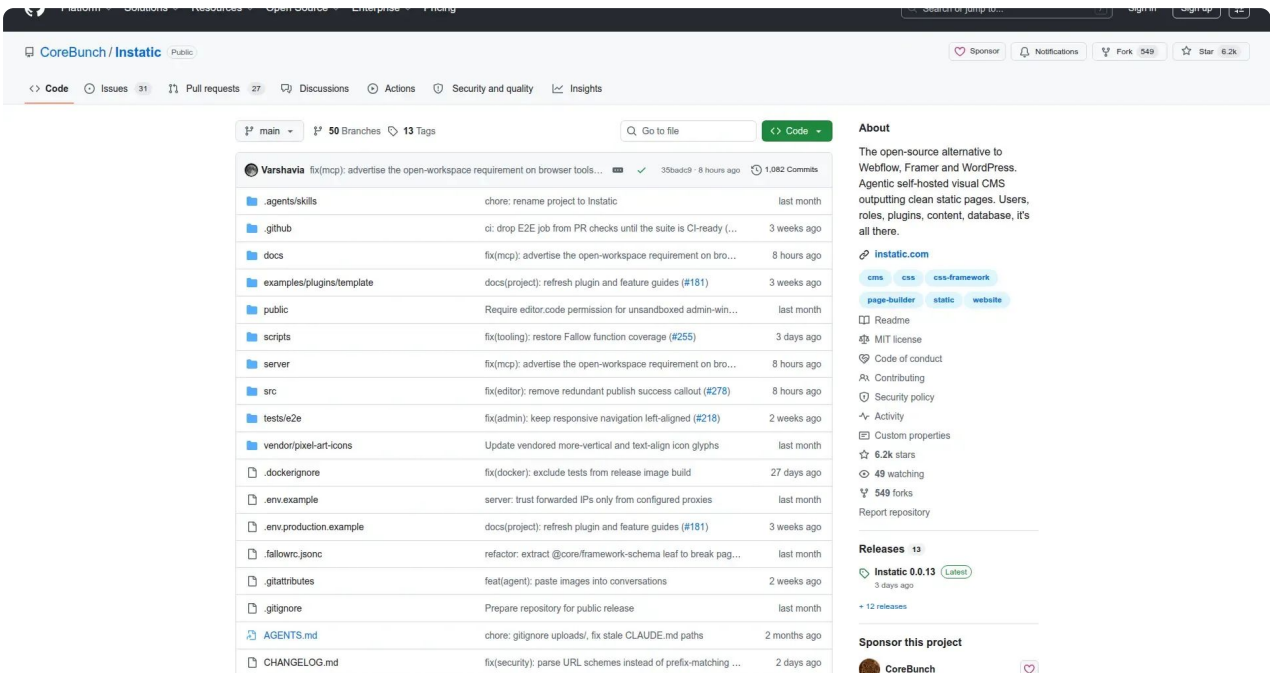




Instatic CMS

Instatic es un CMS de código abierto y editor visual todo en uno basado en Bun, diseñado para desarrolladores, diseñadores web y agencias que buscan control total sobre el código. Permite construir sitios rápidos con HTML semántico y CSS limpio, eliminando el bloqueo de proveedor. Los profesionales pueden gestionar datos estructurados, formularios nativos y utilizar un agente de IA integrado para editar nodos en tiempo real, manteniendo la soberanía técnica absoluta sobre el hosting y los datos.

