

UNIVERSIDAD DE LA REPÚBLICA URUGUAY



FACULTAD DE INGENIERÍA

GISO 2.0

Extensión de Plataforma de Geocodificación Asistida y Análisis Geoespacial

Informe de Proyecto de Grado presentado por

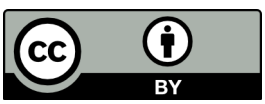
Santiago Hernández González, Lorenzo Motta Santoro

en cumplimiento parcial de los requerimientos para la graduación de la carrera de Ingeniería en Computación de Facultad de Ingeniería de la Universidad de la República

Supervisor

Raquel Sosa

Montevideo, Febrero 2026



GISO 2.0
Extensión de Plataforma de Geocodificación Asistida y Análisis Geoespacial por Santiago Hernández González, Lorenzo Motta Santoro tiene licencia CC Atribución 4.0.

Agradecimientos

Agradecemos a nuestra tutora, Raquel Sosa, por su orientación y acompañamiento a lo largo de todo el proyecto. Su guía fue fundamental para definir el rumbo del trabajo y mantener el foco en los objetivos propuestos.

Agradecemos también a los estudiantes del curso de posgrado que participaron en la validación de la plataforma, así como a los miembros de la empresa que formaron parte de las pruebas, por dedicar su tiempo a evaluar el sistema y brindar una retroalimentación valiosa.

Finalmente, a nuestras familias y amigos, por el apoyo constante durante estos meses de trabajo.

Resumen

Los datos empresariales georreferenciables, como direcciones de clientes o ubicaciones de sucursales, permiten realizar análisis de cobertura, proximidad y accesibilidad para apoyar la toma de decisiones. Sin embargo, aprovechar esta información suele requerir geocodificar direcciones masivamente y ejecutar operaciones espaciales, tareas que dependen de especialistas en Sistemas de Información Geográfica (GIS) y programación. Esta barrera limita el acceso de analistas sin formación especializada a este tipo de análisis.

Este trabajo extiende el proyecto *Plataforma de Geocodificación Asistida y Análisis Geoespacial (GISO)*, presentado en 2024 (GISO 1.0) (Sediey y Segura, 2024), y propone una nueva etapa de evolución (GISO 2.0). En la versión base se implementaron los flujos de gestión de usuarios y workspaces, carga de datos en formato tabular (por ejemplo, CSV), geocodificación asistida con proveedores abiertos, visualización en mapa y análisis espacial básico desde el navegador. A partir de esa línea base, esta etapa incorpora extensiones concretas en geocodificación, análisis espacial, usabilidad y operación.

Concretamente, el sistema incorpora validación temprana de datos y amplía la estrategia multi-fuente integrando el geocodificador de Google Maps. Asimismo, se amplía el conjunto de funciones espaciales disponibles para análisis exploratorio, como buffers, distancias punto-línea, áreas e intersecciones, y se habilita la creación de geometrías temporales directamente en el mapa para formular consultas sin preparar capas auxiliares ni alterar la base de datos.

La arquitectura mantiene el enfoque web basado en una base de datos geográfica PostgreSQL/PostGIS, un servidor de mapas GeoServer y un backend en Java con Spring Boot para la gestión de usuarios, workspaces y procesos de geocodificación asíncrona. El cliente integra Leaflet para visualización cartográfica y Turf.js para cálculos geométricos, y se documenta un despliegue web reproducible en MiNube para facilitar la operación y la evaluación en escenarios de uso reales. Si bien se mantuvo la arquitectura web, en GISO 2.0 se reestructuró en una sola aplicación de frontend lo que originalmente eran dos.

La solución se evalúa mediante pruebas funcionales, diferentes instancias de validación con usuarios y casos de estudio en dominios dispares, mostrando que reduce la dependencia de herramientas GIS de escritorio y habilita análisis geoespaciales reproducibles desde el navegador.

Palabras clave: Geocodificación, análisis espacial, GIS, SIG, GeoServer, PostGIS, Next.js, React, Turf.js, Java Spring Boot

Índice general

1. Introducción **1**
- 1.1. Motivación . . . 1
- 1.2. Objetivos . . . 2
- 1.3. Aportes principales . . . 2
- 1.4. Organización del documento . . . 3

2. Revisión de antecedentes **5**
- 2.1. Datos espaciales . . . 5
- 2.2. Geocodificación . . . 6
- 2.3. Sistemas de Información Geográfica (GIS) . . . 6
- 2.4. Arquitecturas GIS . . . 7
- 2.5. GISO 1.0 (2024) . . . 8
  - 2.5.1. Arquitectura . . . 8
  - 2.5.2. Funcionalidades principales . . . 8
- 2.6. QGIS como referencia de funcionalidades . . . 9
- 2.7. Análisis espacial . . . 10
- 2.8. Geocodificadores . . . 11
- 2.9. Computación en la nube y operación . . . 11
- 2.10. Trabajos relacionados . . . 12

3. Análisis **13**
- 3.1. Contexto . . . 13
- 3.2. Actores identificados de la plataforma . . . 13
- 3.3. Requerimientos . . . 14
  - 3.3.1. Requerimientos funcionales . . . 15
  - 3.3.2. Requerimientos no funcionales . . . 16
- 3.4. Líneas de trabajo propuestas . . . 17
- 3.5. Alcance del proyecto . . . 17

4. Diseño y arquitectura **19**
- 4.1. Modelo conceptual de la plataforma . . . 19
  - 4.1.1. Sujeto y atributos . . . 19
  - 4.1.2. Workspace, POI, sujetos y capas . . . 21
- 4.2. Diseño de la geocodificación y niveles de confianza . . . 21

4.3. Formato de datos y sistemas de referencia . . . 22
4.4. Arquitectura lógica de la solución . . . 22
4.4.1. Capas del backend . . . 23
4.4.2. Integraciones externas . . . 25
4.5. Evolución arquitectónica . . . 25

5. Implementación **27**
5.1. Tecnologías utilizadas . . . 27
5.2. Implementación del backend . . . 29
5.2.1. Estructura de módulos y organización del código . . . 29
5.2.2. Carga y validación de datos . . . 30
5.2.3. Geocodificación multi-fuente y cola de mensajes . . . 31
5.2.4. Gestión de workspaces, sujetos, POIs y funciones geográficas 32
5.3. Implementación del frontend . . . 32
5.3.1. Gestión de estado y comunicación con el backend . . . 33
5.3.2. Mapa, capas y filtros geográficos . . . 34
5.3.3. Geometrías temporales y funciones espaciales en el cliente 34
5.3.4. Exportación de mapas y soporte para distintos SRID . . . 35
5.4. Despliegue y configuración . . . 36

6. Pruebas y validación **39**
6.1. Pruebas funcionales y técnicas . . . 39
6.2. Validación con usuarios . . . 40
6.2.1. Taller con estudiantes de posgrado . . . 40
6.2.2. Uso por parte de empresa del medio . . . 46
6.2.3. Síntesis de la validación con usuarios . . . 46

7. Casos de estudio **47**
7.1. Caso 1: Planificación de un centro de distribución logística . . . 47
7.1.1. Contexto del caso de estudio . . . 47
7.1.2. Datos del negocio . . . 48
7.1.3. Ejecución del caso en la aplicación . . . 49
7.1.4. Análisis de resultados . . . 50
7.2. Caso 2: Análisis geoespacial en sanidad vegetal (Picudo Rojo) . . . 51
7.2.1. Escenario y datos . . . 51
7.2.2. Flujo de análisis epidemiológico . . . 51
7.2.3. Resultados del caso . . . 51

8. Conclusiones y trabajo futuro **53**
8.1. Gestión del proyecto . . . 53
8.2. Conclusiones . . . 55
8.3. Mejoras y trabajo futuro . . . 56

Referencias **59**

Capítulo 1

Introducción

Este capítulo ofrece una visión general del proyecto, incluyendo motivación, objetivos, aportes principales y la organización del documento.

1.1. Motivación

Las organizaciones actuales gestionan grandes volúmenes de datos que resultan fundamentales para la toma de decisiones y el desarrollo de estrategias. Entre estos datos empresariales, la información georreferenciable (como direcciones de clientes, proveedores, sucursales o eventos de interés) permite optimizar la gestión de recursos mediante análisis de localización, cobertura, accesibilidad y proximidad.

Los servicios de geocodificación asignan coordenadas geográficas a direcciones mediante el proceso de geocodificación. Sin embargo, estos servicios procesan habitualmente una dirección a la vez, lo que constituye una limitación significativa al transformar grandes volúmenes de datos. Para gestionar este tipo de tareas masivas, se requiere que un programador desarrolle procesos automatizados capaces de geocodificar múltiples direcciones de manera eficiente.

A su vez, el aprovechamiento de datos georreferenciados requiere análisis espaciales con manejo de capas geográficas y la ejecución de operaciones geométricas complejas. Estas tareas demandan el uso de Sistemas de Información Geográfica (GIS), los cuales exigen conocimientos técnicos avanzados. Esta necesidad de especialización limita la capacidad de analistas no expertos en GIS para realizar estudios geoespaciales de forma autónoma.

El proyecto de grado *Plataforma de Geocodificación Asistida y Análisis Geoespacial (GISO 1.0)* implementó una primera instancia funcional de una plataforma web para la carga de datos tabulares, la geocodificación asistida, el análisis de los datos y la visualización en mapa (Sediey y Segura, 2024). Sin embargo, el análisis de GISO 1.0 evidenció limitaciones que restringen su adopción en escenarios reales y su uso autónomo por analistas no expertos en GIS.

Ante estas limitaciones, se identificó la oportunidad de ampliar la estrategia

de geocodificación, de incorporar un repertorio más amplio de operaciones de análisis espacial y de mejorar la experiencia de usuario mediante flujos de navegación más claros, validaciones tempranas y mensajes más descriptivos. Estas necesidades justifican una nueva etapa de evolución de la plataforma.

En este informe se denomina a esta etapa como *GISO 2.0*.

1.2. Objetivos

Objetivo general

Se plantea como objetivo general continuar la evolución de GISO (GISO 2.0) mediante la expansión de los geocodificadores disponibles y de la capacidad de análisis espacial para analistas no expertos en GIS. Asimismo, se busca habilitar un despliegue web en MiNube que permita el acceso a usuarios potenciales y facilite la validación de la plataforma.

Objetivos específicos

1. Estudiar el dominio de aplicación y trabajos previos en la temática.
2. Definir requerimientos funcionales y no funcionales, así como un plan de validación con usuarios.
3. Optimizar la arquitectura de la plataforma, priorizando la mantenibilidad.
4. Extender la estrategia de geocodificación incorporando geocodificadores adicionales.
5. Implementar funcionalidades adicionales que potencien el análisis espacial y mejoren la experiencia de uso.
6. Definir y ejecutar el despliegue web de la plataforma en MiNube, documentando un procedimiento reproducible de configuración y operación.
7. Validar la plataforma con usuarios potenciales mediante casos de estudio y métricas de usabilidad y desempeño.

1.3. Aportes principales

Los aportes principales de este proyecto se organizan en tres ejes:

- **Evolución funcional**: extensión de la estrategia de geocodificación mediante la incorporación de geocodificadores adicionales y ampliación del conjunto de funciones de análisis espacial disponibles desde el mapa, manteniendo la compatibilidad con PostGIS y GeoServer.

- **Operación, optimización y despliegue**: definición de un despliegue web reproducible en MiNube, junto con documentación técnica y guías de operación que faciliten la instalación, configuración, mantenimiento y desarrollo local. Optimización de la arquitectura para facilitar el mantenimiento a futuro del sistema.
- **Validación y caso de estudio**: ejecución de pruebas funcionales, instancias de validación con usuarios potenciales, complementadas con un caso de estudio que permita ilustrar flujos de trabajo realistas de carga, geocodificación y análisis. También se presentó el sistema en Ingeniería de-Muestra (IdM) 2025, donde se mantuvieron conversaciones con usuarios interesados.

Como entregables, el proyecto produce un prototipo web funcional de la plataforma (frontend, backend y base de datos espacial) desplegado en MiNube con procedimiento de instalación documentado y manual de usuario asociado, junto con el código fuente, la documentación técnica que describe la arquitectura, la guía de despliegue y los manuales de la aplicación.

1.4. Organización del documento

Este informe se estructura en los siguientes capítulos:

- **Capítulo 1: Introducción**. Presenta la motivación del proyecto, los objetivos y los principales aportes.
- **Capítulo 2: Revisión de antecedentes**. Se presentan los conceptos teóricos relevantes, los sistemas y las herramientas utilizados en GISO 1.0, así como un análisis del estado del arte.
- **Capítulo 3: Análisis**. Describe el contexto y alcance de este proyecto, los actores involucrados, los requerimientos funcionales y no funcionales, y las líneas de trabajo que guían la evolución de la plataforma.
- **Capítulo 4: Diseño y arquitectura**. Detalla el modelo conceptual, las decisiones de geocodificación y sistemas de referencia, la arquitectura lógica de la solución y la evolución respecto de GISO 1.0.
- **Capítulo 5: Implementación**. Describe cómo se materializan los componentes de backend y frontend, las integraciones con servicios externos y las nuevas capacidades de análisis espacial desde el punto de vista técnico.
- **Capítulo 6: Pruebas y validación**. Resume las pruebas funcionales, técnicas y de rendimiento realizadas, así como la instancia de validación de la experiencia de usuario.
- **Capítulo 7: Casos de estudio**. Presenta dos casos de estudio: uno aplicado al dominio de logística de distribución, utilizando datos de sucursales

de supermercados, y otro al monitoreo de sanidad vegetal con datos del Picudo Rojo, ilustrando los flujos de trabajo de la plataforma en dominios dispares.

- **Capítulo 8: Conclusiones y trabajo futuro**. Evalúa el cumplimiento de los objetivos, analiza las dificultades encontradas y plantea líneas de trabajo futuro.
- **Referencias**. Contiene la bibliografía utilizada a lo largo del documento.

Capítulo 2

Revisión de antecedentes

El presente capítulo examina los conceptos, herramientas, estándares y sistemas relevantes para el desarrollo de la *Plataforma de Geocodificación Asistida y Análisis Geoespacial (GISO)* (Sediey y Segura, 2024). La revisión consolida el marco teórico-práctico sobre el cual se analiza la versión base (GISO 1.0) y se definen las extensiones propuestas en este proyecto.

2.1. Datos espaciales

Los datos espaciales representan información sobre la localización y las características de fenómenos geográficos (por ejemplo, sujetos, puntos de interés o límites administrativos). Pueden representarse de dos maneras principales: vectorial y ráster. Los datos vectoriales describen entidades discretas (por ejemplo, puntos de interés, tramos de ruta o polígonos de departamentos), mientras que los datos ráster modelan campos continuos (como modelos de elevación o mapas de temperatura) mediante una malla de celdas (Bernabé Poveda y López Vázquez, 2012).

En el contexto de este proyecto se trabaja con datos vectoriales, en los que cada entidad se describe mediante una geometría de tipo *punto*, *línea* o *polígono*, definida a partir de vértices en coordenadas X, Y (y opcionalmente Z).

Para ubicar estas geometrías sobre la superficie terrestre se utilizan Sistemas de Referencia de Coordenadas (SRC), que definen coordenadas para representar posiciones sobre la superficie terrestre. Cada SRC se identifica de forma unívoca mediante un código SRID (*Spatial Reference Identifier*); el registro más utilizado es el mantenido por la EPSG (*European Petroleum Survey Group*), administrado por la *International Association of Oil & Gas Producers* (IOGP)[1]. En Uruguay se emplea de forma recurrente WGS84 (`EPSG:4326`) para coordenadas geográficas y UTM 21S (`EPSG:32721`) para coordenadas proyectadas en

[1] Los códigos EPSG son identificadores numéricos que designan de forma unívoca un sistema de referencia de coordenadas o una proyección cartográfica.

metros; en zonas limítrofes puede emplearse UTM 22S (`EPSG:32722`) (EPSG.io, 2022c, 2022a, 2022b).

