

Componente 2. Entregable 2.6.17 Ministerio de Minas y Energía

Proyecto CTO No. 116231-225-2024

Documentación técnica del proyecto

Fecha: julio de 2025
Elaborado por, ADA S.A.S.
Versión: 7.0

Contenido

1. Introducción	7
2. Objetivo	7
3. Definición del Proyecto	7
4. Objetivos del Aplicativo web.....	8
4.1 Objetivo General	8
4.2 Objetivos Técnicos Específicos.....	8
5. Especificaciones Funcionales	9
5.1 Interfaz de Usuario.....	9
5.2 Visualización de Datos	10
5.3 Comparación de Información	11
5.4 Generación de Reportes	11
5.5 Sistema de Selección.....	12
6. Especificaciones No Funcionales.....	15
6.1 Regulaciones y Usabilidad	15
6.2 Compatibilidad	16
6.3 Adaptabilidad	17
6.4 Pruebas funcionales.....	20
6.4.1 Escenario 1: Consulta de determinantes e indicadores sin comparación.	20
6.4.2 Escenario 2: Consulta de determinantes e indicadores con comparación.	23
6.4.3 Escenario 3: Consulta manual.....	26
6.4.4 Escenario 4: Consulta por coordenadas.....	27
7. Validación de Requisitos	28
7.1 Acceso al Sistema.....	28
7.2 Visualización de Información.....	28
7.3 Reportes Generados.....	29
7.4 Exportación de Datos.....	29
8. Arquitectura y Reglas del Sistema	30
8.1 Infraestructura de Bases de Datos	30
8.2 Transformación de Datos	31
8.3 Representación de Información.....	31
8.4 Configuración de Reglas	32
9. Procesamiento y Visualización.....	33

9.1 Descripción del almacenamiento.....	33
9.1.1 Base de datos de almacenamiento primario.....	33
9.2 Estructura de los datos	33
9.2.1 Tablas normalizadas.....	33
9.2.2 Separación de datos operativos y de referencia	34
9.2.3 Almacenamiento de archivos fuente	34
9.3 Mecanismo de versionado de datasets	34
9.3.1 Fecha de carga	34
9.3.2 Origen de los datos	34
9.3.3 Usuario responsable	34
9.3.4 Hash de los archivos	34
9.4 Detalle control de Versionamiento herramientas usadas para el aplicativo web	35
10. Procesamiento y Visualización.....	35
10.1 Descripción del Proceso.....	36
10.1.1 Backend (Node.js / Python).....	36
10.1.2 Frontend (React.js).....	36
10.2 Ejemplo del flujo de visualización	37
10.3 Ejemplo Diagrama de Flujo Carga de Datos.	38
10.4 Interacciones destacadas.....	39
11. Entrega y Despliegue del Sistema.....	39
11.1 Requisitos mínimos para el correcto funcionamiento del aplicativo web.	39
11.2 Cuenta de Docker	40
11.3 Puertos de red habilitados.....	40
11.4 Muestra del proceso de despliegue.	40
11.5 Entrega del Software	41
11.5.1 Protocolo de la Entrega Técnica	42
11.5.2 Propuestas de CI/CD.....	42
11.6 Requerimientos del Entorno de Despliegue (MME)	42
12. Diccionario de datos	43
12.1 Tabla: spatial_ref_sys.....	43
12.2 Tabla: municipios.....	44
12.3 Tabla: div_territorial_municipios	45
12.4 Tabla: div_territorial_zonas.....	46

12.5 Tabla: divterritorialdepto_determinantes	47
12.6 Tabla: divterritorialmuni_determinantes	50
12.7 Tabla: indicadores_territorial.....	50
13. Código Fuente del Desarrollo	53
14. Garantía sobre el Aplicativo web.....	53
14.1 Condiciones de la Garantía	54
14.1.1 Plazo de cobertura	54
14.1.2 Cobertura	54
14.2 Procedimiento para la Gestión de Incidencias.....	54
14.2.1 Reporte de incidentes	54
14.2.2 Reproducción del error.....	55
14.2.3 Solución y entrega.....	55
14.3 Documentación de garantía	55
15. Aprobación.....	56

Índice de tablas

Tabla 1 Tabla: spatial_ref_sys Fuente: ADA.	44
Tabla 2 Tabla: municipios, Fuente: ADA.	44
Tabla 3 Tabla: div_territorial_municipios, Fuente: ADA.	45
Tabla 4 Tabla: div_territorial_zonas, Fuente: ADA.	47
Tabla 5 Tabla: divterritorialdepto_determinantes, Fuente: ADA.	50
Tabla 6 Tabla: divterritorialmuni_determinantes, Fuente: ADA.	50
Tabla 7 Tabla: indicadores_territorial, Fuente: ADA.	53