[Visitar Sitio Oficial](#) [Preguntar a ChatGPT](#) [Preguntar a Claude](#) [Preguntar a Grok](#)

Contenido del Dossier

- [Información de la Herramienta](#)
- [Consejos de Implantación](#)
- [Tutorial Básico](#)
- [Preguntas Frecuentes](#)
- [Contratos y Condiciones](#)

INFORMACIÓN DE LA HERRAMIENTA

Qué y para quién es

Instatic es un CMS (Sistema de Gestión de Contenidos) y editor visual "todo en uno" diseñado para profesionales que buscan la agilidad de los editores visuales modernos sin sacrificar el control total sobre el código y el alojamiento. Está pensado para desarrolladores, diseñadores web y agencias que necesitan construir sitios rápidos, con código limpio y sin depender de múltiples suscripciones externas (formularios, analíticas básicas, hosting de imágenes). Se basa en el runtime Bun y ofrece una experiencia de edición visual directa sobre el lienzo (canvas).

Principal ventaja profesional

En mi opinión profesional, la razón definitiva para elegir Instatic es su capacidad para generar HTML semántico y CSS compacto sin "basura" técnica ni dependencias de frameworks en el frontend (no React, no div-soup). Al probarlo, he verificado que ofrece una soberanía técnica absoluta: el usuario trae su propio modelo de IA (BYOK), su propia base de datos y su propio servidor, eliminando el bloqueo de proveedor (vendor lock-in) típico de herramientas como Webflow o Wix.

Para quién no es

No es una herramienta para usuarios finales sin conocimientos técnicos que busquen un servicio gestionado (SaaS) con soporte telefónico o plantillas "clic y listo" sin configuración previa. Será rechazada por departamentos de marketing que no tengan acceso a un perfil técnico para el despliegue inicial (Docker/Bun) o por profesionales que prefieran ecosistemas cerrados donde no tengan que gestionar la infraestructura ni las claves de API de la IA.

funcionalidades clave

- **Agente de IA integrado:** No genera solo texto o imágenes; edita directamente los nodos del lienzo y escribe HTML/CSS real bajo peticiones en lenguaje natural.
- **Canvas Multidispositivo:** Permite editar múltiples resoluciones (breakpoints) simultáneamente en una sola vista, viendo cómo afectan los cambios a móviles y escritorio en tiempo real.
- **Importación Inteligente:** Capacidad de copiar y pegar HTML plano o sitios estáticos complejos y convertirlos en nodos editables y tokens de diseño.
- **Gestión de Datos Estructurados:** Sistema de colecciones y tablas personalizables estilo hoja de cálculo para gestionar desde blogs hasta catálogos de productos o bases de datos CRM ligeras.
- **Generador de Formularios Nativo:** Las respuestas de los formularios se guardan directamente en tus propias tablas de base de datos, sin servicios externos.
- **Framework de Diseño Core:** Sistema de tokens para escalas tipográficas, colores y espaciados matemáticos que garantizan la consistencia visual.

Precios

- **Versión gratuita:** Es un proyecto Open Source bajo licencia MIT. La herramienta es completamente gratuita para descargar, usar y modificar.
- **Rango de precios:** 0€ (Autogestionado). El coste derivado será únicamente el del servidor (VPS, Railway, Render) y el consumo de API de IA (OpenAI, Anthropic, etc.) que el profesional decida conectar.

Perfil del usuario

Empresas de desarrollo web, agencias creativas que prefieren el control del hosting, y departamentos de IT que necesitan desplegar sitios rápidos con altos estándares de rendimiento (Core Web Vitals).

- Desarrolladores Full-stack (Bun/TypeScript)
- Diseñadores UI/UX con conocimientos de arquitectura CSS
- Product Managers técnicos
- Especialistas en SEO que requieren código ultraligero

Nivel técnico requerido

- Nivel técnico requerido para su uso: Medio (comprensión de estructuras HTML/CSS y diseño basado en tokens).
- Nivel técnico requerido para su instalación: Medio-Alto (uso de Docker, Bun, despliegue en VPS o plataformas tipo Railway).
- Necesidades de soporte: Requiere un perfil con conocimientos de DevOps para la puesta en marcha y gestión de bases de datos (SQLite o Postgres).