2.2. Geocodificación

La geocodificación consiste en la transformación de descripciones textuales de ubicaciones (por ejemplo, direcciones o nombres de lugares) en coordenadas geográficas que puedan ser utilizadas posteriormente en análisis espacial y visualización en mapas (Goldberg, Wilson, y Knoblock, 2007; Karimi y Sharker, 2019). Un proceso típico incluye: (i) normalización y estandarización del texto de la dirección; (ii) búsqueda de coincidencias con una base de datos de referencia (calles, padrones, puntos de interés) y (iii) asignación de coordenadas y metadatos de calidad.

Un geocodificador es un servicio o herramienta que implementa el proceso de geocodificación, permitiendo transformar direcciones textuales en coordenadas geográficas mediante algoritmos de búsqueda y comparación sobre bases de datos de referencia.

La calidad de un geocodificador suele evaluarse con dos tipos de indicadores: por un lado la *cobertura*, que mide el porcentaje de direcciones que logran ubicarse; y por otro el *error posicional*, entendido como la distancia entre la posición estimada y una posición de referencia considerada correcta (por ejemplo, una medición en campo con GPS o un dato oficial) (Karimi y Sharker, 2019).

En el contexto uruguayo, esta evaluación se ve afectada por la heterogeneidad de los formatos de dirección, la existencia de nomenclaturas locales y la disponibilidad de fuentes oficiales como el Sistema Único de Direcciones del Uruguay (SuDir) (Infraestructura de Datos Espaciales del Uruguay, 2025).

2.3. Sistemas de Información Geográfica (GIS)

Los Sistemas de Información Geográfica (GIS) integran datos, software, hardware y procesos para cargar, guardar, analizar y visualizar información georreferenciada (Longley, Goodchild, Maguire, y Rhind, 2011). Estos sistemas permiten representar fenómenos del mundo real en forma de capas temáticas (por ejemplo, sujetos, puntos de interés, límites administrativos, redes viales) y aplicar operaciones espaciales para apoyar la toma de decisiones.

En la práctica, se destacan aplicaciones de GIS en planificación urbana, epidemiología, transporte y análisis de servicios (Zhang, Luo, y Xu, 2021). En todos los casos, la capacidad de combinar datos espaciales de distintas fuentes, aplicar modelos analíticos y desplegar resultados mediante mapas interactivos es fundamental.

En el dominio empresarial, los GIS se aplican a múltiples propósitos: estudiar la cobertura de redes de sucursales, definir zonas de influencia, apoyar decisiones de localización, analizar tiempos de viaje y evaluar riesgos asociados a fenómenos ambientales o socioeconómicos (Longley y cols., 2011). La plataforma

GISO sigue esta línea, focalizada en el análisis de datos empresariales georreferenciables para consultas de proximidad, accesibilidad y cobertura sobre el territorio uruguayo. En capítulos posteriores se profundiza en estos casos de uso y en cómo las funcionalidades de geocodificación y análisis espacial responden a necesidades concretas de diferentes organizaciones.

2.4. Arquitecturas GIS

La transición desde GIS de escritorio hacia arquitecturas web y distribuidas ha llevado a modelos basados en servicios, donde la arquitectura desacopla almacenamiento, procesamiento y visualización en componentes especializados (Huang y Shao, 2020). En este enfoque, la base de datos espacial gestiona la persistencia de geometrías y atributos (por ejemplo, PostGIS), mientras que servidores de mapas como GeoServer exponen servicios estándar del Open Geospatial Consortium (OGC), como Web Map Service (WMS), Web Feature Service (WFS) y Web Map Tile Service (WMTS) ("GeoServer User Manual", 2025).

Sobre estos servicios se construyen clientes web interactivos utilizando librerías de visualización como Leaflet, una librería JavaScript de código abierto para mapas interactivos ("Leaflet Reference (v1.9)", 2025).

Un ejemplo habitual de esta arquitectura comprende: (i) una base de datos espacial que almacena capas y geometrías; (ii) un backend que expone servicios de dominio; (iii) un servidor de mapas que publica capas y estilos; y (iv) un cliente web que ofrece interfaces de visualización y análisis. Este diseño facilita la escalabilidad, la interoperabilidad con otros sistemas y la evolución progresiva de funcionalidades sin comprometer la base de datos subyacente. La Figura 2.1 ilustra esta arquitectura, que la plataforma GISO instancia con las tecnologías concretas descritas en la Sección 2.5.

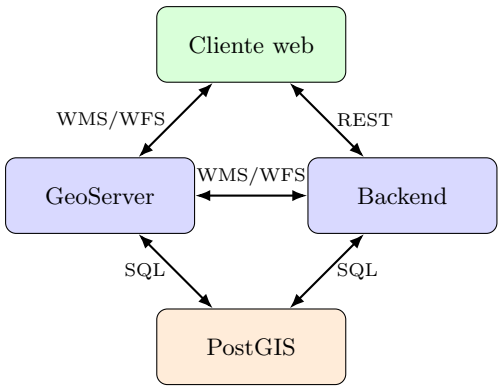


Figura 2.1: Arquitectura GIS web basada en base de datos espacial, servidor de mapas y cliente web.

2.5. GISO 1.0 (2024)

La primera versión de la plataforma se concibió como una aplicación web orientada a analistas y administradores que necesitan geocodificar direcciones de forma masiva y realizar análisis geoespaciales básicos sin recurrir a herramientas de escritorio (Sediey y Segura, 2024), cuyo flujo general se visualiza en la Figura 2.2. Desde el punto de vista arquitectónico, GISO 1.0 siguió un enfoque web basado en una base de datos geográfica y un servidor de mapas.

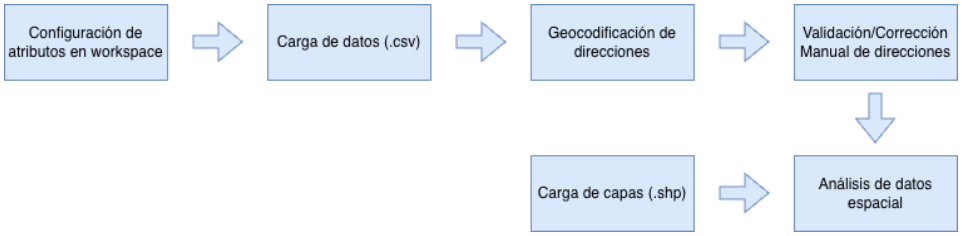


Figura 2.2: Flujo general de carga, geocodificación y análisis en la plataforma GISO 1.0.

2.5.1. Arquitectura

GISO 1.0 se compuso de cuatro bloques principales: (i) una base de datos PostgreSQL con PostGIS para persistir geometrías y atributos; (ii) un backend en Java responsable de exponer una API REST, administrar usuarios y workspaces y orquestar la geocodificación; (iii) GeoServer como servidor de mapas; y (iv) un cliente web para la interacción con usuarios (Sediey y Segura, 2024). En esa versión existían dos aplicaciones web diferenciadas por rol: una interfaz orientada a analistas y otra orientada a administradores. En la Figura 2.3 se puede observar la arquitectura descrita.

2.5.2. Funcionalidades principales

En términos funcionales, GISO 1.0 cubrió los siguientes flujos:

- **Gestión de workspaces y usuarios:** el administrador crea workspaces y configura los atributos relevantes del dominio, habilitando que distintos equipos trabajen de forma aislada.
- **Carga de datos georreferenciables:** el analista carga datos en formato tabular (por ejemplo, archivos CSV) para su posterior geocodificación y visualización.
- **Geocodificación asistida multi-fuente:** la plataforma integra más de un geocodificador y registra metadatos de calidad (por ejemplo, nivel de confianza) para apoyar la revisión y corrección de resultados (Sediey y Segura, 2024).

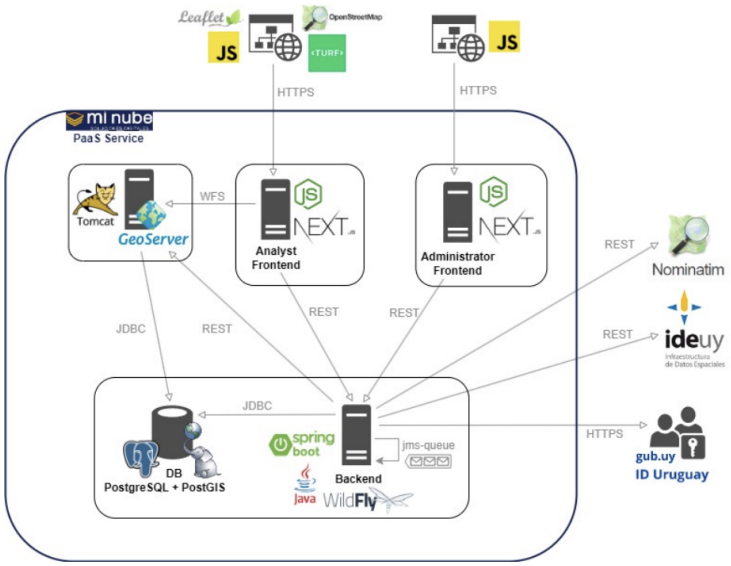


Figura 2.3: Diagrama de arquitectura de GISO 1.0.

- **Visualización en mapa:** se visualizan sujetos y puntos de interés (POI) como capas geográficas, permitiendo explorar y filtrar información espacial.
- **Análisis espacial básico:** se habilitan consultas y mediciones iniciales que funcionan como base para extender la capacidad analítica en etapas posteriores (Sediey y Segura, 2024).

2.6. QGIS como referencia de funcionalidades

Además de los componentes específicos utilizados en la plataforma, resulta relevante considerar herramientas GIS consolidadas que sirven de referencia para el diseño de funcionalidades. QGIS es un sistema de información geográfica de código abierto ampliamente utilizado en ámbitos académicos, gubernamentales y empresariales (QGIS Development Team, 2024). Proporciona un conjunto muy completo de herramientas de geoprocesamiento y un modelo de trabajo basado en capas y proyectos que son estándar en los software GIS especializados.

Específicamente, el panel de herramientas vectoriales de QGIS ofrece operaciones de geoprocesamiento que abarcan desde cálculos geométricos básicos (área, perímetro) hasta análisis de proximidad y superposición de capas. Para GISO 2.0 se tomaron como referencia las operaciones más habituales de ese panel: cálculo de área y perímetro de polígonos, generación de buffers (zonas de influencia), análisis de vecinos más cercanos e intersecciones espaciales. Estas

operaciones se adaptaron a un contexto web multiusuario, con el objetivo de ofrecer capacidades similares a las de un GIS de escritorio pero accesibles desde un navegador, sin requerir instalación de software especializado ni conocimientos avanzados en GIS.

2.7. Análisis espacial

Las funciones de análisis espacial permiten generar nueva información a partir de la relación entre geometrías y atributos (Longley y cols., 2011). Entre las operaciones más habituales se encuentran el cálculo de distancias, la generación de áreas de influencia (buffers), el análisis de proximidad entre conjuntos de puntos y la intersección entre capas geográficas. Estas operaciones constituyen la base de los análisis de cobertura, accesibilidad y zonificación que la plataforma GISO busca ofrecer.

En GISO 1.0 se implementaron funcionalidades básicas como la medición de distancia punto-punto y el conteo de sujetos dentro de un polígono seleccionado (Sediey y Segura, 2024). Algunas otras funcionalidades usuales en los sistemas de información geográficos que son de interés para el proyecto son:

- **Área y perímetro.** Calcula la superficie y el contorno de un polígono seleccionado. Requiere como entrada una geometría de tipo polígono (capa existente o geometría temporal). Resulta útil para dimensionar predios, zonas de cobertura o áreas de influencia.

- **Buffer.** Genera una zona de influencia (polígono) alrededor de una geometría de punto, línea o polígono, a una distancia configurable por el usuario. Permite modelar radios de cobertura, áreas de riesgo o zonas de exclusión.

- **Distancia punto-línea.** Calcula la distancia mínima entre un punto (o conjunto de puntos) y una capa de líneas. Requiere una capa de puntos y una capa de líneas como entrada. Se utiliza, por ejemplo, para evaluar la proximidad de sujetos a redes viales o cursos de agua.

- **Análisis de proximidad (N vecinos).** Para cada punto de una capa, identifica los N vecinos más cercanos dentro de la misma capa o de otra capa de puntos, calculando la distancia a cada uno. Permite detectar agrupaciones, evaluar densidades locales y planificar rutas de reparto.

- **Intersección espacial.** Determina qué entidades de una capa se encuentran dentro de un polígono dado (buffer, geometría temporal o capa de polígonos). Permite responder preguntas del tipo "¿qué sujetos quedan dentro de esta zona?" o "¿qué departamentos cubre este radio de operación?".

2.8. Geocodificadores

Los diferentes geocodificadores abarcan distintos tipos de soluciones: comerciales, de código abierto y auto-gestionadas. Cada tipo presenta fortalezas y limitaciones específicas en términos de cobertura geográfica, calidad de resultados, licenciamiento y costo (Karimi y Sharker, 2019). En el proyecto GISO se adopta una estrategia multi-fuente que integra varios proveedores, con el objetivo de mejorar la cobertura sobre el territorio uruguayo y reducir la dependencia de una única fuente (Sediey y Segura, 2024).

En GISO 1.0 se incorporaron dos geocodificadores principales:

- **IDEuy/SuDir**. Servicio oficial basado en el Sistema Único de Direcciones del Uruguay, ofrece alta calidad y coherencia con cartografía oficial para direcciones nacionales (Infraestructura de Datos Espaciales del Uruguay, 2025).
- **Nominatim**. Geocodificador abierto construido sobre datos de OpenStreetMap, con amplia cobertura global y licenciamiento abierto, aunque con variaciones de calidad según la densidad de contribuciones ("Nominatim – Search API", 2025).

En este proyecto se estudia el geocodificador de **Google Maps** (Google, 2025), utilizado como fuente complementaria para mejorar la cobertura y precisión en zonas donde los servicios abiertos presentan vacíos o errores posicionales.

La Tabla 2.1 resume las principales características de los geocodificadores considerados.

Tabla 2.1: Comparación de geocodificadores estudiados.

Servicio	**Cobertura**	**Costo**	**Limitaciones**
IDEuy/SuDir	Uruguay (oficial)	Gratuito	Solo direcciones nacionales; dependencia de disponibilidad del servicio
Nominatim	Global (OpenStreetMap)	Gratuito	Calidad variable según densidad de contribuciones
Google Maps	Global	Pago por uso	Requiere clave API y tiene límites de cuota

2.9. Computación en la nube y operación

La computación en la nube proporciona un modelo de provisión de recursos de cómputo, almacenamiento y red bajo demanda, con elasticidad y pago por uso. En el ámbito de los sistemas geoespaciales, el despliegue en la nube facilita la

ejecución de cargas de trabajo intensivas en datos y procesamiento, habilitando escenarios de GIS en tiempo real, análisis distribuido y servicios de mapas de alta disponibilidad (Huang y Shao, 2020).

MiNube (minubeantel.uy) es la plataforma de servicios cloud de Antel (Antel, 2025b), que ofrece distintas modalidades de despliegue; entre ellas, Elastic Cloud, un servicio de plataforma como servicio (PaaS) basado en Jelastic (Antel, 2025a).