Índice de ilustraciones

Ilustración 1. Ejemplo generación de documento PDF. Fuente: Imagen Elaborada por ADA	12
Ilustración 2. Sistema de Selección. Fuente: Imagen elaborada por ADA.....	13
Ilustración 3. Determinantes. Fuente: Imagen elaborada por ADA.....	14
Ilustración 4. Indicadores. Fuente: Imagen elaborada por ADA.....	14
Ilustración 5. Prueba responsividad. Fuente: Imagen elaborada por ADA.....	18
Ilustración 6. Pruebas de responsividad. Fuente: Imagen elaborada por ADA.	19
Ilustración 7. Pruebas de responsividad. Fuente: Imagen elaborada por ADA.	19
Ilustración 8. Pruebas de funcionamiento determinantes. Fuente: Imagen elaborada por ADA.	20
Ilustración 9. Pruebas de funcionamiento exportar determinantes. Fuente: Imagen elaborada por ADA.	21
Ilustración 10. Pruebas de funcionamiento indicadores. Fuente: Imagen elaborada por ADA.	21
Ilustración 11. Pruebas de funcionamiento exportar indicadores. Fuente: Imagen elaborada por ADA.	22
Ilustración 12. Pruebas de funcionamiento exportar mapa en PDF. Fuente: Imagen elaborada por ADA.	22
Ilustración 13. Pruebas de funcionamiento indicadores botón limpiar. Fuente: Imagen elaborada por ADA.	23
Ilustración 14. Pruebas de funcionamiento determinantes botón limpiar. Fuente: Imagen elaborada por ADA.	23
Ilustración 15. Pruebas de funcionamiento determinantes. Fuente: Imagen elaborada por ADA.	24
Ilustración 16. Pruebas de funcionamiento indicadores. Fuente: Imagen elaborada por ADA.	25
Ilustración 17. Pruebas de funcionamiento exportar indicadores. Fuente: Imagen elaborada por ADA.	25

Ilustración 18. Pruebas de funcionamiento indicadores botón limpiar. Fuente: Imagen elaborada por ADA.	26
Ilustración 19. Pruebas de funcionamiento Selección Manual. Fuente: Imagen elaborada por ADA.	26
Ilustración 20. Pruebas de funcionamiento Selección Manual. Fuente: Imagen elaborada por ADA.	27
Ilustración 21. Funcionalidad X, Y. Fuente: ADA	27
Ilustración 22. Representación de Información. Fuente: Imagen elaborada por ADA	32
Ilustración 23. Configuración de Reglas. Fuente: Imagen elaborada por ADA	32
Ilustración 24. Control de versionamiento. Fuente: imagen elaborada por ADA	35
Ilustración 25. Descripción del Proceso. Fuente: Imagen elaborada por ADA	36
Ilustración 26. Ejemplo del flujo de visualización. Fuente: Imagen elaborada por ADA	37
Ilustración 27. Diagrama de flujo carga de datos	38
Ilustración 28. Proceso de despliegue. Fuente: Imagen elaborada por ADA.	41

1. Introducción

Desarrollar un aplicativo web bajo el contrato CTO No. 116231-225-2024, suscrito entre ADA S.A.S. y el Ministerio de Minas y Energía (MME), que provea información abierta sobre indicadores y determinantes territoriales relevantes para la toma de decisiones en proyectos del sector minero-energético dentro del marco de la Transición Energética Justa - TEJ.

El sistema permite a los usuarios acceder a datos estructurados que se presentan de forma clara y comprensible, con el propósito de que cualquier persona pueda acceder a información de interés general del sector minero-energético. Además, garantiza que la aplicación sea altamente eficiente, proporcionando la información necesaria con un mínimo de costos operativos. Esto se logra mediante la integración de herramientas avanzadas de búsqueda, filtrado y análisis de datos, que permiten a los usuarios generar informes, realizar consultas detalladas y obtener visualizaciones interactivas, todo dentro de una interfaz de usuario amigable y de fácil acceso.

Los datos provendrán exclusivamente de matrices en Excel y Shapefile de municipios, departamentos y polígonos suministrados por el Ministerio de Minas y Energía - MME.

2. Objetivo

Este documento presenta un análisis detallado del proceso de diseño y construcción colaborativa del aplicativo web del Ministerio de Minas y Energía (MME). Esta herramienta digital nace como una respuesta a la necesidad de contar con un canal accesible, moderno y eficiente que permita consultar información clave relacionada con los principales indicadores y determinantes del sector minero-energético en Colombia. En este contexto, el aplicativo web se convierte en un instrumento fundamental para promover y respaldar la estrategia nacional de Transición Energética Justa, una política que impulsa el desarrollo sostenible y equitativo del país.

El desarrollo del aplicativo se da en un entorno donde el acceso transparente y oportuno a los datos se vuelve esencial para la toma de decisiones, tanto a nivel institucional como ciudadano. Por eso, el diseño del sistema prioriza la agilidad, la usabilidad y la capacidad de conectar múltiples fuentes de información bajo un mismo entorno visual. El resultado es una plataforma digital de libre acceso que no solo centraliza datos técnicos y geoespaciales, sino que también facilita su análisis, interpretación y divulgación a través de herramientas gráficas y mapas interactivos.

3. Definición del Proyecto

El proyecto consiste en el desarrollo de un aplicativo web de acceso público, diseñado específicamente para brindar una experiencia intuitiva y eficiente en la exploración, visualización y análisis de datos relevantes del sector energético y minero. A través de un conjunto de herramientas de búsqueda avanzada y filtros dinámicos, el sistema permite a los usuarios interactuar con grandes volúmenes de información estructurada, garantizando una navegación fluida y orientada a la obtención de conocimientos útiles.