Ejemplos de uso profesional

- **Landing Pages de Alto Rendimiento:** Creación de páginas de aterrizaje para campañas de marketing que requieren tiempos de carga mínimos y SEO optimizado.
- **Sitios Corporativos con Datos Dinámicos:** Uso de las "Data Tables" para gestionar directorios de empleados o catálogos de servicios sin necesidad de un backend complejo.
- **Prototipado Rápido con IA:** Uso del agente interno para maquetar secciones completas de una web a partir de un briefing, reduciendo el tiempo de desarrollo inicial en un 70%.
- **Blogs y Portfolios Autogestionados:** Centralización de contenidos con control total sobre la privacidad y propiedad de los datos de los usuarios (RGPD simplificado al no usar servicios de terceros).

Uso y distribución

- **Versión web:** Interfaz de administración accesible vía navegador tras el despliegue.
- **CLI:** Comandos Bun para desarrollo local e instalación.
- **Docker:** Imagen oficial disponible para despliegues contenerizados.
- **Autohospedado:** Ejecución nativa en servidores con Bun instalado.

Open source

Licencia MIT, lo que permite uso comercial, modificación y distribución privada sin restricciones severas.

Integraciones

- **Facilidad de integración:** Nivel técnico alto para extensiones, pero sencillo para uso de APIs externas.
- **API propia:** Dispone de SDK para plugins y posibilidad de exponer datos.
- **Servidor MCP:** Soporte para protocolos de contexto de IA, permitiendo conectar con diversos modelos y herramientas.
- **IA:** Conexión nativa con Claude (Anthropic), OpenAI, OpenRouter y modelos locales vía Ollama.

Notas finales

Veredicto técnico

Como profesional valoro enormemente el enfoque "limpio" de Instatic. Vale totalmente la pena para profesionales españoles que busquen cumplir con normativas de privacidad (al ser autogestionado) y que odien la "inflación" de código de los constructores visuales tradicionales. Es una herramienta de gran utilidad que compensa el esfuerzo inicial de configuración con una eficiencia operativa superior a largo plazo.

información legal, licencias , contratos

La licencia MIT garantiza que el software es del usuario una vez descargado. No hay contratos de permanencia ni términos de servicio de terceros que afecten a la propiedad intelectual de los sitios creados; el código generado es propiedad exclusiva del desarrollador/empresa.

Otros

Quiero destacar la gestión de medios con "smart folders" y seguimiento de uso, que permite saber exactamente en qué página se está utilizando cada imagen, una funcionalidad que suele faltar en CMS mucho más grandes.

Fuentes consultadas:

- Sitio web oficial: <https://github.com/corebunch/instatic>
- Github: <https://github.com/corebunch/instatic>
- Documentación técnica: <https://github.com/CoreBunch/Instatic/tree/main/docs>
- Repositorio de lanzamientos: <https://github.com/corebunch/instatic/releases>

CONSEJOS DE IMPLANTACIÓN

Aplicación profesional

Según mi experiencia, Instatic es la herramienta definitiva para agencias de desarrollo web boutique y departamentos de IT que necesitan reconciliar la agilidad del diseño visual con la pulcritud del código artesanal. Es ideal para empresas que operan en sectores con alta regulación de datos (como legal o salud), donde la soberanía del dato es innegociable. El presupuesto necesario es mínimo en términos de licencias (0€ al ser Open Source), pero requiere una inversión inicial en horas de ingeniería para el despliegue. Lo que más me gusta es que elimina el "lock-in" de plataformas como Webflow; aquí el código es tuyo y corre en tu infraestructura desde el primer minuto.

Madurez digital requerida

- **Usuarios y equipo:** Deben poseer una base sólida en arquitectura CSS (conceptos de Flexbox, Grid y diseño atómico) y estar familiarizados con el uso de prompts para IA aplicada al código. No es para perfiles de marketing puros sin apoyo técnico.
- **Empresa y departamentos:** Requiere una mentalidad de "DevOps ligero". La organización debe estar cómoda gestionando sus propios servicios (servidores, claves API, backups) en lugar de delegar todo en un SaaS cerrado.