2.10. Trabajos relacionados

El proyecto de grado previo en el que se basa este trabajo realizó una revisión detallada de herramientas GIS y plataformas relacionadas, que constituye la base de la presente síntesis (Sediey y Segura, 2024). En concreto, se analizaron visualizadores y plataformas web de datos espaciales disponibles en Uruguay (como el GeoPortal del MTOP, el visualizador del Ministerio de Ambiente, MonteviMap y el visualizador de la IDEuy), relevando sus capacidades de visualización, medición, descarga de datos y geocodificación puntual (de Datos Espaciales del Uruguay, 2024).

Asimismo, se revisaron aplicaciones de GIS y análisis geoespacial reportadas en la literatura científica, con énfasis en estudios de epidemiología y salud pública que ilustran el potencial de las técnicas espaciales para apoyar decisiones basadas en evidencia (Zhang y cols., 2021). Finalmente, se compararon diversas plataformas de geocodificación, evaluando cobertura, calidad y facilidad de integración para direcciones en Uruguay, lo que orientó el análisis de IDEuy, Nominatim y Google Maps como posibles proveedores.

Capítulo 3

Análisis

Este capítulo describe el contexto, alcance y requerimientos del proyecto, con el objetivo de fundamentar las decisiones de diseño y la planificación de la evolución de la plataforma.

3.1. Contexto

Este proyecto (GISO 2.0) se inscribe como una extensión del proyecto GISO (Sediey y Segura, 2024). GISO 1.0 consistió en una plataforma web para la georreferenciación, visualización y análisis de datos empresariales, presentada en 2024 como primera instancia funcional de la aplicación. Esa versión se toma como línea base para la presente evolución.

Concretamente, GISO 1.0 validó los flujos de gestión de usuarios y workspaces, la carga de datos tabulares (por ejemplo, CSV), la geocodificación asistida multi-fuente con proveedores abiertos, la visualización en mapa y un conjunto acotado de operaciones de análisis espacial.

3.2. Actores identificados de la plataforma

En cuanto a los actores del sistema, estos se mantienen con respecto a GISO 1.0, ya que el modelo de interacción no ha cambiado. Se identifican tres perfiles:

- **Super Administrador:** usuario con privilegios globales sobre la plataforma. Tiene acceso total al sistema y puede administrar todos los workspaces, usuarios, configuraciones y recursos, sin restricción por pertenencia a un workspace específico.

- **Administrador:** encargado de la gestión general del sistema, incluyendo el alta de usuarios, configuración de los diferentes workspaces, capas y atributos. También tiene acceso a funcionalidades analíticas.

- **Analista:** usuario con acceso a la carga (archivos CSV y *shapefiles*), visualización, consulta y análisis de los datos cargados. Puede aplicar filtros, utilizar operaciones espaciales (como distancia punto-punto, buffer, distancia punto-línea, área y perímetro, análisis de proximidad e intersección espacial), seleccionar y exportar resultados, pero no tiene acceso a configuraciones administrativas.

Regla de permisos por rol y pertenencia: el control de acceso se define por rol y pertenencia al workspace. El Super Administrador tiene alcance global. El Administrador solo puede crear, visualizar, editar y eliminar recursos del o de los workspaces a los que administra. El Analista opera sobre los datos y herramientas disponibles en su workspace sin permisos de administración.

La Figura 3.1 sintetiza los actores operativos principales y los casos de uso más representativos asociados a cada perfil. Se trata de un diagrama resumido que destaca los flujos principales; los casos de uso completos se derivan de los requerimientos detallados en la Sección 3.3.1.

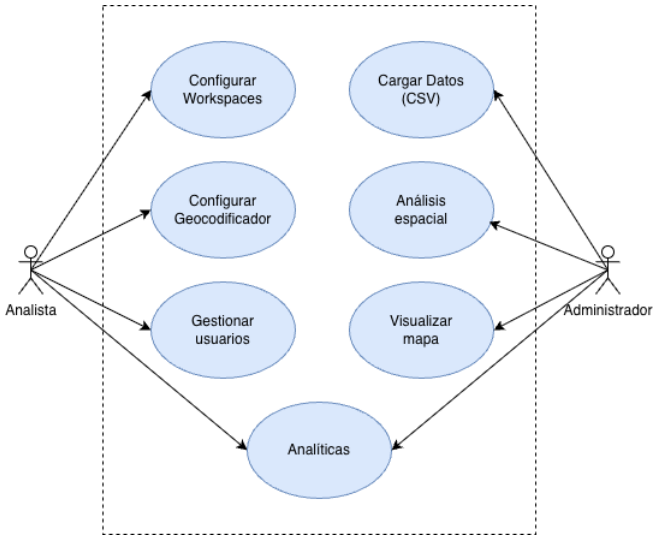


Figura 3.1: Actores principales y casos de uso de la plataforma.

3.3. Requerimientos

El análisis de GISO 1.0 evidencia que, además de incorporar nuevas capacidades de geocodificación y análisis, es necesario reformular los requerimientos con un foco explícito en usabilidad y mantenibilidad. Entre los problemas concretos identificados se destacan:

- Textos que asumían conocimiento previo de GIS.

- Navegación sin opciones claras de retroceso.
- Mensajes de error genéricos sin indicar la causa.
- Ausencia de confirmaciones visuales y de indicadores de progreso.
- Falta de diálogos de confirmación en operaciones destructivas.
- Existencia de dos aplicaciones web separadas (analista y administrador) con lógica y componentes casi idénticos, lo que genera duplicación de código y esfuerzo de mantenimiento.

A continuación se resumen los requerimientos que guían la extensión de la plataforma, centrados en lo que se agrega o rediseña.

3.3.1. Requerimientos funcionales

- **Incorporación de nuevos servicios de geocodificación:** el sistema debe permitir la geocodificación de direcciones utilizando nuevos proveedores.
- **Validación robusta de datos CSV:** se debe permitir la carga de archivos CSV de sujetos y atributos. El sistema verifica que el archivo no esté vacío, utiliza la primera fila como encabezado, comprueba la existencia de la columna obligatoria definida como identificador de sujeto y mapea automáticamente las columnas a atributos del workspace. Las columnas adicionales se ignoran y los atributos faltantes se rellenan con valores nulos. La validación falla ante errores de lectura o ausencia de identificadores.
- **Funciones geográficas ampliadas:** el sistema debe incorporar las funciones espaciales descritas en la Sección 2.7 (área y perímetro, buffer, distancia punto-línea, análisis de proximidad e intersección espacial). Estas operaciones deben poder ejecutarse tanto sobre capas de sujetos y POIs como sobre capas cargadas mediante *shapefiles*.
- **Gestión de geometrías temporales en el mapa:** el sistema debe permitir crear y utilizar geometrías temporales dibujadas sobre el mapa como insumo de análisis espacial. Esta funcionalidad debe habilitar su uso en operaciones como buffer, análisis de proximidad y distancia punto-línea, según corresponda.
- **Gestión de datos para analistas:** el perfil analista debe disponer de vistas detalladas de sujetos y POIs, con acceso a sus atributos y ubicación geográfica, así como capacidades de edición de datos guardados (atributos, latitud, longitud y SRID) y actualización en tiempo real de resultados de geocodificación.
- **Gestión de workspaces para administradores:** el administrador debe poder gestionar capas asociadas a cada workspace, personalizar colores de

las mismas, configurar atributos visibles en el mapa, configurar el geocodificador para el workspace y definir parámetros del workspace como capa base y SRID por defecto. Todas estas operaciones deben limitarse a los workspaces a los que el administrador pertenece.

- **Flujos de gestión de usuarios:** la aplicación debe permitir crear usuarios asociados obligatoriamente a un workspace, garantizar que los administradores queden vinculados al workspace que crean, y restringir la visualización y edición de usuarios y reportes a los workspaces propios del administrador.
- **Flujo de activación de cuenta:** el sistema debe enviar un correo con enlace temporal para establecer la contraseña inicial.

3.3.2. Requerimientos no funcionales

- **Usabilidad para usuarios no expertos en GIS:** la interfaz debe ser simple, descriptiva y manejar mensajes comprensibles para usuarios que no manejan conceptos técnicos de GIS, evitando jerga innecesaria y destacando el impacto de las operaciones.
- **Navegación clara y reversible:** el sistema debe ofrecer una estructura de navegación simple, con rutas bien definidas y opciones de navegación para retroceder en cualquier flujo de trabajo, preservando el contexto actual y la información ingresada por el usuario para evitar frustraciones y facilitar la exploración de la plataforma.
- **Feedback explícito de errores y éxito:** ante cualquier operación (carga de datos, análisis, configuración) el sistema debe mostrar mensajes de error claros que indiquen la causa (incluyendo problemas de permisos) y mensajes de confirmación visibles cuando una operación se complete correctamente (por ejemplo, mediante *toasts*).
- **Visibilidad del estado del sistema:** las acciones que implican procesamiento deben acompañarse de indicadores visuales (*loaders*, deshabilitado de botones) que informen al usuario que la operación está en curso y prevengan acciones repetidas o inconsistentes.
- **Ejecución asincrónica de tareas intensivas:** las operaciones de geocodificación y análisis espacial deben ejecutarse de forma asincrónica, evitando bloqueos de interfaz y permitiendo que el usuario continúe trabajando mientras se procesan colas de direcciones u operaciones sobre grandes volúmenes de datos.
- **Consistencia y reutilización en la interfaz:** la aplicación debe ofrecer patrones de interacción coherentes para crear, editar y eliminar entidades, reutilizando componentes y lógica de código en lugar de mantener variantes duplicadas.

- **Simplificación de arquitectura:** en lugar de mantener aplicaciones separadas para administración y análisis, la solución debe ofrecer una única aplicación frontend, donde el acceso a rutas y funcionalidades se controle exclusivamente a través de permisos, reduciendo la duplicación de código y facilitando la evolución del sistema.

3.4. Líneas de trabajo propuestas

A partir de los problemas detectados y de los requerimientos definidos, se plantean líneas de trabajo que guían el diseño de la solución y delimitan el alcance técnico:

- **Despliegue web en MiNube.** Se define un procedimiento reproducible de instalación, configuración y operación de los componentes principales de la plataforma (base de datos PostGIS, GeoServer, backend y cliente web), con foco en facilitar su uso sostenido y su validación con usuarios.
- **Expansión de geocodificadores.** Se propone extender el enfoque multifuente, permitiendo configurar y seleccionar geocodificadores por workspace según necesidades de cobertura, precisión y costo.
- **Ampliación de análisis espacial.** Se busca extender el conjunto de funciones disponibles desde el mapa, priorizando operaciones frecuentes y parametrizables para analistas no expertos en GIS.
- **Usabilidad y mantenibilidad.** Se planifica mejorar flujos de navegación, validaciones y mensajes de la interfaz, y reducir duplicación de código en el frontend para facilitar la evolución de la plataforma.

En conjunto, los requerimientos funcionales y no funcionales descritos se concretan en los casos de uso representados en la Figura 3.1 y se abordan a través de las líneas de trabajo aquí planteadas.

3.5. Alcance del proyecto

Sobre la línea base anteriormente descrita, GISO 2.0 incorpora extensiones en geocodificación, análisis espacial, usabilidad y operación, como se detalla a continuación.

En términos de alcance, este proyecto incluye:

- La incorporación del geocodificador de Google Maps como servicio adicional, complementando los servicios existentes en la plataforma (IDEuy y Nominatim).
- La ampliación de operaciones espaciales disponibles desde el mapa, tales como distancia punto-línea, proximidad, buffers e intersecciones.

- La extensión de capacidades de visualización y análisis, incluyendo ajustes de sistema de referencia de coordenadas, selección de capa base y exportación de mapas.
- La mejora de la experiencia de usuario mediante validaciones, flujos de navegación más claros y mensajes de error y confirmación más descriptivos.
- La simplificación de la arquitectura del frontend para reducir duplicación de código y facilitar el mantenimiento.
- El despliegue web de la plataforma en MiNube, contemplando un procedimiento reproducible de instalación, configuración y operación.

Capítulo 4

Diseño y arquitectura

Este capítulo presenta el diseño conceptual y la arquitectura de GISO. En primer lugar se resume la propuesta de solución de GISO 1.0 (Sediey y Segura, 2024), destacando las decisiones que se mantienen como base. A continuación se describen las extensiones y ajustes incorporados en esta fase, tanto a nivel de modelo de datos como de arquitectura lógica y de integración con servicios externos.

4.1. Modelo conceptual de la plataforma

El diseño conceptual de GISO mantiene la plataforma independiente del dominio de negocio, de forma que pueda adaptarse a distintos contextos (salud, logística, gestión pública, entre otros) sin requerir cambios estructurales en la base de datos. Para ello se utilizan entidades genéricas que representan *sujetos*, *puntos de interés (POI)*, capas y atributos, desacoplando la estructura de almacenamiento de la semántica específica de cada caso de estudio. La Figura 4.1 presenta el modelo conceptual utilizado para organizar la información en workspaces, sujetos, POIs, capas y atributos.

4.1.1. Sujeto y atributos

En GISO 1.0 se evaluaron dos enfoques para modelar los atributos asociados a sujetos y eventos: un modelo con atributos fijos por columna y un modelo con atributos variables almacenados en tablas de pares clave-valor. El primer enfoque simplifica las consultas, pero obliga a realizar modificaciones estructurales en la base de datos (`ALTER TABLE`) cada vez que se incorporan nuevos atributos, lo que no resulta recomendable en escenarios de producción. El segundo enfoque, en cambio, permite definir atributos dinámicos y asumir un costo mayor en las consultas a cambio de flexibilidad y escalabilidad.

Finalmente, en GISO 1.0 se adoptó el enfoque de atributos variables, unificando la noción de sujeto y evento en una única entidad lógica `Subject`. Esta

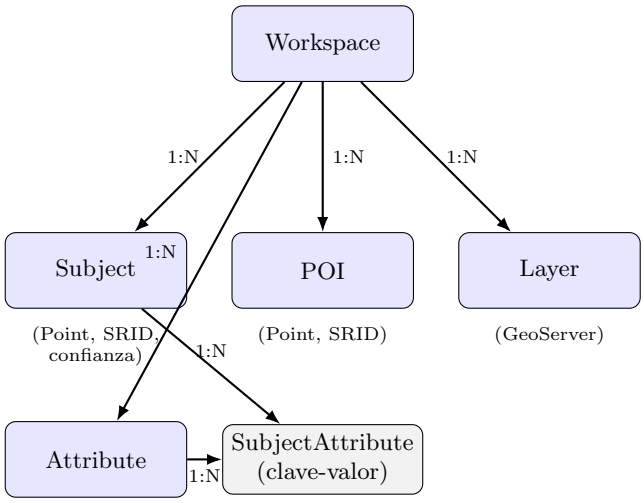


Figura 4.1: Modelo conceptual de sujetos, atributos, workspaces, puntos de interés y capas en GISO.

decisión se mantiene en GISO 2.0. La elección prioriza la flexibilidad del modelo sobre la eficiencia de las consultas, lo cual resulta adecuado para una plataforma multidominio donde la estructura de datos varía entre casos de uso. Cada fila del archivo CSV cargado representa un dato a georreferenciar y sus atributos se almacenan en una tabla de atributos de sujeto, donde cada registro contiene el identificador del sujeto, el nombre lógico del atributo y su valor. De este modo:

- Se reduce la necesidad de manejar múltiples archivos CSV relacionados por identificadores.
- Se habilita la incorporación de nuevos atributos sin cambios estructurales en la base de datos.

Para mantener un mínimo de estructura se utiliza una capa de *metadatos* de atributos. La tabla de atributos define, para cada workspace, el conjunto de atributos disponibles, su tipo de dato, si representan información de dirección y si identifican de forma única al sujeto. Esta capa permite:

- Configurar un atributo obligatorio como identificador del sujeto.
- Marcar los atributos que intervienen en la geocodificación (por ejemplo, calle, número, localidad, departamento).
- Controlar qué atributos se exponen en las vistas de análisis y en las capas geográficas.