El núcleo funcional del aplicativo se centra en facilitar la consulta interactiva de información geoespacial y estadística del sector minero-energético, alineado con la Transición Energética Justa. El sistema integra datos estructurados desde matrices Excel y archivos Shapefile del MME, permitiendo visualizaciones en mapas (para determinantes territoriales) y tablas dinámicas (para determinantes e indicadores socioeconómicos). Los usuarios pueden aplicar filtros avanzados mediante acordeones jerárquicos -con opciones de selección múltiple y marcado total-, comparar hasta dos entidades territoriales (departamentos/municipios), y generar reportes en PDF con mapas integrados o exportar datos tabulares a Excel. Cumple con normativas de Gobierno Digital en accesibilidad, usabilidad y seguridad, incluyendo cifrado de credenciales, y ofrece un diseño responsivo compatible con dispositivos móviles y navegadores modernos. La plataforma combina funciones analíticas con una interfaz intuitiva, democratizando el acceso a información estratégica para la toma de decisiones técnicas y ciudadanas.

4. Objetivos del Aplicativo web

4.1 Objetivo General

Diseñar e implementar una plataforma web pública altamente interactiva que permita al Ministerio de Minas y Energía (MME), así como al público en general, consultar, analizar, visualizar y reportar información estratégica relacionada con la Transición Energética Justa. El aplicativo busca convertirse en un ecosistema digital accesible, abierto y confiable para la gestión de datos geoespaciales y estadísticos, provenientes de diversas fuentes, bajo criterios de precisión, claridad y eficiencia técnica. Su enfoque está centrado en facilitar la toma de decisiones, promover la transparencia y fomentar el acceso libre al conocimiento sobre el sector minero-energético.

4.2 Objetivos Técnicos Específicos

- **Integración automatizada de fuentes de datos:** Se establece un sistema que permite conectar y validar múltiples fuentes de información en formatos como Excel y Shapefile. Estos procesos automáticos aseguran la consistencia de los datos y permiten su actualización periódica sin intervención manual extensiva.
- **Diseño de una interfaz responsiva y moderna:** La plataforma incorpora una interfaz desarrollada en React.js con soporte para dispositivos móviles y de escritorio, utilizando Tailwind CSS como base para garantizar un diseño visual limpio, adaptable y funcional.

- **Visualización interactiva de datos:** Se implementan bibliotecas gráficas como Chart.js y Recharts para generar visualizaciones estadísticas, complementadas con mapas interactivos contruidos con la herramienta Leaflet.js. Estas herramientas permiten explorar la información a diferentes niveles territoriales, ofreciendo comparaciones visuales entre municipios, departamentos y/o regiones. Las versiones de las herramientas usadas son gratuitas y de código abierto.
- **Generación de reportes dinámicos:** Los usuarios pueden crear y descargar reportes personalizados en formatos PDF y Excel, mediante el uso de librerías como jsPDF y SheetJS. Esta funcionalidad permite exportar información de manera estructurada y reutilizable para diversos fines institucionales o investigativos.
- **Sistema de filtrado avanzado:** Se habilita un sistema de filtros inteligentes, con criterios recursivos y búsquedas en tiempo real, que mejora significativamente la experiencia de consulta. Esta funcionalidad es clave para permitir un análisis detallado y específico de los datos.
- **Comparación entre unidades territoriales:** Incorpora mecanismos que permiten comparar determinantes e indicadores entre diferentes niveles territoriales, ya sea entre municipios o departamentos. Esta capacidad de análisis paralelo aporta una dimensión comparativa muy útil para la planificación y el seguimiento de políticas públicas.
- **Cumplimiento de estándares de calidad:** El desarrollo se alinea con los principios de accesibilidad digital, garantizando que la plataforma pueda ser utilizada por personas con diferentes capacidades. Además, se asegura el buen desempeño en múltiples navegadores y dispositivos, respetando las buenas prácticas de desarrollo.

5. Especificaciones Funcionales

5.1 Interfaz de Usuario

La interfaz de usuario se diseña con el objetivo de ofrecer una experiencia clara, intuitiva y eficiente. Para lograrlo, se utilizan tecnologías modernas del ecosistema frontend, específicamente **React.js** como biblioteca principal para la construcción de componentes dinámicos y reutilizables, y **Tailwind CSS** para aplicar estilos de manera rápida, coherente y altamente personalizable.

Dentro de esta estructura, se destacan varios componentes clave que definen la organización visual y funcional del sistema:

- Una **barra de navegación superior**, que permanece visible durante el uso del aplicativo y facilita el acceso directo a los distintos módulos del sistema. Esta barra incluye enlaces organizados que guían al usuario a través de las secciones principales sin generar confusión.

- Un **panel lateral izquierdo**, que contiene una serie de filtros ajustables según el tipo de consulta que el usuario desee realizar. Este panel ofrece un control refinado sobre los datos a visualizar, permitiendo filtrar por región, tipo de indicador, fecha, entre otros parámetros disponibles.
- Un **área principal de visualización**, que actúa como el centro de análisis. En este espacio se renderizan de forma dinámica los distintos tipos de resultados: gráficos estadísticos, tablas interactivas o mapas temáticos, dependiendo del tipo de información consultada y los filtros aplicados.