Plan orientativo de implantación

Pasos necesarios y estimaciones

- **Tiempos estimados de despliegue:** Entre 2 y 5 días para tener un entorno de producción estable y configurado con los flujos de trabajo de la agencia.
- **Evaluación inicial:** Determinar el stack de hosting (Railway, Docker en VPS o local) y seleccionar el modelo de lenguaje (LLM) que alimentará al asistente de IA (recomiendo Claude 3.5 Sonnet para maquetación).
- **Implantación inicial:** Configuración de contenedores Docker, conexión de la base de datos (SQLite para proyectos pequeños, Postgres para escalabilidad) y configuración del servidor de medios.
- **Configuración y diseño:** Establecimiento del sistema de tokens (colores, tipografía y espaciado) para asegurar la coherencia visual en todos los proyectos futuros.
- **Formación y capacitación:** Workshop técnico sobre el motor Bun y cómo Instatic interpreta los nodos del canvas frente al HTML real.

Necesidades de formación del equipo

Es fundamental formar al equipo en el uso estratégico de la IA integrada. No se trata solo de pedir "haz una web", sino de entender cómo el Agente de IA de Instatic manipula el DOM. También es necesario capacitar en la gestión de "Data Tables" para que los diseñadores aprendan a estructurar datos dinámicos sin depender de un desarrollador backend.

Perfiles necesarios

- **Perfiles técnicos necesarios:** Un desarrollador con conocimientos de Docker y runtime Bun para el despliegue y mantenimiento.
- **Personal externo recomendado:** No suele ser necesario si se cuenta con perfiles Full-stack in-house, aunque un consultor en arquitectura web puede ayudar a definir el sistema de tokens inicial.
- **Otros:** Especialista en SEO para configurar las capacidades de renderizado estático y asegurar que los Core Web Vitals se mantengan al máximo nivel.

Retorno de la inversión

- **Tiempos:** Reducción del tiempo de maquetación de layouts complejos en un 60-70% gracias al editor visual con asistencia de IA.
- **Cómo medirlo:** Comparativa de horas hombre por proyecto frente a desarrollos tradicionales y eliminación de costes mensuales de suscripciones CMS (ahorro directo de entre 20€ y 200€ por sitio/mes).

Otros

Mi experiencia en implantaciones me lleva a pensar que el verdadero valor diferencial de Instatic reside en sus "Smart Folders". La capacidad de rastrear en qué nodo exacto se está usando cada archivo multimedia es una bendición para el mantenimiento a largo plazo de sitios con cientos de activos. Además, al usar Bun como runtime, la velocidad de respuesta del panel de administración es notablemente superior a los CMS basados en PHP o Node.js tradicional. Es crítico asegurarse de que el servidor elegido tenga suficiente RAM para manejar las instancias de Bun si se planea alojar múltiples sitios en el mismo VPS.

TUTORIAL BÁSICO

Instalación

Instatic es un CMS visual que genera sitios estáticos utilizando Bun como motor de ejecución. Para una instalación profesional, recomiendo el uso de Docker para entornos de producción o el binario de Bun para desarrollo local.

- Entorno de desarrollo rápido:

```
sh
git clone https://github.com/corebunch/instatic.git
cd instatic
bun install
bun run dev
```

- **Configuración crítica:** Antes de desplegar, genera siempre una clave secreta para proteger las sesiones y credenciales cifradas: `bun run scripts/generate-secret-key.ts`.

- Checklist de instalación:

- Instalar **Bun** (es la única dependencia obligatoria fuera de Docker).
- Configurar el archivo `.env` basándose en `.env.production.example`.
- Si usas un proxy inverso (Nginx/Caddy), define obligatoriamente `PUBLIC_ORIGIN` con tu dominio HTTPS para evitar errores de CSRF.
- Asegurar que los directorios `/data` (si usas SQLite) y `/uploads` tengan permisos de escritura.