En la entidad `Subject` se almacenan los aspectos geográficos de cada sujeto geocodificado. La geometría se representa como un punto (`POINT`) con un sistema de referencia de coordenadas (`SRID`) explícito y un nivel de confianza en

la geocodificación. Además, la entidad incluye el estado del sujeto (pendiente, activo, etc.) y metadatos de creación.

4.1.2. Workspace, POI, sujetos y capas

El concepto de *workspace* actúa como unidad de organización y aislamiento lógico dentro de la plataforma. Cada workspace agrupa:

- Su configuración de geocodificador preferido y de sistema de referencia por defecto.
- Sus propios usuarios.
- Sus puntos de interés (POIs), que representan ubicaciones relevantes para el dominio (hospitales, sucursales, centros logísticos, etc.).
- Sus propios sujetos y atributos.
- Sus capas geográficas (por ejemplo, departamentos, rutas, zonas de servicio).

Las capas se modelan como referencias a conjuntos de geometrías almacenadas en la base de datos geográfica y publicadas a través de GeoServer. Cada capa mantiene metadatos como nombre, tipo geométrico principal (punto, línea, polígono), nombre del datastore asociado y atributos configurables para visualización y análisis.

Los POIs se gestionan como una entidad separada, con su propia tabla y atributos configurables, pero comparten el modelo geométrico y de atributos con los sujetos. Esta simetría permite utilizar tanto sujetos como POIs como insumos en las operaciones espaciales.

4.2. Diseño de la geocodificación y niveles de confianza

El proceso de geocodificación transforma direcciones textuales en coordenadas geográficas que el sistema almacena en la entidad `Subject`. En GISO 1.0 se definieron distintas *formas normales* de dirección, inspiradas en el Modelo de Direcciones de la Infraestructura de Datos Espaciales de Uruguay (IDEuy) (Infraestructura de Datos Espaciales del Uruguay, 2025). Estas formas normalizadas combinan campos como departamento, localidad, calle, número, kilómetro o punto de interés para aumentar las probabilidades de obtener una ubicación precisa.

Cada geocodificador puede utilizar indicadores diferentes para expresar la precisión de una respuesta. Por ejemplo, el servicio de IDEuy incluye el campo `stateMsg`, mientras que Nominatim no define un indicador explícito de precisión. Para homogeneizar estos criterios, el modelo asigna a cada resultado un nivel de confianza discreto (alta, media o baja), derivado de los metadatos devueltos

por cada servicio y de reglas específicas. Estos niveles se codifican mediante colores en la interfaz (verde, amarillo, rojo), facilitando la identificación rápida de direcciones problemáticas.

En los casos en que la geocodificación no logra ubicar una dirección con confianza suficiente, el diseño contempla la intervención manual del usuario. Para ello se utilizan coordenadas de referencia precomputadas (por ejemplo, la capital departamental correspondiente o la ciudad de Montevideo) que permiten posicionar un marcador inicial sobre el mapa y acelerar la corrección manual.

GISO 1.0 integraba dos geocodificadores: IDEuy/SuDir y Nominatim. En GISO 2.0 se mantiene esta concepción general y se extiende incorporando Google Maps como fuente complementaria, permitiendo seleccionar el geocodificador por workspace. Así, cada organización puede priorizar la fuente que mejor se ajusta a su ámbito geográfico, su conjunto de datos o sus restricciones de uso.

4.3. Formato de datos y sistemas de referencia

La plataforma admite la importación de datos mediante archivos CSV (para sujetos, atributos y POIs) y capas vectoriales (*shapefiles*) para capas de contexto. En el diseño original se definió que todos los datos geométricos se normalizan internamente a un único sistema de referencia de coordenadas, `EPSG:4326` (WGS 84), con el objetivo de simplificar la interoperabilidad con servicios web geográficos, en particular con los servicios de IDEuy y GeoServer ("PostGIS Manual (v3.4)", 2025).

Esta decisión se mantiene en GISO 2.0. Los usuarios pueden cargar coordenadas o capas en distintos SRID, pero el backend transforma los puntos al SRID de referencia antes de almacenarlos. Esta transformación se realiza utilizando librerías de manejo de sistemas de referencia y de geometrías, garantizando la coherencia interna de las operaciones espaciales.

Al mismo tiempo, se amplía el diseño para contemplar la selección de sistemas de referencia en el cliente. El frontend incorpora un selector de sistemas de coordenadas que permite trabajar, por ejemplo, con `EPSG:4326`, `EPSG:32721` o sistemas nacionales como `EPSG:5382`, según la disponibilidad en el servidor GeoServer. El sistema mantiene una configuración de SRID por defecto a nivel de workspace, que se utiliza tanto para las consultas a GeoServer como para formatear las coordenadas mostradas al usuario.

4.4. Arquitectura lógica de la solución

La arquitectura de GISO se organiza en torno a tres componentes principales y un conjunto de servicios externos, como se resume en la Figura 4.2:

- **Frontend**: aplicación web basada en React y Next.js que se implementa como interfaz unificada de administración y análisis. El acceso a vistas y operaciones se regula mediante controles de permisos basados en el rol del usuario (Administrador o Analista).

- **Backend**: aplicación Spring Boot que centraliza la lógica de negocio de la plataforma. Gestiona la persistencia en la base de datos espacial, orquesta los procesos de geocodificación y expone una API REST para la comunicación con el frontend. La geocodificación masiva se ejecuta de manera asíncrona mediante una cola de mensajes Java Message Service (JMS) (*Jakarta Messaging Specification*, 2023) alojada en un servidor WildFly, desacoplando la carga de datos del proceso intensivo de consultas a servicios externos.
- **Infraestructura geoespacial**: conformada por una base de datos PostgreSQL con la extensión PostGIS, que almacena geometrías y atributos espaciales. Sobre esta base, un servidor GeoServer publica las capas como servicios web estándar del Open Geospatial Consortium (OGC), en particular Web Map Service (WMS) y Web Feature Service (WFS).
- **Sistemas externos**: incluye servicios de geocodificación (IDEuy, Nominatim, Google Maps), servicio de correo electrónico (Amazon Web Services Simple Email Service (AWS SES)) para notificaciones y activación de usuarios, y el servicio de autenticación ID Uruguay (GubUy) para identidad digital del Estado uruguayo.

4.4.1. Capas del backend

El backend sigue una arquitectura que separa las responsabilidades en diferentes capas:

- **Capa de presentación**: contiene los controladores REST que exponen los endpoints de la API. Estas clases reciben las peticiones del frontend, ejecutan casos de uso específicos y devuelven respuestas en formatos JSON o archivos CSV.
- **Capa de dominio**: concentra las reglas de negocio de la aplicación. Incluye entidades de dominio (sujeto, workspace, capa, POI, atributo, función geográfica), casos de uso (por ejemplo, carga de sujetos, selección de geocodificador, creación de funciones geográficas) e interfaces que definen las operaciones esperadas sobre las fuentes de datos.
- **Capa de datos**: implementa las interfaces del dominio mediante repositorios que acceden a la base de datos, a GeoServer, a los geocodificadores externos y a la cola de mensajes. Esta capa se encarga de los detalles tecnológicos (SQL, protocolos HTTP, JPA, JMS), manteniendo aislada la lógica de negocio.

Esta organización facilita el testeo de la lógica de negocio y reduce el acoplamiento con tecnologías específicas, permitiendo evolucionar la infraestructura (por ejemplo, cambiar de proveedor de base de datos o de servicio de mensajería) con un impacto acotado.

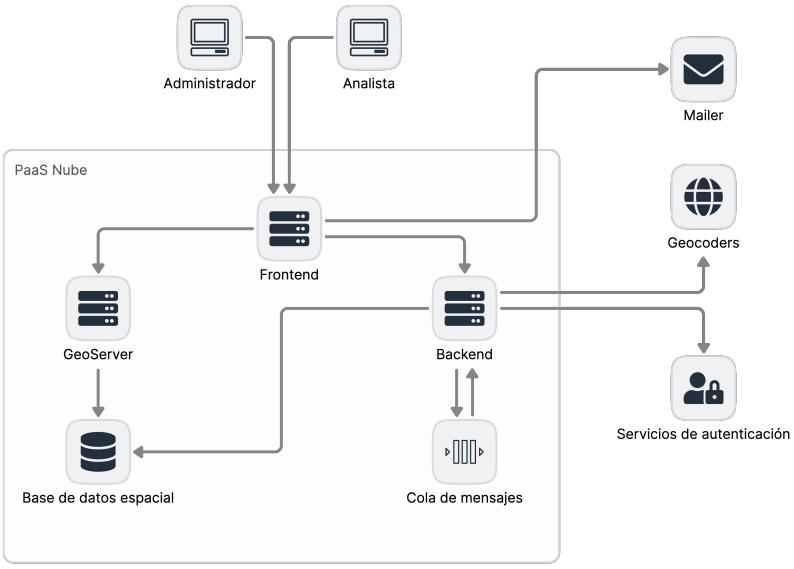


Figura 4.2: Arquitectura lógica de la plataforma GISO 2.0 con frontend unificado, backend, base espacial, GeoServer, geocodificadores externos y cola de mensajes.

4.4.2. Integraciones externas

La integración con GeoServer se realiza mediante las APIs WFS y la interfaz de administración REST. Desde la perspectiva del diseño, GeoServer cumple los siguientes roles:

- Publicar las vistas de sujetos y POIs como capas WFS filtradas por workspace, lo que permite al frontend consumir datos espaciales estandarizados.
- Gestionar capas adicionales cargadas por el administrador (por ejemplo, departamentos, rutas nacionales), con soporte para estilos y filtros avanzados.
- Exponer descripciones de esquema (*DescribeFeatureType*) que el backend utiliza para inferir la estructura de atributos de cada capa y ofrecer opciones de configuración al administrador.

Los geocodificadores externos se consumen mediante clientes HTTP que encapsulan los detalles de cada API (rutas, parámetros, formatos de respuesta). Desde el punto de vista de diseño, todos implementan una interfaz común que devuelve candidatos de dirección con coordenadas y metadatos de calidad. Esto permite al dominio tratar los resultados de IDEuy, Nominatim y Google Maps de forma homogénea y aplicar la misma lógica de selección y registro de confianza.

El servicio de correo electrónico se integra mediante AWS SES, que proporciona capacidades de envío transaccional de correos. Se optó por este servicio por la simplicidad de integración evitando la necesidad de configurar y mantener un servidor SMTP propio.

Finalmente, la autenticación con ID Uruguay se integra mediante el flujo de autorización de OAuth 2.0 (Hardt, 2012), en el que el backend intercambia códigos de autorización por tokens de acceso y consulta los datos básicos del usuario autenticado.

4.5. Evolución arquitectónica

Además de reutilizar el diseño conceptual y la arquitectura general de GISO 1.0, visualizada en la Figura 2.3, GISO 2.0 incorpora varias decisiones de diseño orientadas a mejorar la mantenibilidad, la usabilidad y las capacidades analíticas de la plataforma:

- **Unificación de frontends**: en GISO 1.0 existían dos aplicaciones web separadas, *giso-frontend* (orientada a analistas) y *giso-admin* (orientada a administradores), con estructuras muy similares pero código duplicado. En GISO 2.0 se integran ambas en un único frontend basado en Next.js, donde la separación de responsabilidades se expresa mediante rutas protegidas y controles de permisos. Esta unificación busca reducir la duplicación de lógica y componentes, y simplifica el despliegue a un solo artefacto con

un único conjunto de variables de entorno. Desde la perspectiva de desarrollo, trabajar sobre una única aplicación facilita la aplicación coherente de patrones de diseño y mejoras de usabilidad.

- **Gestión avanzada de workspaces**: se refuerza el workspace como unidad de aislamiento, de forma que cada administrador opera únicamente sobre sus propios datos, usuarios y capas. La configuración de atributos visibles, geocodificador y SRID por defecto se modela explícitamente en el dominio.
- **Geometrías temporales para análisis interactivo**: se incorpora una capa de geometrías temporales que permite a los usuarios crear puntos, líneas, polígonos y círculos directamente en el mapa. Estas geometrías se manejan en memoria y se utilizan como entrada para las funciones geográficas, sin necesidad de persistirlas como capas permanentes.

Estas decisiones preparan el terreno para la implementación descrita en el siguiente capítulo, donde se profundiza en cómo se materializan los componentes y flujos aquí presentados.

Capítulo 5

Implementación

En este capítulo se describe la estrategia de implementación de la plataforma GISO 2.0. Partiendo de la arquitectura presentada en el capítulo anterior, se detalla cómo se materializan los componentes de backend y frontend, las integraciones con servicios externos y el soporte para las capacidades de análisis espacial.

5.1. Tecnologías utilizadas

La estrategia de implementación se apoya en el stack tecnológico definido en GISO 1.0, manteniendo las herramientas de desarrollo web y geoespacial que han demostrado ser efectivas en términos de interoperabilidad con estándares abiertos y curva de aprendizaje. Además, se realizaron actualizaciones en las versiones de las librerías instaladas, con el objetivo de asegurar la mantenibilidad y estabilidad del sistema.

La Figura 5.1 resume la arquitectura de la plataforma y las tecnologías utilizadas en esta etapa.

Backend

Se implementa el backend en Java utilizando el framework Spring Boot ("Spring Boot Reference Documentation", 2025), con soporte para inyección de dependencias, configuración por perfiles y tareas programadas. Para la capa de persistencia se utiliza PostgreSQL junto con la extensión PostGIS ("PostGIS Manual (v3.4)", 2025). La comunicación con servicios externos se centraliza mediante una clase base de cliente HTTP que gestiona tiempos de espera y serialización JSON, sobre la cual se construyen los clientes de geocodificación (Sección 5.2.3).

Frontend

El frontend consiste en una única aplicación web basada en React y Next.js ("React Documentation", 2025; "Next.js Documentation", 2025). El uso del *App Router*

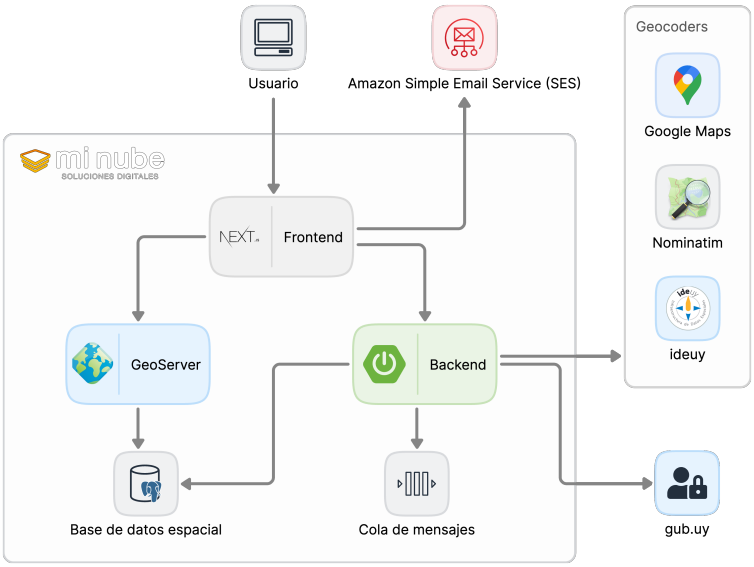


Figura 5.1: Diagrama de arquitectura con tecnologías de la plataforma GISO.

de Next.js permite organizar el código por rutas y segmentos anidados, diferenciando claramente las secciones de gestión de datos, análisis y administración. Para la gestión del estado de la interfaz (capas activas, filtros, funciones geográficas, geometrías temporales, sistema de coordenadas, exportación del mapa) se utiliza la librería Zustand ("Zustand Documentation", 2025), que ofrece un modelo de estado global simple y predecible.