La experiencia del usuario se cuida en todo momento con la inclusión de **validaciones de formularios** implementadas mediante **React Hook Form**, una librería eficiente que permite gestionar el estado de los formularios y aplicar reglas de validación de forma clara y reactiva. Además, se integran **indicadores de carga (spinners)** que informan al usuario cuando se está procesando una consulta, así como **mensajes de error contextualizados** que brindan retroalimentación clara en caso de fallos o datos faltantes.

5.2 Visualización de Datos

Uno de los pilares del aplicativo web es su capacidad para presentar datos complejos de forma clara, interactiva y visualmente comprensible. Para ello, se incorporan distintos mecanismos de visualización que se adaptan al tipo de información que se desea explorar:

- **Gráficos de líneas**, utilizados especialmente para comparar tendencias entre distintos territorios o indicadores. Estos gráficos se renderizan con componentes personalizables que permiten observar datos de los indicadores.
- **Mapas interactivos**, contruidos sobre tecnologías que soportan el uso de capas geoespaciales en formato **GeoJSON**. Cada mapa incluye una capa base sobre la cual se aplican sombras de color (sombreado temático) en función de los determinantes seleccionados. A su vez, se integran **tooltips dinámicos**, que muestran información detallada al dar clic sobre un polígono y se permite aplicar **zoom** para examinar los datos con mayor detalle.
- **Tablas de datos**, que permiten una exploración más precisa y detallada de los valores numéricos. Estas tablas incluyen funcionalidades como ordenación por columnas y filas, y se construyen utilizando librerías especializadas como **React Table**, garantizando un buen rendimiento incluso con grandes volúmenes de información.

Además de la visualización pasiva, se priorizan las **interacciones** dentro de cada componente. El usuario puede interactuar directamente con los elementos visuales, explorar los datos desde múltiples ángulos y obtener retroalimentación visual inmediata. Las **leyendas** también se generan dinámicamente y se ajustan automáticamente según los datos cargados y los filtros aplicados.

5.3 Comparación de Información

Dentro del módulo de comparación, se plantea una solución técnica que permite al usuario contrastar datos entre distintos territorios, facilitando el análisis comparativo entre regiones. Esta funcionalidad es particularmente útil para evaluar diferencias o similitudes entre departamentos y municipios en relación con los indicadores energéticos disponibles.

El sistema ofrece un componente visual diseñado para mostrar datos **lado a lado**, permitiendo seleccionar **hasta dos municipios y dos departamentos** de manera simultánea. Una vez realizada la selección, se cargan de forma sincronizada **gráficos, estadísticas y mapas**, lo que permite observar las diferencias entre los territorios en una misma vista.

Desde el backend, se habilitan **endpoints específicos** que reciben como entrada los **códigos DANE** correspondientes a cada territorio. A partir de esa solicitud, el sistema devuelve conjuntos de datos estructurados que alimentan las visualizaciones. Esta arquitectura permite mantener una lógica clara de separación entre la capa de presentación y el procesamiento de datos, garantizando tanto eficiencia como escalabilidad.

5.4 Generación de Reportes

Para complementar la exploración interactiva de los datos, se incorpora una funcionalidad robusta para la **generación de reportes automáticos**. Esta opción está disponible en todas las vistas del módulo de análisis, permitiendo al usuario descargar un resumen detallado de los datos visualizados.

El botón "**Generar Reporte**" se encuentra siempre visible dentro del área principal, y al hacer clic sobre él, el sistema permite elegir entre distintos **formatos de salida**, dependiendo de las necesidades del usuario:

- **Formato PDF**, generado utilizando la librería **jsPDF**, que permite construir documentos portables con alta calidad visual. El reporte incluye, los filtros aplicados, la fecha de generación, los gráficos renderizados tal como se presentan en la interfaz y la fuente suministrada por el Ministerio de Minas y Energía (MME). Este formato es ideal para compartir información consolidada con otros equipos o entidades.
- **Formato Excel**, construido a través de la librería **SheetJS**, que facilita la exportación de datos tabulares. Este archivo incluye tanto la tabla de datos visible en pantalla como una hoja de cálculo adicional con todos los datos sin procesar, lo que permite un análisis más detallado o el uso en sistemas externos.

A continuación, se muestra un ejemplo simple de generación de un documento PDF:

```
javascript

const doc = new jsPDF({
  orientation: "portrait",
  unit: "mm",
  format: "a4"
});

// Configuración inicial
doc.setFont("helvetica");
doc.setFontSize(16);
doc.text("Reporte TEJ - Departamento del Meta", 10, 10);

// Insertar gráfico
doc.drawImage(graficoRenderizado, 'PNG', 10, 20, 190, 150);

// Guardar PDF
doc.save("reporte_meta.pdf");
```

Ilustración 1. Ejemplo generación de documento PDF. Fuente: Imagen Elaborada por ADA

Cada reporte generado incluye un bloque de **metadatos** que aporta contexto adicional al análisis. Entre los datos que se registran se encuentran:

- La **fecha** de generación del reporte.
- El **departamento** y el **municipio** seleccionados en la consulta.
- La **fuentes oficial** de los datos: **Ministerio de Minas y Energía (MME)**.