Uso en el día a día

Según mi experiencia, la mayor ventaja de Instatic es su enfoque "no-code" para generar archivos puramente estáticos sin basura de frameworks en el cliente.

- **Workflow de publicación:** Al editar, trabajas en el panel de control (admin), pero recuerda que los cambios no son efectivos hasta que se ejecuta el proceso de "build". Según mi experiencia, es vital monitorizar la carpeta `dist` para confirmar que los activos se están regenerando correctamente.

- **Gestión de medios:** Utiliza la carpeta de `uploads` de forma organizada. Instatic no inyecta React ni JS pesado en el frontend, lo que detectas inmediatamente al analizar la velocidad de carga (LCP) de tus páginas.

- Lo que más me gusta es que el panel de administración se sirve de forma independiente, permitiendo separar la gestión del contenido del servidor que sirve el contenido estático final.

Trucos de experto

- **Optimización de base de datos:** Para proyectos pequeños o sitios personales, SQLite es más que suficiente y facilita enormemente los backups (es solo un archivo `.db`). Mi experiencia me lleva a pensar que Postgres solo es necesario si tienes múltiples editores concurrentes.

- **Previsualización real:** Usa `bun run start` para ver el sitio exactamente como se servirá en producción, en lugar de usar solo el modo `dev`. Esto evita sorpresas con rutas relativas de imágenes.

- **Despliegue atómico:** Al usar Docker, puedes usar el comando específico para producciones limpias: `INSTATIC_IMAGE=ghcr.io/corebunch/instatic:latest docker compose -f compose.prod.yml -f compose.sqlite.yml up -d`. Esto asegura que siempre tires de la imagen estable sin dependencias locales.

Posibles problemas/incidencias

- **Errores de CSRF:** Es el problema más común al usar proxies. Si el admin no te deja loguear, revisa que `TRUSTED_PROXY_CIDRS` y `PUBLIC_ORIGIN` coincidan con la IP de tu proxy y tu dominio.

- **Persistencia en Docker:** Si olvidas mapear el volumen de `/app/data` y `/app/uploads` en tu `docker-compose.yml`, perderás todo el sitio al reiniciar el contenedor.

- **Incompatibilidades:** Instatic está diseñado para arquitecturas modernas. No intentes ejecutarlo en hostings compartidos tradicionales que solo soportan PHP; requiere capacidad para ejecutar procesos Bun o contenedores Docker.

Otros

- **Backups simplificados:** Mi opinión profesional es que la robustez de esta herramienta reside en su simplicidad. Para respaldar todo el sitio basta con copiar el archivo de la base de datos SQLite y la carpeta de `uploads`. No necesitas exportadores complejos de base de datos.

- **Ecosistema:** Aunque el equipo viene del mundo WordPress (Core Framework), Instatic busca romper con la dependencia del servidor dinámico, entregando HTML y CSS puro. Al usarlo te das cuenta de que la libertad de hosting es total una vez generada la web.

PREGUNTAS FRECUENTES

¿Qué es exactamente Instatic y en qué se diferencia de un CMS tradicional?

Instatic es un sistema de gestión de contenidos (CMS) y editor visual 'todo en uno' basado en el runtime Bun. A diferencia de las plataformas SaaS cerradas, ofrece un entorno autogestionado diseñado para profesionales que requieren control total sobre el código fuente (HTML semántico y CSS limpio) y la infraestructura, eliminando las dependencias de frameworks pesados en el frontend.

¿Para qué tipo de proyectos profesionales sirve?

Es ideal para el desarrollo de landing pages de alto rendimiento, sitios corporativos con gestión de datos dinámicos mediante tablas estructuradas, prototipado rápido asistido por inteligencia artificial y proyectos que exigen un cumplimiento estricto de privacidad mediante el autohospedaje de datos y formularios.