La visualización de mapas y capas geográficas se realiza con Leaflet y su integración con React (*React-Leaflet*) ("Leaflet Reference (v1.9)", 2025; "React-Leaflet Documentation", 2025). Los cálculos espaciales interactivos (distancias, áreas, intersecciones) se implementan en el cliente utilizando Turf.js ("Turf.js Documentation", 2025), optando por el procesamiento en el navegador para ofrecer retroalimentación inmediata durante el análisis exploratorio y reducir la carga del backend en escenarios de uso concurrente. Como contrapartida, esta decisión puede limitar el rendimiento con conjuntos de datos de gran volumen. Para la visualización de estadísticas y gráficos se emplea Apache ECharts ("Apache ECharts Documentation", 2025), integrándolo mediante componentes React.

Infraestructura geoespacial e integraciones

Sobre la infraestructura descrita en la Sección 4.4, el frontend consume principalmente WFS en formato GeoJSON, lo que facilita el procesamiento con Turf.js y la visualización directa en el navegador. La geocodificación se realiza contra los servicios IDEuy, Nominatim y Google Maps, seleccionados por workspace según las necesidades del usuario.

5.2. Implementación del backend

La implementación del backend sigue la organización en capas descrita en el diseño, separando controladores, casos de uso y repositorios. A continuación se detallan los aspectos más relevantes vinculados a las extensiones en GISO 2.0.

5.2.1. Estructura de módulos y organización del código

El código del backend se encuentra dividido en dos módulos principales:

- El módulo `application`, que contiene la aplicación Spring Boot, los controladores, la configuración, las tareas programadas y los repositorios concretos.
- El módulo `domain`, que concentra las entidades de dominio, interfaces de repositorios y casos de uso. Este módulo se mantiene independiente de las tecnologías específicas de persistencia o comunicación.

Dentro del módulo de aplicación, el paquete `controller` se organiza por subdominios (usuarios, workspaces, sujetos, POIs, capas, filtros, geocodificadores, funciones geográficas), y cada controlador expone endpoints que delegan en

un caso de uso específico. Los casos de uso se declaran en el módulo de dominio y se instancian en la configuración de Spring, lo que permite combinarlos de forma explícita y reutilizable.

La capa de datos se implementa mediante distintos tipos de repositorios:

- Repositorios Java Persistence API (JPA) para entidades persistentes como *sujetos*, atributos, capas, usuarios y funciones geográficas.
- Repositorios basados en `JdbcTemplate` para consultas especializadas o de agregación.
- Repositorios REST para servicios externos (geocodificadores, GeoServer, GubUy).
- Repositorios de cola de mensajes para la interacción con la infraestructura JMS.

5.2.2. Carga y validación de datos

La carga de sujetos y POIs desde archivos CSV se implementa mediante un flujo que combina validaciones en el backend y en el frontend. En el backend, una utilidad de lectura de CSV se encarga de:

- Verificar el tipo de archivo.
- Interpretar la primera fila como encabezado, normalizando los nombres de columna y resolviendo problemas de codificación de caracteres.
- Construir una representación intermedia del contenido, con filas y celdas asociadas al nombre de cada atributo.

Sobre esta estructura se aplican las reglas de negocio definidas en el dominio: comprobar que el archivo no está vacío, verificar que existe el atributo identificado como clave del sujeto en el workspace, mapear automáticamente columnas de encabezado a atributos y rellenar con valores nulos los atributos configurados que no aparecen en el archivo. Los atributos adicionales que no forman parte del modelo se ignoran, siempre que la estructura general del archivo sea válida. Ante errores de lectura o ausencia de identificadores se genera una excepción específica, que el controlador traduce en mensajes claros para el usuario.

En el frontend se incorpora un validador previo que inspecciona el encabezado del CSV antes de enviarlo al servidor. Este componente muestra al usuario qué columnas del archivo coinciden con los atributos definidos en el workspace, cuáles faltan y cuáles son columnas adicionales. También verifica la presencia del atributo identificado como clave del sujeto. Esta validación previa reduce el número de intentos de carga fallidos y se mejora el diagnóstico temprano de problemas de formato, manteniendo la lógica de validación de negocio en el backend.

La Figura 5.2 muestra la interfaz de carga y el proceso de validación previa del CSV antes de su envío al backend.



Figura 5.2: Interfaz de carga y validación de archivos CSV antes de su envío al backend.

La carga de capas vectoriales se realiza a través del backend, que recibe archivos comprimidos con los componentes de un *shapefile* y los publica en GeoServer como nuevos datastores asociados a un workspace. Tras la publicación se consulta el esquema de la capa mediante la operación *DescribeFeatureType* de WFS y se filtran los atributos geométricos, construyendo una lista de atributos disponibles para su configuración en la interfaz de administración.

5.2.3. Geocodificación multi-fuente y cola de mensajes

El proceso de geocodificación implementa la estrategia multi-fuente descrita en el capítulo de diseño. Para cada workspace se almacena la selección de geocodificador preferido, y un repositorio específico consulta esta configuración antes de resolver una dirección. El flujo es el siguiente:

1. Al cargar un CSV de sujetos, el backend almacena cada registro con estado pendiente y sin coordenadas.

2. Por cada sujeto, se publica un mensaje en una cola JMS que incluye el identificador del sujeto, el identificador del workspace y los atributos de dirección relevantes.

3. Un componente consumidor escucha la cola y, por cada mensaje, construye una dirección normalizada a partir de los atributos configurados en el workspace.

4. El consumidor invoca al repositorio de geocodificación, que delega en el cliente IDEuy, Nominatim o Google Maps según la configuración, aplicando reintentos y manejando tiempos de espera.

5. Si se obtiene un candidato válido, se transforma la geometría al SRID interno y se actualiza el sujeto con el punto resultante, el nivel de confianza y el estado activo. En caso contrario, el sujeto se mantiene sin coordenadas y el sistema queda preparado para una corrección manual.

Los clientes de geocodificación se desarrollan extendiendo una clase base de cliente REST, que centraliza la construcción de peticiones HTTP, el manejo de cabeceras, la deserialización de las respuestas y el control de errores. Cada servicio implementa las particularidades de su API (parámetros, rutas, estructura de respuesta) y expone un método que devuelve un candidato de dirección en el modelo interno de la aplicación.

5.2.4. Gestión de workspaces, sujetos, POIs y funciones geográficas

La gestión de workspaces y usuarios se implementa mediante controladores y casos de uso que aplican las reglas de permisos definidas en el análisis. Cada administrador solo puede crear, editar y consultar workspaces propios, y los usuarios deben estar asociados al menos a un workspace para poder acceder a la plataforma. Los controladores de usuario exponen operaciones para crear cuentas, actualizar datos, asignar usuarios a workspaces y gestionar sus contraseñas, incluyendo el flujo de establecimiento inicial de contraseña a partir de un token.

Los sujetos y POIs se exponen mediante endpoints paginados que permiten búsquedas por identificador, filtrado por nivel de confianza de geocodificación y acceso a vistas detalladas. El backend mantiene vistas materializadas o consultas parametrizadas que exponen estos datos como capas en GeoServer filtradas por workspace y estado, facilitando el consumo desde el frontend.

Las funciones geográficas se representan en la base de datos mediante una tabla de operaciones y una tabla de funciones definidas por el usuario. Cada función almacena su nombre, el tipo de operación (conteo de puntos, distancia promedio al vecino más cercano, distancia punto-línea, buffer, área de polígonos, intersección espacial, análisis de proximidad), el workspace al que pertenece y los parámetros necesarios (capas fuente y destino, distancia, número de vecinos, entre otros).

El backend expone endpoints para listar, crear, actualizar y eliminar funciones, y para recuperar el conjunto de operaciones disponibles. El cálculo numérico asociado a estas funciones se realiza en el frontend, como se detalla más adelante, mientras que el backend se encarga de persistir la definición y de proveer acceso a las capas involucradas.

5.3. Implementación del frontend

Como se describe en la Sección 4.5 del capítulo de Diseño y arquitectura, en GISO 2.0 se unifican las dos aplicaciones web de GISO 1.0 en un único frontend basado en Next.js, controlando el acceso por roles mediante rutas protegidas.

Se definieron segmentos y *layouts* específicos para cada área funcional:

- Selección de workspace, que actúa como puerta de entrada a la aplicación y determina el contexto de los datos.
- Gestión de datos, donde se concentran la carga y edición de sujetos, POIs y capas.
- Análisis de datos, que agrupa las herramientas de filtros, funciones geográficas y visualización en mapa.
- Administración, que incluye dashboards de resumen, gestión de usuarios, workspaces y geocodificadores.

Cada ruta de Next.js utiliza componentes de alto nivel que a su vez componen estructuras más pequeñas, siguiendo un patrón de separación entre contenedores (que obtienen datos mediante llamadas al backend) y componentes de presentación.

5.3.1. Gestión de estado y comunicación con el backend

Para la comunicación con el backend se utilizan servicios centralizados que encapsulan las llamadas HTTP y el manejo del token de autenticación. Estos servicios incluyen funciones para recuperar atributos, sujetos, POIs, capas, filtros, funciones geográficas y métricas agregadas, así como para invocar acciones de creación y actualización mediante *server actions* de Next.js.

La librería Zustand se utiliza para mantener el estado compartido de la aplicación. Se definen almacenes de estado (*stores*) específicos para:

- Capas cargadas en el mapa, incluyendo su orden, color y conjunto de features.
- Filtros definidos por el usuario y sus resultados geográficos.
- Funciones geográficas y sus resultados visuales.
- Sistema de coordenadas actual y sistemas disponibles, incluyendo la lógica para verificar qué SRID soporta el GeoServer configurado.
- Geometrías temporales dibujadas por el usuario en el mapa.
- Parámetros de exportación del mapa, como resolución y elementos visibles.

Este enfoque facilita la sincronización entre los distintos paneles de la interfaz (lista de capas, configurador de funciones, mapa, paneles de detalle) y reduce la duplicación de lógica de estado.

5.3.2. Mapa, capas y filtros geográficos

La visualización principal se implementa mediante React-Leaflet, usando un componente contenedor que inicializa el mapa con un centro y nivel de zoom predefinidos, y una serie de componentes hijos responsables de:

- Cargar y mostrar las capas base (por ejemplo, OpenStreetMap).
- Renderizar las capas de sujetos, POIs y capas de contexto obtenidas desde GeoServer mediante WFS.
- Aplicar clustering a conjuntos densos de puntos para mantener la legibilidad del mapa.
- Mostrar ventanas informativas con atributos relevantes al hacer clic sobre un elemento.
- Gestionar eventos de interacción (clics, zoom, movimiento) y comunicar estos cambios a los stores correspondientes.

Los filtros definidos en el backend se materializan como consultas CQL[1] sobre las vistas de sujetos. El frontend construye estas consultas a partir de las condiciones configuradas (atributos, operadores relacionales y lógicos, condiciones geográficas como `DWITHIN` y `WITHIN`) y las envía a GeoServer para obtener las features filtradas. Los resultados se incorporan al mapa como una capa adicional, con estilo diferenciado.

5.3.3. Geometrías temporales y funciones espaciales en el cliente

Una de las extensiones relevantes es la incorporación de geometrías temporales para análisis interactivo. La implementación se basa en un store de geometrías temporales que mantiene:

- La lista de geometrías creadas (puntos, líneas, polígonos), con identificadores únicos, estilo y metadatos básicos.
- El estado de creación actual (tipo de geometría en construcción, puntos acumulados, operaciones en curso).
- La selección de capas y filtros a considerar como insumo de las funciones geográficas.
- Opciones de operación como radio de buffer, número de vecinos y uso de todos los puntos destino o solo del más cercano.

[1] CQL (*Common Query Language*) es un lenguaje de consulta y filtrado utilizado en servicios geoespaciales, como GeoServer, para expresar condiciones sobre atributos y relaciones espaciales.

El componente de mapa registra los clics del usuario y, según el estado de creación, va construyendo la geometría temporal correspondiente. Una vez finalizada, se añade a la colección de geometrías temporales y se representa en el mapa mediante una capa específica. Estas geometrías pueden utilizarse como entrada para funciones espaciales como buffers, análisis de proximidad o intersección con capas existentes.

Los cálculos espaciales asociados a las funciones geográficas descritas en la Sección 2.7 se implementan en el cliente utilizando Turf.js. Como se mencionó anteriormente, los cálculos se ejecutan en el cliente para ofrecer menor latencia, sin requerir *round-trips* HTTP al servidor. Se desarrolla un módulo de utilidades que:

- Normaliza y valida las geometrías obtenidas desde GeoServer, corrigiendo problemas de codificación y estructuras inválidas cuando es posible.
- Calcula distancias punto-punto, funcionalidad ya presente en GISO 1.0 que se extiende con el cálculo del vecino más cercano y, opcionalmente, distancias a todos los puntos destino. Incorpora además distancia punto-línea (nueva en GISO 2.0) mediante proyección sobre las líneas o rutas seleccionadas.
- Calcula áreas y perímetros de polígonos, devolviendo métricas expresadas en metros, kilómetros cuadrados y hectáreas.
- Evalúa intersecciones espaciales entre conjuntos de polígonos o entre puntos y polígonos, usando estrategias de optimización como índices espaciales en cuadrícula y procesamiento por lotes para evitar bloqueos de la interfaz.

Los resultados de estos cálculos se almacenan en un store específico que mantiene, para cada función geográfica, la lista de mediciones y estadísticas agregadas (número de elementos, distancias mínimas, máximas y promedio, áreas totales e individuales, número de intersecciones). Un componente de visualización se encarga de representar estas mediciones sobre el mapa mediante segmentos, conectores, etiquetas y polígonos de intersección, utilizando paletas de colores diferenciadas por tipo de operación.

La Figura 5.3 ilustra un ejemplo de función **Buffer** aplicada sobre una capa de sujetos, con visualización directa del resultado en el mapa. En términos funcionales, esta evolución amplía el repertorio de operaciones espaciales de GISO 1.0, que contaba con dos capacidades base (distancia punto-punto y conteo en polígono), a siete operaciones disponibles en GISO 2.0.

5.3.4. Exportación de mapas y soporte para distintos SRID

La exportación de mapas se implementa mediante un contexto específico que coordina la inicialización de una librería de impresión sobre Leaflet y la captura del estado actual del mapa (capas visibles, estilo, geometrías temporales, resultados de funciones geográficas). El usuario puede generar imágenes de alta

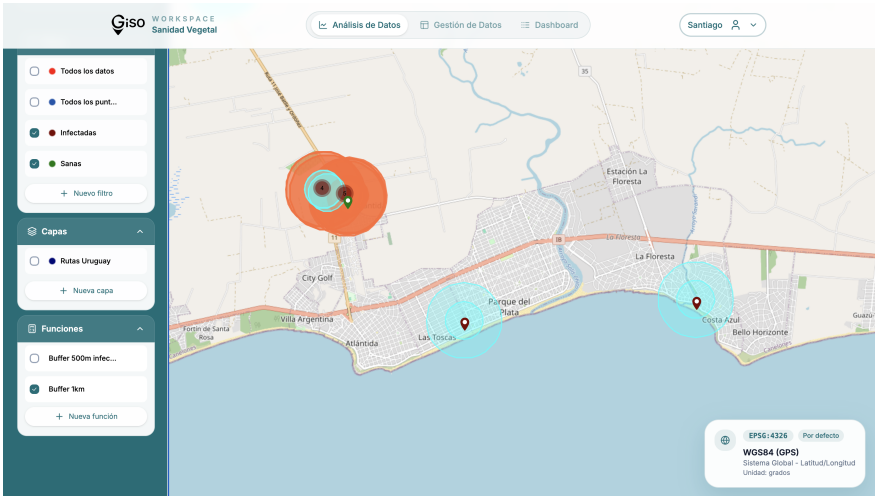


Figura 5.3: Ejemplo de función buffer aplicada sobre una capa de sujetos, mostrando las áreas de influencia generadas.

resolución que incluyen la información relevante para documentar análisis o incorporar resultados en informes externos.