Esta funcionalidad asegura que cualquier análisis o consulta realizada dentro del sistema pueda ser fácilmente documentada, compartida o archivada, apoyando la toma de decisiones basada en evidencia.

5.5 Sistema de Selección

El sistema de selección está diseñado para facilitar la identificación y elección de los determinantes dentro de la plataforma de manera rápida y eficiente. Utiliza un enfoque dinámico que permite la interacción directa con los usuarios mediante el uso de acordeones desplegables. Estos acordeones funcionan como un mecanismo interactivo que, al ser desplegado, presenta una lista organizada de opciones o categorías relacionadas con los determinantes disponibles en el sistema.

La flexibilidad de este sistema radica en su capacidad para adaptarse a las necesidades específicas de los usuarios en función del contexto de uso. Los acordeones desplegables permiten al usuario explorar de manera intuitiva diferentes opciones sin sobrecargar la interfaz, brindando un espacio limpio y organizado. Al seleccionar una opción, el sistema responde de forma inmediata, mostrando solo las categorías o subcategorías relevantes según la elección previa, lo que optimiza el flujo de trabajo y asegura que el proceso de selección sea ágil y preciso.

Este diseño dinámico y adaptativo no solo mejora la experiencia de usuario, sino que también facilita la navegación dentro de grandes volúmenes de datos o categorías complejas, permitiendo que los usuarios encuentren rápidamente los determinantes que buscan, sin perder tiempo en desplazamientos innecesarios o en opciones irrelevantes. De este modo, el sistema de selección optimiza tanto la eficiencia como la usabilidad, favoreciendo una interacción fluida y sin barreras para los usuarios.

Determinante - Categoría

1-Social ^

☐ Seleccionar todo

Nombre del Determinante	Selección
Resguardo Indígena	☐
Solicitud Coloniales - RIOC	☐
Solicitud Ancestrales - Dec 2333 de 2014	☐
Solicitud legalización	☐

Ilustración 2. Sistema de Selección. Fuente: Imagen elaborada por ADA

En el aplicativo web, se implementa un sistema de acordeones despleables que facilita la selección de diferentes elementos dentro de la plataforma. Este sistema permite al usuario navegar de manera sencilla y eficiente a través de varias opciones para elegir los datos que desea consultar o analizar. En primer lugar, el usuario tiene la opción de seleccionar entre dos categorías principales: **Determinantes** e **Indicadores**.

Cada una de estas categorías es fundamental para estructurar la consulta de información de manera detallada. Los **Determinantes** representan los factores clave que influyen los datos, proporcionando un marco de referencia para las siguientes selecciones. A continuación, se pueden elegir los **indicadores**, que permiten especificar con mayor precisión los elementos que se desean analizar dentro de los determinantes previamente

seleccionados, ofreciendo datos específicos y medibles relacionados con las variables seleccionadas.

Además, este sistema de acordeones está diseñado con filtros dinámicos que se adaptan de acuerdo con los datos cargados en el sistema. Es decir, conforme el usuario va seleccionando opciones, los filtros se actualizan automáticamente para reflejar solo aquellas opciones que son relevantes, garantizando que la búsqueda sea siempre pertinente y precisa. Esto permite una experiencia de usuario mucho más ágil, al reducir la posibilidad de errores y mejorar la efectividad de la consulta.

Seleccione Zona o Lista Recurrente

Seleccionar zona ▼

Estructura 1

Estructura 2

Estructura 3

País

PDET

Región Amazonía

Región Andina

Ilustración 3. Determinantes. Fuente: Imagen elaborada por ADA

INDICADORES SOCIOECONÓMICOS	
VARIABLE	Sel
[+] PIB por habitante	☐
[+] Actividades primarias	☐
[+] Actividades secundarias	☐
[+] Actividades terciarias	☐
[+] Pobreza Monetaria Extrema	☐
[+] Población Rural	☐
[+] Población Urbano	☐
[+] Población Total	☐
[+] NBI	☐
[+] IPM	☐
[+] Índice de Cobertura de Energía Eléctrica en la Cabecera Municipal	☐
[+] Cobertura de Energía Eléctrica en el Resto	☐
[+] Índice de Cobertura de Energía Eléctrica Total	☐
[+] Posición nacional según desempeño integral	☐
[+] Índice de Desarrollo Municipal	☐
[+] Cobertura efectiva de gas natural	☐
[+] Tasa de homicidios por cada 100.000 habitantes	☐
[+] Población en condición de miseria	☐
[+] Cobertura de acueducto	☐
[+] Cobertura de alcantarillado	☐

Ilustración 4. Indicadores. Fuente: Imagen elaborada por ADA.

La persistencia del filtro en **localStorage** es una funcionalidad que permite al sistema recordar y conservar las preferencias de filtrado de los usuarios incluso después de que la sesión haya finalizado o la página haya sido recargada. Este comportamiento se logra mediante el uso de **localStorage**, un mecanismo de almacenamiento web que guarda datos de manera local en el navegador del usuario.

Cuando un usuario aplica un filtro para modificar la visualización de los datos dentro de la plataforma, el sistema almacena esta preferencia en **localStorage**. Este proceso asegura que, si el usuario regresa al sitio o recarga la página, el filtro seleccionado previamente se mantenga activo, ofreciendo una experiencia de usuario más fluida y personalizada. De esta forma, no es necesario que el usuario vuelva a configurar los filtros cada vez que interactúa con la plataforma.