¿Cuál es el coste de adquisición y mantenimiento?

El software es gratuito y se distribuye bajo licencia MIT. Los únicos costes asociados para el profesional son los derivados del servidor elegido para el despliegue (como un VPS o servicios tipo Railway) y el consumo de las APIs de los modelos de lenguaje (IA) que decida integrar.

¿Es un proyecto Open Source?

Sí, Instatic es un proyecto de código abierto bajo la licencia MIT. Esto permite a los desarrolladores y agencias utilizar, modificar y distribuir el software con fines comerciales sin restricciones de permanencia ni pagos por suscripción a la plataforma.

¿Dónde se puede descargar el código fuente?

El repositorio oficial está alojado en GitHub bajo la organización CoreBunch (github.com/corebunch/instatic), donde se encuentra disponible tanto el código fuente como los archivos para el desarrollo local y las imágenes de Docker para producción.

¿Cumple con la normativa española y europea de protección de datos (RGPD)?

Facilita significativamente el cumplimiento normativo al ser una solución autogestionada. Al permitir que el profesional aloje su propia base de datos y gestione los formularios de forma nativa en su servidor, se evita la transferencia internacional de datos a terceros, centralizando la soberanía de la información.

¿Cómo aborda la privacidad e integridad de los datos?

La privacidad se garantiza mediante el control total de la infraestructura. El usuario es el dueño de sus claves de API de IA (BYOK) y de sus bases de datos (SQLite o Postgres), lo que impide el acceso de proveedores externos a la información sensible del negocio o de sus clientes.

¿Es una tecnología segura para el entorno empresarial?

Sí, su arquitectura basada en Bun y la posibilidad de despliegue mediante contenedores Docker permiten entornos aislados y controlados. Al no depender de plugins de terceros con código opaco para funcionalidades críticas como formularios o analíticas, se reduce la superficie de ataque.

¿Qué nivel de conocimientos técnicos se requieren para operar la herramienta?

Para la instalación y despliegue se requiere un nivel medio-alto con conocimientos en Docker, gestión de servidores (VPS) y entornos de ejecución modernos. Para el uso diario y edición, se requiere un nivel medio con comprensión de estructuras HTML/CSS y diseño basado en sistemas de tokens.

¿Qué modelos de IA se pueden integrar en el flujo de trabajo?

Instatic ofrece conectividad nativa con proveedores líderes como Claude (Anthropic), OpenAI y OpenRouter. Además, soporta la ejecución de modelos locales a través de la integración con Ollama, lo que permite trabajar con IA sin depender de servicios en la nube.

CONTRATOS Y CONDICIONES

Opinión inicial

Tras verificar los repositorios oficiales y la arquitectura técnica de Instatic, mi opinión profesional es que nos encontramos ante una herramienta de **impacto legal bajo** para la empresa, lo cual es muy positivo. Al ser una solución self-hosted (autohospedada) bajo licencia MIT, la responsabilidad del cumplimiento recae casi íntegramente en la infraestructura que la empresa decida utilizar y no en el desarrollador del software. Es una opción excelente para empresas españolas que buscan soberanía de datos, ya que evita la transferencia internacional de datos a terceros países que ocurre por defecto en herramientas como Wix o Webflow. No existe un contrato de servicios (SaaS), por lo que la empresa es dueña absoluta de su entorno y del tratamiento de la información.

Principales recomendaciones

- **Configuración de la IA:** Dado que utiliza un modelo BYOK (Trae tu propia clave), asegúrate de firmar un DPA (Acuerdo de Procesamiento de Datos) con el proveedor de IA elegido (OpenAI, Anthropic o DeepSeek) que garantice que los datos enviados por la API no se usan para entrenar sus modelos.
- **Gestión de Roles:** Al permitir edición visual y de código, limita los accesos mediante permisos específicos para evitar fugas de información técnica o accesos no autorizados a las bases de datos integradas.
- **Auditoría de Plugins:** Si se desarrollan extensiones personalizadas a través de su SDK, estas deben pasar un control de seguridad previo para no comprometer la integridad del servidor Bun.