El soporte para distintos sistemas de coordenadas se implementa combinando una configuración estática de sistemas relevantes para Uruguay y una verificación dinámica contra GeoServer. Al iniciar la sesión, el frontend consulta qué SRID están disponibles a través de una operación *GetCapabilities* y ajusta la lista de sistemas seleccionables. Cuando el usuario cambia el sistema de referencia, el frontend solicita nuevamente las capas a GeoServer con el `srsName` correspondiente y actualiza el formato de las coordenadas mostradas, manteniendo la coherencia entre los datos y el sistema elegido.

5.4. Despliegue y configuración

Para el desarrollo local se configura un entorno reproducible que orquesta GeoServer, el backend y el frontend mediante scripts y configuración declarativa. GeoServer se ejecuta en un contenedor Docker (Docker, Inc., 2025) con un directorio de datos persistente y la extensión QueryLayer habilitada para soportar consultas avanzadas sobre capas (“GeoServer User Manual”, 2025). El backend Spring Boot se ejecuta con un perfil local que apunta a una instancia de PostgreSQL con PostGIS y al servidor WildFly responsable de la cola JMS. El frontend se ejecuta en modo desarrollo a través de los comandos estándar de Next.js.

GISO 1.0 desplegó la solución en Elastic Cloud de MiNube de Antel (Antel, 2025a), un servicio de plataforma como servicio (PaaS), separando los componentes en distintos entornos (frontend, backend, GeoServer y base de datos).

En GISO 2.0 se vuelve a realizar el despliegue en Elastic Cloud de MiNube, documentando un procedimiento reproducible de configuración y operación que facilita la portabilidad a plataformas equivalentes. Los detalles concretos de configuración de credenciales y variables de entorno se documentan en el manual de instalación.

El siguiente capítulo describe las pruebas funcionales, técnicas y de validación con usuarios que verifican el cumplimiento de los requerimientos planteados.

Capítulo 6

Pruebas y validación

Este capítulo describe las pruebas realizadas sobre la plataforma GISO 2.0. Se abordan verificaciones funcionales y técnicas, junto con instancias de validación con usuarios finales orientada a evaluar la experiencia de uso. El objetivo es demostrar que las funcionalidades implementadas operan correctamente, que el sistema mantiene un rendimiento adecuado bajo cargas realistas y que las mejoras introducidas facilitan el trabajo de analistas no expertos en GIS.

6.1. Pruebas funcionales y técnicas

Se realizan pruebas orientadas a validar las funcionalidades del sistema, sin involucrar aún a usuarios finales. Estas pruebas abarcan los siguientes aspectos:

- **Pruebas de las funcionalidades de carga y validación de archivos**: se verifica la capacidad del sistema para aceptar archivos CSV y *shapefiles* de distintos tamaños, validar su estructura (encabezados, tipos de datos) y contenido (columnas obligatorias, SRID) e incorporarlos correctamente al *workspace* correspondiente. Se introducen archivos mal formados para comprobar que el sistema genere mensajes de error claros.
- **Pruebas de visualización y edición**: se comprueba el correcto renderizado de capas en el mapa, la visibilidad de estilos configurados por *workspace* y el comportamiento de las vistas detalladas de sujetos y puntos de interés (POI). También se valida la edición de atributos, latitud, longitud y SRID desde la interfaz, así como el cambio de capa base y de sistema de referencia de coordenadas en el cliente.
- **Pruebas de operaciones espaciales**: se prueban las funciones de área y perímetro, buffer, distancia punto-línea, análisis de proximidad (N vecinos) e intersección espacial sobre conjuntos de prueba.
- **Pruebas de geometrías temporales**: se verifica la creación interactiva de puntos, líneas, polígonos y círculos sobre el mapa, la gestión de su estado

(creación, finalización, cancelación, limpieza) y su uso como entrada en funciones como buffer, análisis de proximidad e intersección.

- **Pruebas de exportación de mapas**: se comprueba la capacidad de generar imágenes del estado actual del mapa, incluyendo capas activas, geometrías temporales y resultados de análisis. Se verifica que las exportaciones conserven la simbología y los elementos de contexto relevantes.

- **Pruebas de activación de cuenta**: se verifica el flujo completo de activación de nuevos usuarios, incluyendo el envío del correo electrónico con enlace, la validez y expiración del token de activación, el manejo de tokens expirados o inválidos, y el establecimiento correcto de la contraseña inicial.

- **Pruebas de manejo de errores y permisos**: se introducen operaciones forzando errores de permisos, validación y carga para comprobar que el sistema muestre mensajes comprensibles y diferencie claramente la causa del problema. Se verifica que los administradores solo puedan operar sobre sus *workspaces* y usuarios asociados.

Como síntesis de estas pruebas, se ejecutaron siete grupos de pruebas sobre los diferentes flujos. No se detectaron fallas bloqueantes en los recorridos principales. Se identificaron dos incidencias relevantes para la evolución del sistema: la necesidad de mayor detalle en algunos mensajes de carga de CSV y la degradación de tiempos de geocodificación cuando coinciden lotes grandes en la cola. En términos de volumen, las pruebas realizadas en el sistema abarcan desde lotes menores a 500 registros hasta un escenario de 65.661 registros en el caso de estudio del Capítulo 7.

6.2. Validación con usuarios

La validación con usuarios se organiza en dos instancias complementarias. La primera corresponde a un taller estructurado con estudiantes de posgrado, de alcance temporal acotado, y la segunda consiste en un uso prolongado del sistema en una organización real, con seguimiento del uso mediante reuniones e intercambios por correo.

6.2.1. Taller con estudiantes de posgrado

Además de las pruebas del sistema, se realiza una instancia orientada a evaluar la experiencia de uso de la plataforma. El objetivo es verificar si las mejoras introducidas (navegación, mensajes, validación de archivos CSV, geometrías temporales y nuevas funciones geográficas) facilitan el trabajo de analistas no expertos en GIS.

Antes de presentar los resultados, conviene aclarar las limitaciones de esta instancia de validación. La cantidad de participantes es reducida y de conve-

niencia: participaron 13 estudiantes de un curso de posgrado en datos y análisis espacial, de los cuales 6 completaron el formulario de evaluación.

Estas características implican que los resultados deben interpretarse como orientativos y no como generalizables a la población de usuarios finales. En particular, la familiaridad previa con conceptos geoespaciales puede sobreestimar la facilidad de uso percibida. Futuras evaluaciones deberían incorporar usuarios sin formación en GIS, evitando sesgar la muestra.

Cada participante accede a la aplicación y trabaja con conjuntos de datos propios (por ejemplo, sucursales de cadenas de supermercados en Uruguay). Al finalizar, cada uno completa un formulario en línea que recoge información objetiva sobre el uso del sistema y la percepción subjetiva de la experiencia.

Diseño del cuestionario

El formulario se estructura en secciones que combinan preguntas cerradas con escala Likert [1] de cinco puntos (para cuantificar resultados) y preguntas abiertas (para recoger comentarios cualitativos). A continuación se sintetizan los bloques principales con ejemplos representativos de cada categoría.

Perfil y contexto de uso

- **Dominio de trabajo durante la prueba**: selección de opciones (por ejemplo, logística de supermercados, salud, otro).
- **Experiencia previa con GIS o mapas web**: escala de 1 (ninguna) a 5 (experto).
- **Rol actual**: opciones como *perfil técnico*, *perfil de análisis de datos*, *perfil de negocio*, *otro*.

Datos utilizados y geocodificación

- **Volumen aproximado de datos cargados**: rangos (por ejemplo, menos de 500 registros, entre 500 y 2.000, más de 2.000).
- **Cantidad total de direcciones geocodificadas**: respuesta numérica.
- **Cantidad de direcciones corregidas o ubicadas manualmente**: respuesta numérica.
- **Percepción sobre el proceso de geocodificación**: escala 1-5 para aspectos como rapidez, claridad de los estados (en cola, procesando, completado) y confianza en los resultados.

[1] La escala Likert es una escala ordinal de respuesta utilizada para medir percepciones o valoraciones en categorías ordenadas. En este trabajo se utiliza una escala de 1 a 5, donde 1 representa la valoración más baja y 5 la más alta.

Uso de funciones espaciales

- **Funciones utilizadas**: selección múltiple (área y perímetro, buffer, distancia punto–línea, análisis de proximidad, intersección espacial, geometrías temporales, otras).
- **Claridad de las funciones geográficas**: escala 1-5 para evaluar qué tan fácil resulta entender qué hace cada función a partir de su nombre y descripción.
- **Utilidad percibida**: escala 1-5 para valorar si las funciones disponibles permiten responder las preguntas del caso de estudio.

Navegación y diseño de la interfaz

- **Facilidad para encontrar las principales acciones**: escala 1-5 para operaciones como cargar datos, ejecutar geocodificación, lanzar un análisis espacial, editar datos.
- **Claridad de la navegación entre vistas**: escala 1-5 para valorar la estructura de menús y la posibilidad de volver atrás usando botones de regreso.
- **Comprensión de textos y etiquetas**: escala 1-5 sobre la claridad de nombres de botones, menús y mensajes, especialmente para usuarios sin experiencia previa en GIS.

Feedback, errores y validaciones

- **Claridad de los mensajes de error**: escala 1-5 acompañada de una pregunta abierta sobre algún mensaje que resulte confuso.
- **Utilidad de los mensajes de éxito (*toasts*)**: escala 1-5 para medir si el sistema confirma adecuadamente las operaciones completadas.
- **Percepción sobre *loaders* e indicadores de progreso**: escala 1-5 para valorar si los indicadores ayudan a entender que una operación está en curso.

Geometrías temporales y mapa

- **Facilidad para crear y editar geometrías temporales**: escala 1-5 y una pregunta abierta para describir dificultades específicas.
- **Uso de geometrías temporales en el análisis**: pregunta de selección múltiple sobre para qué se utilizan (definir zona de estudio, modelar rutas, definir área de cobertura, otro).

Valoración global y comentarios abiertos

- **Facilidad de uso general de la plataforma**: escala 1-5.
- **Probabilidad de usar una herramienta similar en su trabajo**: escala 1-5.
- **Aspecto más positivo de la experiencia**: pregunta abierta.
- **Aspecto más confuso o a mejorar**: pregunta abierta.

Este conjunto de preguntas permite relacionar la retroalimentación de los usuarios con las decisiones de diseño tomadas en el proyecto, en particular en lo relativo a usabilidad, flujos de navegación y funciones geográficas.

Resultados del taller

De los trece estudiantes que participaron en la sesión, seis completaron el formulario de evaluación. Como se indicó previamente, el tamaño reducido de la muestra ($n = 6$) implica que los valores numéricos que se presentan a continuación deben interpretarse como indicadores orientativos de tendencias y no como resultados estadísticamente generalizables.

La Tabla 6.1 presenta un resumen de las valoraciones obtenidas.

Tabla 6.1: Valoración media de las principales dimensiones evaluadas en la encuesta de experiencia de usuario (escala 1–5, $n = 6$).[3]

Dimensión	**Media**	**Rango (mín–máx)**
Carga de datos	5,0	5–5
Navegación	4,5	4–5
Claridad de funciones	4,0	3–5
Facilidad general	4,0	4–5
Utilidad de funciones	3,5	3–4
Geometrías temporales	3,5	3–4
Feedback del sistema	3,5	3–4
Geocodificación	3,8	3–5

Perfil de los participantes

La mitad de quienes respondieron reportaron poca experiencia previa con GIS (nivel 2 en la escala de cinco puntos), mientras que el resto indicó un nivel intermedio (nivel 3). Los roles declarados abarcaron perfiles técnicos (líder técnico) y usuarios con conocimientos previos en herramientas GIS.

[3] Valores orientativos; el tamaño de la muestra no permite establecer significancia estadística.

Carga de datos y geocodificación

Todos los participantes que respondieron la encuesta trabajaron con conjuntos de menos de 500 registros. En promedio, cada uno geocodificó 140 direcciones. La percepción del proceso resultó favorable. La rapidez obtuvo 3/5, la claridad de los estados (en cola, procesando, completado) alcanzó 3,5/5 y la confianza en los resultados 5/5. La cantidad de direcciones ubicadas manualmente fue baja en todos los casos (entre 0 y 2 correcciones por participante), lo que sugiere que los participantes no recurrieron a esta función durante la sesión.

Funciones espaciales

Las funciones más utilizadas fueron **área y perímetro** y **buffer**, empleadas por todos los participantes. La mitad también utilizó **distancia punto-línea** y **geometrías temporales**. La claridad de las funciones geográficas obtuvo una valoración media de 4/5, mientras que la utilidad percibida para responder las preguntas del caso de estudio fue de 3,5/5.

Navegación e interfaz

La facilidad para encontrar las principales acciones recibió calificaciones altas:

- Cargar datos: 5/5
- Ejecutar geocodificación: 4,5/5
- Lanzar un análisis espacial: 4,5/5
- Editar datos: 4,5/5

La claridad de la navegación entre vistas alcanzó 4,5/5 y la comprensión de textos y etiquetas 4/5. Todos los participantes indicaron que pudieron encontrar las funciones principales sin dificultad significativa.

Feedback y mensajes del sistema

La claridad de los mensajes de error obtuvo 4/5. Un participante mencionó que algunos mensajes al cargar archivos CSV resultaron poco claros. La utilidad de los *toasts* de confirmación fue valorada con 3/5, mientras que los indicadores de progreso (*loaders*) obtuvieron 3,5/5. Estos resultados sugieren que, aunque los mensajes de éxito cumplen su función, podrían enriquecerse con información adicional sobre el estado del proceso.

Geometrías temporales

La funcionalidad de geometrías temporales fue utilizada por la mitad de los participantes. Su facilidad de uso recibió una valoración de 3,5/5. Los usos reportados incluyeron definir zonas de estudio y delimitar áreas de cobertura para análisis de proximidad.

Valoración global

La facilidad de uso general de la plataforma obtuvo una calificación media de 4/5, con el 100 % de los participantes asignando 4 o más. La probabilidad de usar una herramienta similar en el trabajo fue de 3,5/5.

Comentarios cualitativos

Entre los aspectos positivos destacados se mencionaron:

- *"Es intuitivo, claridad de navegación entre vistas, claridad de funciones geográficas."*
- *"Otra manera de ver estas funciones agrupadas."*

Como puntos a mejorar se señalaron:

- *"Agregar más mensajes de estado en el proceso de carga de datos que permita identificar en qué etapa se encuentra de la carga luego del procesamiento."*
- Un participante reportó que el mensaje mostrado al cargar un archivo CSV con columnas no reconocidas resultó confuso, ya que no indicaba con claridad cuáles columnas se ignoraron ni por qué motivo.
- Uno de los usuarios reportó demoras en el proceso de geocodificación. Se identificó que este problema se debió a otro usuario que cargó un CSV con una cantidad muy grande de puntos, lo que enlenteció el proceso para el resto de los participantes.

Uso observado de participantes sin formulario

Si bien siete participantes de la instancia no completaron el formulario, se registró uso efectivo de la plataforma durante la sesión. En particular, se observaron interacciones sobre los flujos principales de trabajo (carga y revisión de datos, geocodificación y ejecución de funciones espaciales), lo que aporta evidencia complementaria de adopción inicial más allá de las respuestas informadas.

