

RuralINT — Arquitectura

Stack tecnológico abierto, sin licencias y listo para producción

Principio rector: cero dependencia de licencias

Toda la infraestructura de RuralINT se construye sobre tecnologías de código abierto, estándares OGC reconocidos y APIs propias. No hay costes de licencia por usuario, por servidor ni por volumen de datos. Cualquier territorio puede desplegar su propio nodo sin compromisos comerciales con terceros.

Stack completo de producción

CAPA DE PRESENTACIÓN	
HTML5 / CSS3 / JS vanilla	Interfaz web responsiva sin frameworks propietarios. Visores interactivos, informe dinámico y agente IA integrados en páginas estáticas servidas por nginx.
Leaflet + GeoServer WMS/WFS	Visualización cartográfica con capas temáticas en tiempo real. Simbología normalizada por módulo (Go/Watch/No-Go) servida directamente desde GeoServer.
CAPA DE SERVICIOS IA	
Agente Data Analyst	Responde preguntas sobre KPIs, rankings y comparativas municipales. Accede a la base de datos en tiempo real y contextualiza resultados con referencias regionales.
Agente Policy Expert	Analiza el territorio desde la perspectiva de política pública. Relaciona indicadores con marcos normativos europeos y oportunidades de financiación.
Motor LLM	Modelo de lenguaje grande (LLM) de última generación accedido via API estándar. El sistema es agnóstico al proveedor — puede sustituirse sin cambios en la arquitectura.
CAPA DE API	
FastAPI (Python)	Backend de alto rendimiento con endpoints <code>/api/halland/</code> para datos y <code>/agent/</code> para los agentes IA. Autenticación por API key, rate limiting y caché TTL integrados.
nginx (proxy inverso)	Gestión de tráfico, SSL/TLS, routing hacia los servicios internos y servicio de estáticos. Configuración unificada para todos los componentes.
CAPA DE DATOS	
PostgreSQL + PostGIS	Base de datos relacional con extensión espacial. Esquema ruralint con tablas modulares por dimensión (demografía, economía, servicios, territorio) y vistas optimizadas para los visores.

GeoServer	Servidor de geodatos OGC. Publica capas WMS y WFS consumidas por los visores y compatibles con cualquier cliente SIG (QGIS, ArcGIS, OpenLayers).
SCB / INE / fuentes oficiales	Los datos provienen exclusivamente de fuentes estadísticas oficiales de cada territorio. Scripts Python automatizados para descarga, validación y carga.

Modelos de despliegue

RuralINT puede desplegarse en tres modalidades según las necesidades del territorio y la capacidad técnica del operador:

Cloud (SaaS)	Local (on-premise)	Híbrido
El territorio accede a la plataforma centralizada de Smart Terra. Sin infraestructura propia. Actualizaciones automáticas.	Despliegue completo en servidores del cliente o de la administración. Control total de datos. Cumplimiento NIS2 / GDPR simplificado.	API y datos en servidores locales. Interfaz y agente IA en cloud. Balance entre seguridad y coste.
<i>Ideal para municipios y mancomunidades sin equipo TI propio.</i>	<i>Ideal para diputaciones, comunidades autónomas y proyectos europeos.</i>	<i>Ideal para entidades con requisitos de soberanía de datos parcial.</i>

App propia para cada piloto

Cada territorio puede disponer de su propia instancia de RuralINT con identidad visual, dominio y roles personalizados. El sistema de roles permite configurar accesos diferenciados:

Rol	Acceso	Casos de uso
Administrador	Completo	Gestión de datos, configuración de capas y usuarios
Editor	Lectura + carga	Actualización de KPIs y documentos del territorio
Visor público	Solo lectura	Ciudadanía, medios, investigadores
Agente IA	API restringida	Consultas conversacionales con trazabilidad

Seguridad y trazabilidad

- Autenticación por API key con rotación configurable
- Rate limiting por IP para protección contra abuso
- SSL/TLS en todos los endpoints públicos (Let's Encrypt)

- Logs estructurados con latencia, IP y ruta para auditoría completa
- Separación de roles a nivel de base de datos (PostgreSQL RBAC)
- Caché TTL en memoria — sin datos en tránsito a terceros

Documentación técnica y demo en vivo · smart-terra.es · ruralint@smart-terra.es