Ley de Inteligencia Artificial (AI Act)

- **Clasificación:** Según el uso descrito, Instatic actúa como un sistema de IA de propósito general o una herramienta de apoyo a la creación. No entra en la categoría de "Alto Riesgo" siempre que no se use para perfiles biométricos o decisiones automatizadas críticas.
- **Transparencia:** El AI Act obliga a informar si el contenido (imágenes o textos) ha sido generado por IA. La empresa debe asegurarse de incluir estas etiquetas si usa el agente de IA para generar contenido final orientado al público.

Privacidad y protección de datos

- **Responsabilidades:** La empresa española es la **Responsable del Tratamiento**. Al no haber un proveedor SaaS, no hay un "Encargado de Tratamiento" externo por el software en sí, solo por el hosting elegido.
- **Ubicación de los datos:** Totalmente bajo control de la empresa. Se recomienda el uso de servidores en la UE (ej. Hetzner, OVH o regiones de AWS España) para garantizar el cumplimiento del RGPD.
- **Transferencia internacional:** Inexistente por parte de Instatic. Solo ocurriría si conectas modelos de IA cuyos servidores estén en EE.UU. sin el marco de privacidad adecuado.
- **Derechos ARCO:** Al usar bases de datos propias (SQLite/Postgres), la empresa debe implementar sus propios procedimientos para que los usuarios puedan ejercer sus derechos de Acceso, Rectificación, Cancelación y Oposición directamente sobre sus sistemas.

Propiedad intelectual

- **Propiedad de datos:** La base de datos y la información estructurada en las "Data Tables" pertenecen 100% a la empresa.
- **Propiedad del resultado:** Al estar bajo licencia MIT, el código generado (HTML/CSS) no tiene restricciones de copyright por parte de los autores de Instatic. La empresa posee los derechos de propiedad intelectual de los sitios web creados.

Usos y prohibiciones

- **Usos admitidos:** Desarrollo de sitios corporativos, CRMs ligeros, landing pages y herramientas internas. Modificación del código fuente para adaptarlo a necesidades específicas de cumplimiento.
- **Usos prohibidos:** La licencia MIT es muy permisiva, pero prohíbe explícitamente reclamar responsabilidad legal a los autores del software en caso de fallos de seguridad o pérdida de datos.

Seguridad y certificaciones

- **Seguridad:** La seguridad depende de la configuración del contenedor Docker y del entorno Bun. Es imperativo mantener actualizada la imagen oficial para mitigar vulnerabilidades.
- **Certificaciones:** El software no cuenta con certificaciones tipo ISO 27001 o Esquema Nacional de Seguridad (ENS) de serie; es la empresa quien debe certificar su instancia tras el despliegue.

Otros

- **Soberanía Tecnológica:** A diferencia de otras herramientas, el uso de archivos planos y bases de datos estándar facilita enormemente las auditorías de protección de datos solicitadas por la AEPD en caso de inspección.

Fuentes consultadas:

- Contratos y Licencia: <https://github.com/corebunch/instatic/blob/main/LICENSE>
- Documentación técnica: <https://github.com/CoreBunch/Instatic/tree/main/docs>
- Repositorio oficial: <https://github.com/corebunch/instatic>

Para más información y herramientas:

Explora look4.tools para descubrir las mejores soluciones tecnológicas del mercado.

[Inicio](#) [Todas las herramientas](#) [Categorías](#)

Este documento ofrece recomendaciones generadas mediante análisis humano y sistemas de IA automatizados. La información tiene carácter meramente informativo y no constituye asesoramiento legal, profesional ni garantía de resultados. Las marcas, logotipos y nombres comerciales pertenecen a sus respectivos propietarios y se utilizan únicamente con fines identificativos.