Síntesis del taller

Los resultados del taller indican que las mejoras en navegación y mensajes, y la incorporación de funciones geográficas avanzadas recibieron valoraciones favorables. La plataforma logró una buena aceptación general entre los participantes de esta instancia.

Las áreas de mejora identificadas (principalmente relacionadas con el feedback durante operaciones de carga) permiten priorizar ajustes para futuras iteraciones sin comprometer la funcionalidad central del sistema.

6.2.2. Uso por parte de empresa del medio

Como segunda instancia de validación, se realizó una prueba de uso con un usuario de una empresa del medio del rubro distribución y logística. El contacto se estableció a través de Ingeniería deMuestra (IdM), a partir del interés en evaluar la plataforma por parte de la empresa. El perfil participante correspondió a usuarios no técnicos del área comercial, con experiencia previa en Google Earth y dificultades para sostener trabajo colaborativo con la herramienta utilizada hasta ese momento.

Para esta prueba se crearon usuarios, se asignó un administrador y se habilitó un workspace para trabajar con datos reales no sensibles del negocio. La plataforma estuvo disponible durante más de dos meses; no obstante, el período incluyó verano y licencias, por lo que el uso fue intermitente.

La retroalimentación se relevó mediante reuniones de seguimiento e intercambio por correo. En esta ocasión se destacó que la plataforma resultó intuitiva y amigable para el perfil de usuario, y también se valoró positivamente el acompañamiento del equipo durante el periodo. A su vez, se indicó que aún resta profundizar en algunas funcionalidades por la disponibilidad acotada de los usuarios durante el período de prueba. También se señaló como línea pendiente la evaluación específica del trabajo colaborativo y se manifestó interés en replicar en GISO procesos que actualmente se realizan en Google Earth.

6.2.3. Síntesis de la validación con usuarios

En conjunto, las dos instancias aportan evidencia complementaria. El taller con estudiantes permitió obtener métricas comparables sobre usabilidad y uso de funciones específicas en una actividad acotada. La prueba en empresa aportó una validación cualitativa en un contexto operativo real y permitió identificar necesidades de adopción sostenida, en particular para escenarios de trabajo colaborativo.

En el capítulo siguiente se presentan dos casos de estudio que ilustran la aplicación de la plataforma en dominios dispares.

Capítulo 7

Casos de estudio

Este capítulo presenta dos escenarios de aplicación práctica de la plataforma GISO 2.0 que validan su flexibilidad en dominios dispares, así como también las funciones de análisis introducidas en este proyecto. El primero, desarrollado en profundidad, describe la planificación de un centro de distribución logística utilizando datos reales de cadenas de supermercados. El segundo presenta un caso de análisis en sanidad vegetal (monitoreo del Picudo Rojo), demostrando la capacidad de la herramienta para gestionar grandes volúmenes de datos en vigilancia epidemiológica.

7.1. Caso 1: Planificación de un centro de distribución logística

Este escenario permite mostrar la aplicabilidad práctica de la solución utilizando datos basados en cadenas de supermercados de Uruguay, destacando su potencial para apoyar decisiones sobre la localización de centros de distribución y la cobertura de los repartos, especialmente cuando se emplean vehículos con autonomía limitada (por ejemplo, camiones eléctricos).

7.1.1. Contexto del caso de estudio

En este caso de estudio, los sujetos representan sucursales de diferentes cadenas de supermercados distribuidas en varios departamentos del país. Una empresa de logística que abastece a estas sucursales desea evaluar la ubicación de un nuevo centro de distribución (depósito). El objetivo es maximizar la cantidad de tiendas atendidas dentro de un cierto radio de operación y, al mismo tiempo, mantener una buena accesibilidad a rutas nacionales principales.

El problema se centra en responder preguntas como: ¿qué ubicación candidata para el depósito permite abastecer la mayor cantidad de sucursales con camiones que tienen una autonomía máxima de 30 km (distancia máxima de

recorrido sin recargar)?, ¿qué departamentos quedarían cubiertos por ese depósito?, ¿cómo se relaciona la localización propuesta con las rutas principales del Uruguay (por ejemplo, la Ruta 5)?, y ¿qué conjuntos de sucursales podrían agruparse en un mismo reparto según su proximidad geográfica?

La Figura 7.1 muestra la distribución de sucursales y las capas de contexto utilizadas (departamentos y rutas nacionales) para el análisis.

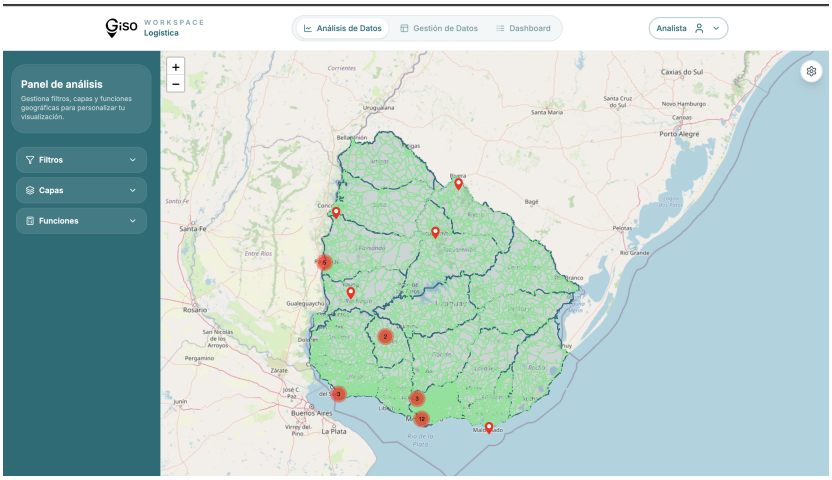


Figura 7.1: Distribución de sucursales y capas de contexto (departamentos y rutas nacionales) en el caso de estudio de logística.

7.1.2. Datos del negocio

Para este caso de estudio se construye un conjunto de datos a partir de información pública de cadenas de supermercados de Uruguay. Se dispone de un archivo CSV con aproximadamente 400 registros, cada uno representando una sucursal, con encabezados que incluyen la cadena a la que pertenece la sucursal (por ejemplo, Ta-Ta, Disco, Devoto), la denominación específica (por ejemplo, TATA Unión, Disco Cordón), la dirección textual con calle, número y localidad (por ejemplo, 18 de Julio 1593, Montevideo) y el departamento en el que se encuentra la sucursal.

Estas direcciones se utilizan como insumo para el proceso de geocodificación, generando puntos con coordenadas de latitud y longitud. Además, se consideran capas de contexto (cargadas como *shapefiles* provenientes de fuentes oficiales) que incluyen los límites departamentales, las principales rutas nacionales y candidatos de depósito creados dinámicamente como puntos o polígonos que representan posibles ubicaciones de un nuevo centro de distribución.

7.1.3. Ejecución del caso en la aplicación

A continuación se describe el flujo de trabajo seguido por un usuario con rol de analista para ejecutar el caso de estudio y aplicar las distintas funcionalidades de la plataforma.

Configuración del workspace y carga de datos

Un usuario administrador crea el workspace y el analista configura los atributos asociados a la capa de sucursales y carga el archivo CSV; el módulo de validación verifica la integridad del archivo y realiza el mapeo automático entre los encabezados y los atributos del workspace. En forma análoga, se cargan las capas de departamentos y rutas nacionales como *shapefiles*, configurando su SRID y estilos de visualización.

Geocodificación de sucursales

Una vez validado el CSV, se ejecuta la geocodificación masiva utilizando el geocodificador seleccionado para el workspace, almacenando coordenadas y nivel de confianza cuando corresponde. Durante el proceso, la interfaz muestra el progreso mediante *loaders* y permite actualizar la vista para observar cómo se van localizando las sucursales en el listado.

Definición de un sitio candidato mediante geometrías temporales

Para estudiar la ubicación de un nuevo centro de distribución, el analista utiliza la funcionalidad de geometrías temporales. Activando el modo de creación sobre el mapa, puede crear un punto que represente la ubicación exacta del depósito propuesto o un polígono que represente el predio potencial. Estas geometrías se mantienen en memoria durante la sesión y se utilizan directamente como insumo para las funciones de análisis.

La función de **Área y perímetro** permite calcular la superficie del predio del depósito, validando que cumpla con los requerimientos mínimos de tamaño para playa de maniobras y almacenamiento.

Análisis de cobertura y autonomía de los camiones

Para modelar la autonomía de los camiones eléctricos (por ejemplo: 30 km), el analista aplica la función de **Buffer** sobre el sitio candidato, generando un área circular de 30 km de radio.

La Figura 7.2 ilustra el resultado del análisis de cobertura mediante buffers alrededor del centro de distribución candidato y las sucursales atendidas.

A partir de este buffer, se utiliza la intersección espacial para identificar todas las sucursales que quedan dentro del radio de 30 km y determinar qué departamentos quedan cubiertos. También se aplica un análisis de proximidad que calcula para cada sucursal cubierta sus tres vecinos más cercanos ($N{=}3$), con el fin de proponer agrupaciones coherentes para las rutas de reparto.

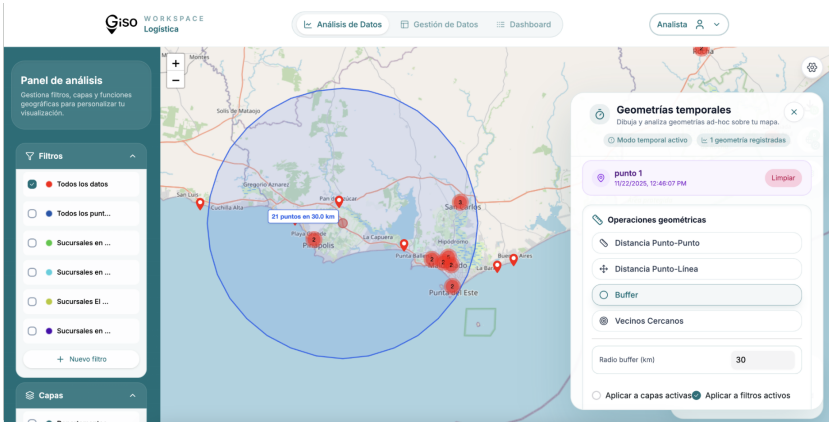


Figura 7.2: Ejemplo de áreas de cobertura (buffers) alrededor de un centro de distribución candidato y sucursales atendidas.

Análisis respecto a rutas principales

En el caso de estudio, la accesibilidad a la red vial es importante, por lo que se utiliza la capa de rutas nacionales. En primer lugar, se aplica la función de distancia punto-línea entre el depósito y la capa de rutas, verificando que la ubicación propuesta esté próxima a una ruta principal. Luego se genera un buffer lineal alrededor de la ruta. Al intersectar este buffer con la capa de sucursales y el buffer del depósito, se analizan los recorridos más eficientes que combinan proximidad al centro de distribución y facilidad de tránsito por rutas nacionales.

7.1.4. Análisis de resultados

La ejecución muestra que la plataforma procesa el conjunto de 400 sucursales sin demoras perceptibles. El uso de geometrías temporales agiliza las etapas tempranas de planificación al permitir explorar ubicaciones sin crear capas permanentes. Las funciones espaciales integradas permiten responder de forma integral a preguntas de cobertura, autonomía y accesibilidad vial, facilitando que analistas no expertos en GIS tomen decisiones informadas sobre la ubicación de infraestructuras críticas.

El segundo caso de estudio se centra en un dominio diferente con el objetivo de demostrar que las mismas herramientas de análisis espacial se adaptan a contextos dispares y a volúmenes de datos significativamente mayores.

7.2. Caso 2: Análisis geoespacial en sanidad vegetal (Picudo Rojo)

Este caso explora la aplicación de GISO 2.0 en el monitoreo del picudo rojo (*Rhynchophorus ferrugineus*) en palmeras de Uruguay, tratando la sanidad vegetal como un subdominio de la vigilancia epidemiológica.

7.2.1. Escenario y datos

Se utiliza un conjunto de datos de **65.661 registros** proveniente del sistema ciudadano `Picudo Rojo Uruguay` (Uruguay, 2025). Estos registros contienen ubicaciones de palmeras ya georreferenciadas en el sistema de origen, por lo que no requieren geocodificación y se cargan directamente como puntos con coordenadas. El objetivo es identificar focos activos y zonas de riesgo de dispersión biológica.

7.2.2. Flujo de análisis epidemiológico

En primer lugar se realiza una identificación de focos mediante un análisis de proximidad (k=3), que permite detectar clústeres de palmeras afectadas. Distancias menores a 50 metros sugieren una alta densidad local de la plaga. Luego se generan buffers de 500 m, 1 km y 2 km alrededor de las palmeras infectadas para delimitar zonas de riesgo según la capacidad de vuelo del insecto. Por último, se calcula la distancia punto-línea respecto a la red vial nacional, evaluando cómo las carreteras facilitan la dispersión del insecto a través del transporte de material vegetal infestado.

La Figura 7.3 presenta un ejemplo del análisis de distancia punto-línea entre focos de infección y la red vial nacional.

7.2.3. Resultados del caso

La validación con este volumen de datos demuestra la robustez de la arquitectura. Las mismas herramientas aplicadas en el caso logístico (buffers, distancias) permiten aquí caracterizar el avance de una plaga, confirmando la versatilidad de la plataforma para múltiples dominios de análisis territorial.

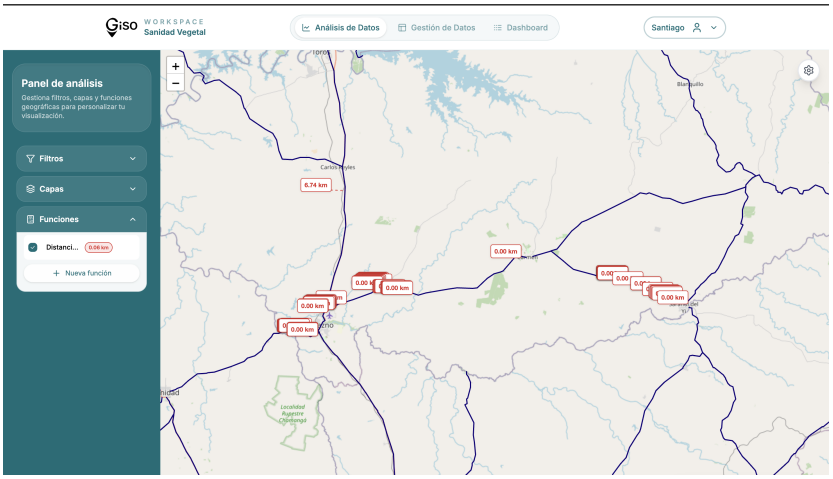


Figura 7.3: Análisis de distancia punto-línea entre palmeras infectadas y la red vial nacional.

Capítulo 8

Conclusiones y trabajo futuro

Este capítulo presenta la gestión del proyecto, las conclusiones del trabajo realizado y las líneas de mejora y trabajo futuro. En primer lugar se describe la planificación y ejecución del cronograma, proporcionando contexto sobre el proceso de desarrollo. A continuación se evalúan los resultados en relación con los objetivos planteados y, por último, se identifican oportunidades de evolución de la plataforma.

8.1. Gestión del proyecto

La gestión del proyecto se organizó mediante un cronograma por etapas, con actividades alineadas a los objetivos específicos y a los entregables del informe. El cronograma planificado consideró ocho meses, estructurados en fases de estudio inicial del dominio y definición de requerimientos, diseño de la solución con pruebas de concepto, definición de pruebas de validación, implementación, y finalmente pruebas del sistema y documentación.

La Figura 8.1 muestra el Gantt planificado para distribuir el trabajo y fijar hitos intermedios. El plan original contemplaba el período de abril a noviembre.