La clave de esta funcionalidad radica en la capacidad de **localStorage** de guardar los datos de manera persistente sin la necesidad de almacenar información en el servidor. Al ser independiente de la sesión del usuario, la persistencia de los filtros no se ve afectada por el cierre del navegador o por el tiempo que transcurra entre sesiones. Esto ofrece una gran ventaja en términos de rendimiento, ya que no requiere que el servidor gestione y recupere estos datos cada vez que el usuario interactúa con la aplicación.

Además, esta persistencia facilita la experiencia de navegación al reducir la necesidad de configurar opciones repetitivamente, lo que puede resultar tedioso, especialmente en aplicaciones que manejan grandes volúmenes de datos. Los usuarios pueden disfrutar de una interfaz más cómoda y adaptada a sus necesidades, optimizando su tiempo de uso del sistema.

En resumen, la persistencia del filtro en **localStorage** no solo mejora la interacción del usuario con la plataforma al conservar sus preferencias, sino que también optimiza el rendimiento general de la aplicación al reducir la dependencia de las configuraciones de filtro en el servidor.

6. Especificaciones No Funcionales

6.1 Regulaciones y Usabilidad

Es fundamental garantizar que el aplicativo web cumpla con las regulaciones de accesibilidad y usabilidad digital para ofrecer una experiencia inclusiva y eficiente a todos los usuarios, sin importar sus capacidades o el dispositivo que utilicen. Con este objetivo en mente, se ha implementado una serie de prácticas y herramientas que aseguran que la navegación sea totalmente accesible y funcional:

- **Accesibilidad de la navegación mediante teclado:** La navegación dentro del sitio web se ha diseñado para ser completamente accesible mediante el uso del teclado. Esto se logra mediante la implementación de atributos como `tabindex`, `aria-labels` y roles adecuados en los elementos interactivos. Así, cualquier usuario puede navegar

por el sitio sin la necesidad de un ratón, garantizando que personas con diversas discapacidades motoras puedan interactuar de manera efectiva con la plataforma.

- **Validación del contraste visual:** Para asegurar una correcta visualización del contenido, especialmente para usuarios con dificultades de visión, se ha realizado una validación exhaustiva del contraste visual de la interfaz. Esto se ha llevado a cabo mediante herramientas como **Lighthouse** y **axe-core**, que son esenciales para verificar que el contraste entre los textos y los fondos cumpla con los estándares recomendados para una buena legibilidad.
- **Pruebas de usabilidad:** La experiencia de usuario (UX) ha sido puesta a prueba en diversas fases del proceso de desarrollo, comenzando con prototipos en **Figma** y luego validada en la implementación con **React**. Estas pruebas de usabilidad permiten identificar y resolver posibles problemas en la interacción del usuario con la interfaz, asegurando que la plataforma sea intuitiva y fácil de usar.
- **Funcionamiento del Backend:** El aplicativo está dividido en dos partes principales: Backend y FrontEnd. El Backend está hecho en Python exponiendo endPoints para ser llamados desde el FrontEnd. En el Backend se tiene la conexión a la base de datos para entregar la información de los requerimientos que se hagan de acuerdo a la consulta solicitada. Para conocer el detalle ver anexo 2. Documentación técnica del código Backend.
- **Funcionamiento del FrontEnd:** El Front está hecho en React basados en componentes de las distintas funcionalidades para los mapas y filtros de los determinantes. Éste hace los llamados al Backend para obtener los datos y aplicarlos en la presentación de acuerdo a la funcionalidad que se esté aplicando. Para conocer el detalle ver anexo 1. Documentación técnica del código FrontEnd.

6.2 Compatibilidad

La compatibilidad con diferentes navegadores y sistemas operativos es esencial para garantizar que el aplicativo web pueda ser utilizado por una amplia gama de usuarios. Con este fin, se asegura que la aplicación sea completamente funcional en las versiones más recientes de los navegadores más utilizados:

- **Navegadores compatibles:**
 - **Google Chrome:** Versión 100 o superior.
 - **Mozilla Firefox:** Versión 100 o superior.
 - **Microsoft Edge:** Versión 100 o superior.
 - **Safari:** Versión 14 o superior.

Además de asegurar la compatibilidad entre navegadores, se implementa una **estrategia de pruebas entre navegadores** para garantizar que la aplicación funcione correctamente en todas estas plataformas. Esta estrategia incluye:

- **Pruebas manuales** en ambiente de pruebas controlado, lo que permite identificar problemas específicos que puedan ser ajustados o mejorados antes de pasar al ambiente de producción.

Por otro lado, el diseño también soporta múltiples resoluciones, asegurando que la interfaz se adapte y sea completamente funcional en dispositivos con pantallas desde 320px en adelante, lo que garantiza su accesibilidad en una amplia gama de dispositivos, desde teléfonos móviles hasta pantallas grandes de escritorio.