En la ejecución no se siguió estrictamente una secuencia por fases. La implementación comenzó a partir del segundo mes, ya que en la etapa de estudio se realizó la instalación de GISO 1.0 para familiarizarse con la aplicación, lo que hizo que se comenzaran algunos cambios en etapas tempranas. El análisis y refinamiento de requerimientos se realizó de forma iterativa durante el desarrollo, ajustando decisiones y alcance conforme se avanzaba. Este enfoque iterativo permitió mayor adaptabilidad, pero dificultó el seguimiento del cronograma original.

Por otra parte, se observó una diferencia relevante en el tramo final. Aunque el plan apuntaba a cerrar el desarrollo hacia noviembre, en línea con el cierre del curso de posgrado en diciembre, el proyecto se extendió hasta febrero. La

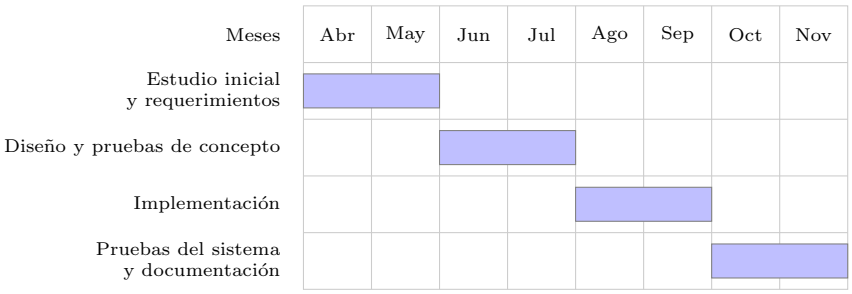


Figura 8.1: Cronograma planificado del proyecto (abril–noviembre).

extensión se debió principalmente a la coordinación de la validación con usuarios (el curso de posgrado finalizaba en diciembre) y a la decisión de refinar la documentación de todo el proyecto. Además, esta extensión permitió concretar una segunda instancia de validación con el contacto generado a partir de Ingeniería DeMuestra (IdM).

La Figura 8.2 presenta el Gantt ejecutado, en el que se refleja la extensión del período de pruebas y cierre del informe.

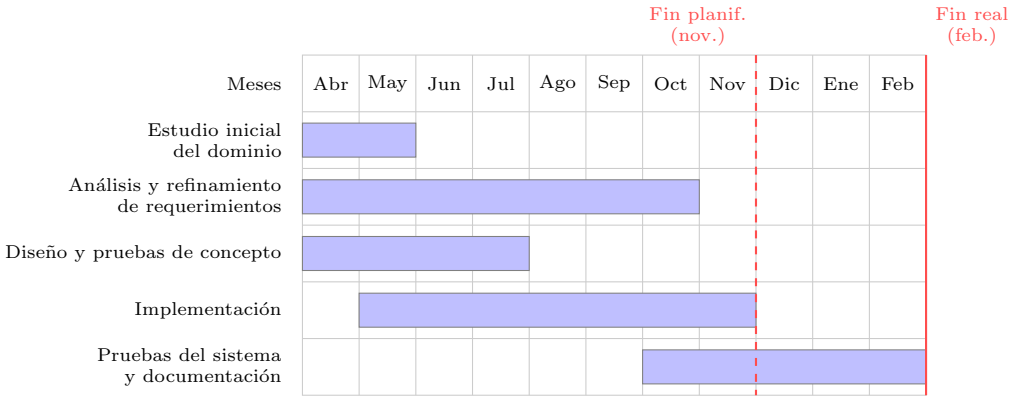


Figura 8.2: Cronograma ejecutado del proyecto (abril–febrero).

Esta extensión permitió disponer de más tiempo para coordinar la participación de usuarios potenciales, analizar los resultados de la validación e incorporar ajustes de usabilidad y claridad en los flujos principales. Además, se aprovechó el período adicional para revisar la consistencia del documento y completar la documentación técnica vinculada a instalación y operación.

8.2. Conclusiones

Este proyecto (GISO 2.0) continúa la línea de trabajo iniciada con GISO 1.0 y busca reducir la dependencia de especialistas en GIS para tareas habituales de geocodificación masiva y análisis espacial. La plataforma integra un flujo guiado de carga de datos, validación, geocodificación asistida y análisis sobre mapa, manteniendo un enfoque centrado en analistas no expertos en GIS.

Respecto del objetivo general, la plataforma amplía la estrategia de geocodificación con un nuevo servicio y enriquece el repertorio de operaciones espaciales disponibles. Además, se incorpora un despliegue web en MiNube y documentación asociada, con el fin de facilitar la operación sostenida del sistema y su evaluación en escenarios de uso reales.

Cumplimiento de objetivos

Respecto de los objetivos específicos definidos en el Capítulo 1, el proyecto aborda y cumple cada uno de ellos:

1. **Estudiar el dominio de aplicación y trabajos previos**: se realiza una revisión del estado del arte y se toma como línea base el proyecto GISO 1.0 ejecutado en 2024 (Capítulo 2).

2. **Definir requerimientos y plan de validación**: se definen requerimientos funcionales y no funcionales, junto con un plan de validación con usuarios que orienta las etapas posteriores del desarrollo (Capítulo 3, Capítulo 6).

3. **Optimizar la arquitectura de la plataforma**: se prioriza una estructura extensible y mantenible, unificando los dos frontends de GISO 1.0 en una única aplicación con control de acceso por roles (Capítulo 4).

4. **Extender la estrategia de geocodificación**: se incorpora Google Maps como geocodificador adicional, con criterios de selección configurables por workspace (Capítulo 5).

5. **Implementar funcionalidades de análisis espacial y experiencia de uso**: se integran operaciones de buffer, intersección, proximidad, distancia punto-línea y área/perímetro, junto con geometrías temporales y mejoras de navegación (Capítulo 5).

6. **Definir y ejecutar el despliegue web en MiNube**: se documenta un procedimiento reproducible de configuración y operación, facilitando el acceso de usuarios y la validación del sistema (Capítulo 5).

7. **Validar la plataforma con usuarios potenciales**: se realizan pruebas funcionales, dos casos de estudio en dominios dispares y validaciones tanto con estudiantes de posgrado como con una empresa del medio, obteniendo datos de usabilidad y desempeño (Capítulos 6 y 7).

Limitaciones y aprendizajes

Durante la instancia de validación con usuarios se identifican aspectos de usabilidad y operación que se pueden mejorar del uso de la plataforma. En concreto, se observa que el proceso de aprendizaje requiere apoyo adicional (manuales y guías) y que existen oportunidades de mejora en varios frentes:

- **Concurrencia en la geocodificación asíncrona:** cuando un participante cargó un lote de gran volumen, el resto experimentó demoras en el procesamiento de sus direcciones, evidenciando la necesidad de políticas de equidad en la cola de mensajes.
- **Retroalimentación durante la carga de archivos CSV:** los mensajes resultaron insuficientes para algunos usuarios, que no pudieron identificar con claridad qué columnas se ignoraron ni el motivo.
- **Curva de aprendizaje de geometrías temporales:** aunque la funcionalidad fue valorada positivamente por quienes la utilizaron, su adopción sugiere la necesidad de ayudas integradas en la interfaz.

Estas observaciones orientan las líneas de trabajo futuro descritas a continuación.

8.3. Mejoras y trabajo futuro

Al realizarse la prueba con usuarios se identifican oportunidades de mejora que se priorizan según su impacto en usabilidad, operación y escalabilidad. A continuación se describen las principales líneas de trabajo futuro.

Mejoras de usabilidad y documentación

En el ámbito de la experiencia de usuario, se detecta la necesidad de reforzar el proceso de onboarding mediante manuales más detallados, guías paso a paso y videos breves orientados a nuevos usuarios. Aún con orientación inicial, la plataforma puede resultar compleja en los primeros usos, por lo que se recomienda incorporar ayudas contextuales y material de entrenamiento integrado en la interfaz.

En cuanto a la gestión de workspaces, se propone revisar el modelo de invitaciones y administración para que las operaciones críticas queden explícitamente asociadas al rol de administrador y la gestión de permisos resulte clara y trazable. Adicionalmente, se sugiere habilitar mecanismos para reiniciar un workspace o reasignarlo a un administrador, así como permitir la edición de atributos cuando aún no se cargan puntos, reduciendo la fricción en etapas tempranas de configuración.

Escalabilidad y procesamiento

Desde la perspectiva de escalabilidad, se identifica un factor limitante en el procesamiento de la cola de geocodificación cuando un usuario carga un lote grande y el resto experimenta demoras. Para mitigar este problema se recomienda aumentar el paralelismo (por ejemplo, mediante más *workers* o *threads*) y considerar políticas de equidad por workspace o usuario que eviten la monopolización del procesamiento por parte de un único lote.

En relación con la infraestructura, se propone evaluar alternativas para eliminar la dependencia de WildFly (contenedor de aplicaciones Java) como infraestructura de cola. Un reemplazo por componentes más livianos o servicios administrados simplifica la operación, reduce el acoplamiento y facilita el escalado del procesamiento asíncrono. Asimismo, se recomienda reducir el *downtime* actual mediante la automatización de builds, despliegues y migraciones, instrumentando un pipeline de integración continua y despliegue continuo (CI/CD) con estrategias de despliegue gradual.

Extensiones funcionales

En cuanto a funcionalidades adicionales, se plantea incorporar la posibilidad de crear registros directamente desde la interfaz (sin carga por CSV) y evaluar la persistencia opcional de geometrías temporales como capas reutilizables, permitiendo pasar de análisis exploratorios a escenarios repetibles.

Otra línea de trabajo relevante consiste en incorporar soporte para distinguir, dentro de un mismo workspace, diferentes tipos de datos (por ejemplo, sucursales y depósitos) sin depender exclusivamente de metadatos poco expresivos. Esta necesidad puede resolverse mediante tipos de sujeto, colecciones separadas o capas con semántica explícita.

Finalmente, se propone permitir el consumo de capas externas mediante servicios Web Map Service (WMS), lo cual posibilita incorporar capas publicadas por organismos (por ejemplo, GeoPortales) directamente, evitando la carga y mantenimiento manual de *shapefiles* cuando la información ya se encuentra publicada. Esta posible extensión surgió de los intercambios realizados durante las pruebas con la empresa del medio.

En resumen, GISO 2.0 muestra que es viable ofrecer capacidades de geocodificación masiva y análisis espacial interactivo a través de una plataforma web accesible para analistas sin formación especializada en GIS. Las extensiones implementadas, la validación con usuarios y los casos de estudio en dominios dispares confirman la flexibilidad del enfoque adoptado y sientan las bases para continuar la evolución de la plataforma.

Referencias

Antel. (2025a). *Elastic cloud – plataforma como servicio.* https://minubeantel.uy/index.php?NAME_PATH=Elastic_Cloud. (Último acceso: febrero de 2026)

Antel. (2025b). *Mi nube – servicios cloud.* https://minubeantel.uy/. (Último acceso: febrero de 2026)

Apache echarts documentation [Manual de software informático]. (2025). https://echarts.apache.org/en/index.html. (Último acceso: febrero de 2026)

Bernabé Poveda, M. A., y López Vázquez, C. M. (2012). *Fundamentos de las infraestructuras de datos espaciales (ide).* Madrid, España: UPM Press.

de Datos Espaciales del Uruguay, I. (2024). *Plataformas y visualizadores.* https://visualizador.ide.uy. (Último acceso: febrero de 2026)

Docker, Inc. (2025). Docker documentation [Manual de software informático]. https://docs.docker.com/. (Último acceso: febrero de 2026)

EPSG.io. (2022a). *Epsg:32721 – wgs 84 / utm zone 21s.* https://epsg.io/32721. (Último acceso: febrero de 2026)

EPSG.io. (2022b). *Epsg:32722 – wgs 84 / utm zone 22s.* https://epsg.io/32722. (Último acceso: febrero de 2026)

EPSG.io. (2022c). *Epsg:4326 – wgs 84.* https://epsg.io/4326. (Último acceso: febrero de 2026)

Geoserver user manual [Manual de software informático]. (2025). https://docs.geoserver.org/2.24.x/en/user/. (Último acceso: febrero de 2026)

Goldberg, D. W., Wilson, J. P., y Knoblock, C. A. (2007). From text to geographic coordinates: The current state of geocoding. *URISA Journal*, *19*(1), 33–46.

Google. (2025). *Geocoding api – google maps platform.* https://developers.google.com/maps/documentation/geocoding. (Último acceso: febrero de 2026)

Hardt, D. (2012). *The oauth 2.0 authorization framework* (Inf. Téc. n.º RFC 6749). Internet Engineering Task Force (IETF). https://www.rfc-editor.org/rfc/rfc6749. (Último acceso: febrero de 2026)

Huang, Z., y Shao, Y. (2020). Microservice architecture-based distributed geographic information system. *Journal of Geographic Information System*, *12*(5), 556–568.

Infraestructura de Datos Espaciales del Uruguay. (2025). *Sistema único de di-*

recciones del uruguay (sudir) – recursos ideuy. https://catalogodatos.gub.uy/dataset/ide-direcciones-geograficas-del-uruguay. (Último acceso: febrero de 2026)

Jakarta messaging specification (Inf. Téc.). (2023). Eclipse Foundation. https://jakarta.ee/specifications/messaging/. (Último acceso: febrero de 2026)

Karimi, H. A., y Sharker, M. H. (2019). Geocoding techniques and applications: a review. *International Journal of Geographical Information Science*, *33*(5), 874–895.

Leaflet reference (v1.9) [Manual de software informático]. (2025). https://leafletjs.com/reference.html. (Último acceso: febrero de 2026)

Longley, P. A., Goodchild, M. F., Maguire, D. J., y Rhind, D. W. (2011). *Geographic information systems and science* (3.ª ed.). Chichester, UK: John Wiley and Sons.

Next.js documentation [Manual de software informático]. (2025). https://nextjs.org/docs. (Último acceso: febrero de 2026)

Nominatim – search api [Manual de software informático]. (2025). https://nominatim.org/release-docs/latest/api/Search/. (Último acceso: febrero de 2026)

Postgis manual (v3.4) [Manual de software informático]. (2025). https://postgis.net/documentation/. (Último acceso: febrero de 2026)

QGIS Development Team. (2024). *Qgis - a free and open source geographic information system.* https://www.qgis.org/. (Último acceso: febrero de 2026)

React documentation [Manual de software informático]. (2025). https://react.dev/learn. (Último acceso: febrero de 2026)

React-leaflet documentation [Manual de software informático]. (2025). https://react-leaflet.js.org/. (Último acceso: febrero de 2026)

Sediey, V., y Segura, V. (2024). *Plataforma de geocodificación asistida y análisis geoespacial.* https://www.colibri.udelar.edu.uy/jspui/handle/20.500.12008/48512. (Informe de Proyecto de Grado, Facultad de Ingeniería, UdelaR)

Spring boot reference documentation [Manual de software informático]. (2025). https://docs.spring.io/spring-boot/docs/current/reference/html/. (Último acceso: febrero de 2026)

Turf.js documentation [Manual de software informático]. (2025). https://turfjs.org/. (Último acceso: febrero de 2026)

Uruguay, P. R. (2025). *Picudo rojo uruguay: datos ciudadanos.* https://picudorojo.uy/. (Último acceso: febrero de 2026)

Zhang, C., Luo, L., y Xu, W. (2021). Spatial analytics for epidemiological studies using gis: a systematic review. *ISPRS International Journal of Geo-Information*, *10*(6), 383.

Zustand documentation [Manual de software informático]. (2025). https://docs.pmnd.rs/zustand. (Último acceso: febrero de 2026)