6.3 Adaptabilidad

El diseño del aplicativo web ha sido concebido para ofrecer una experiencia de usuario óptima tanto en dispositivos móviles como en escritorios, adaptándose a diferentes tamaños de pantalla. Para ello, se ha implementado un enfoque **responsive** que asegura que el contenido se vea correctamente en cualquier dispositivo:

- **Diseño responsive utilizando Tailwind CSS:** Gracias a la implementación de **Tailwind CSS**, el diseño se adapta dinámicamente utilizando técnicas como **grid**, **flex** y **breakpoints**. Esta solución asegura que los componentes de la interfaz se reorganicen y ajusten según las dimensiones de la pantalla, brindando una experiencia coherente y agradable sin importar el dispositivo.
- **Layout adaptable:** El diseño del layout es adaptable a los diferentes tipos de dispositivos, lo que mejora significativamente la experiencia de navegación:
 - En pantallas de escritorio, los datos se muestran en una disposición de columnas, con los elementos uno al lado del otro, aprovechando el espacio disponible.
 - En **tablets** y **móviles**, la vista se reorganiza en un formato tipo carrusel o scroll vertical, optimizando el espacio de pantalla disponible y facilitando la navegación táctil.
- **Optimización para dispositivos móviles:** Además, se ha integrado una librería llamada **react-device-detect**, que permite identificar el tipo de dispositivo móvil utilizado por el usuario. Esto permite ofrecer comportamientos personalizados de la aplicación según el dispositivo, asegurando una experiencia adecuada para cada tipo de usuario.

Mapa Principal

Búsqueda por coordenadas

Seleccionar municipio manualmente:

El mapa muestra la distribución de la precipitación por hora en Colombia. La leyenda indica los siguientes rangos de precipitación por hora:

- 100% (Rojo)
- 80% (Naranja)
- 60% (Amarillo)
- 40% (Verde claro)
- 20% (Verde oscuro)
- 0% (Azul)

El mapa muestra que las áreas de alta precipitación (rojo/naranja) están principalmente en el norte y centro del país, mientras que las áreas de baja precipitación (verde/azul) están principalmente en el sur y oeste.

Carrera 35A No. 15B - 35 Of. 501 Edificio Prisma – Medellín
Av. Calle 26 # 68c – 61 Oficina 531-1 Edificio Torre Central -Bogotá
<https://ada.co/>

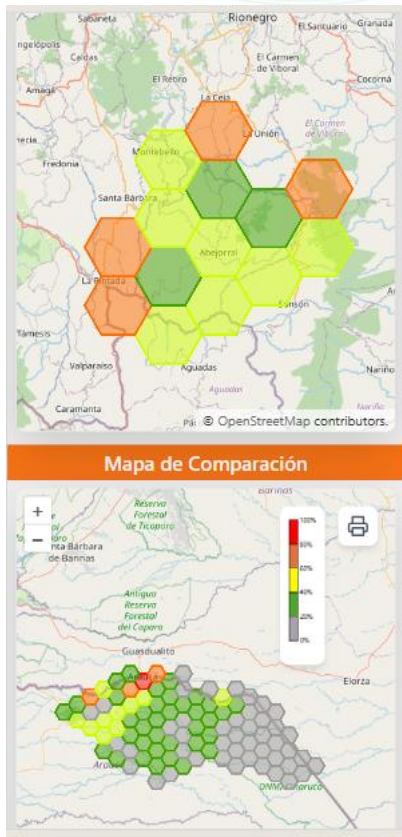
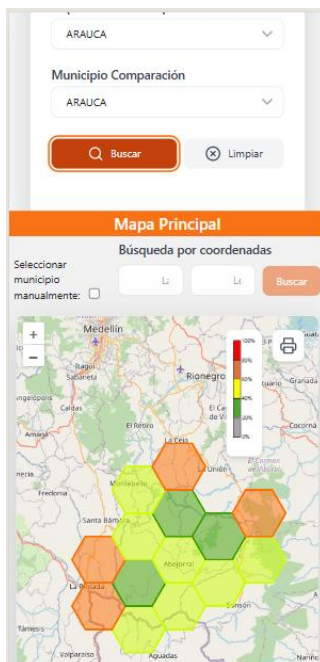


Ilustración 7. Pruebas de responsividad. Fuente: Imagen elaborada por ADA.



ARAUCA

Municipio Comparación

ARAUCA

Buscar

Limpiar

Mapa Principal

Búsqueda por coordenadas

Seleccionar municipio manualmente: ☐

Ilustración 8. Pruebas de responsividad. Fuente: Imagen elaborada por ADA.

6.4 Pruebas funcionales

Para garantizar el correcto funcionamiento y la calidad continua de la aplicación, se implementa un sistema de pruebas funcionales en ambientes controlados. Este proceso estará bajo la responsabilidad de la fábrica de software ADA SAS, asegurando que cualquier cambio o mejora se realice de manera controlada y segura.

Las pruebas que se llevaron a cabo se detallan a continuación:

6.4.1 Escenario 1: Consulta de determinantes e indicadores sin comparación.

1. Se seleccionaron los filtros (zona, departamento, municipio), y los determinantes que se van a consultar (social, ambiental, sectorial, infraestructura), el sistema permite seleccionar todos los determinantes y también de manera individual.
2. Se seleccionó el radio botón “NO” en la sección comparar departamento/municipio.
3. Se dio clic en el botón “Buscar”. El sistema carga la información solicitada en el mapa que se visualiza a la derecha de la pantalla.

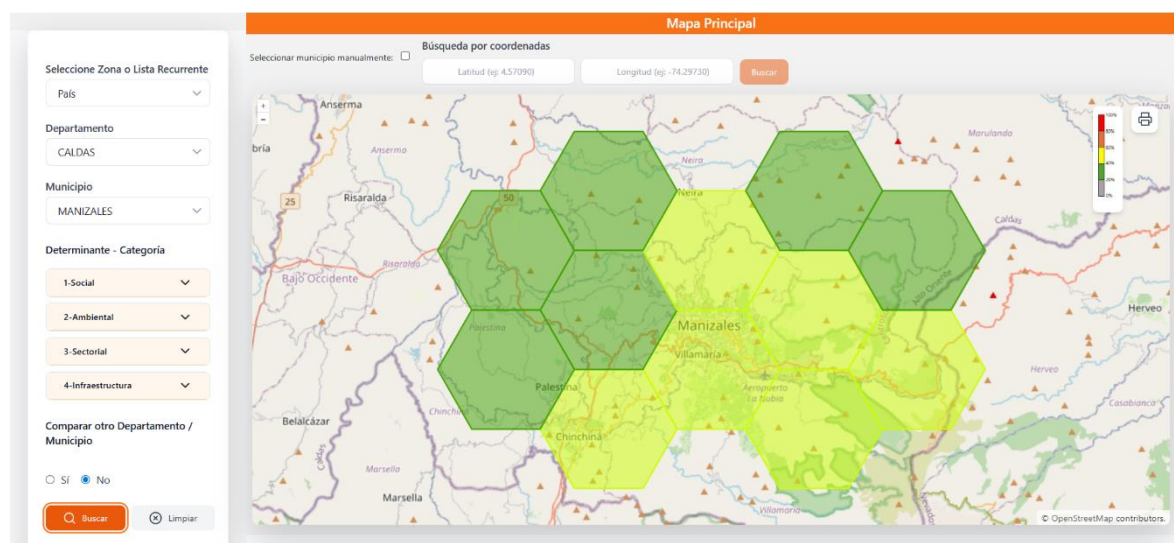
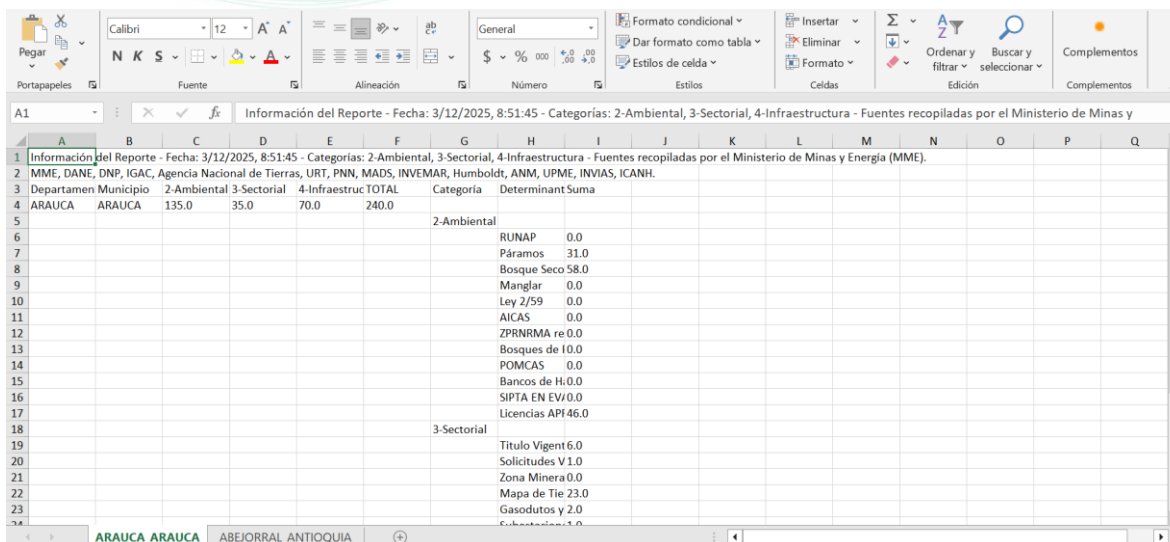


Ilustración 9. Pruebas de funcionamiento determinantes. Fuente: Imagen elaborada por ADA.

4. Una vez cargada la información exporta la información en formato .xlsx de los determinantes a través del botón “Exportar a Excel”.



Información del Reporte - Fecha: 3/12/2025, 8:51:45 - Categorías: 2-Ambiental, 3-Sectorial, 4-Infraestructura - Fuentes recopiladas por el Ministerio de Minas y Energía (MME).																	
MME, DANE, DNP, IGAC, Agencia Nacional de Tierras, URT, PNN, MADS, INVEMAR, Humboldt, ANM, UPME, INVIAS, ICANH.																	
Departamen Municipio	2-Ambiental	3-Sectorial	4-Infraestructura	TOTAL	Categoría	Determinant	Suma										
ARAUCA	ARAUCA	135.0	35.0	70.0	240.0												
						2-Ambiental											
						RUNAP	0.0										
						Páramos	31.0										
						Bosque Seco	58.0										
						Manglar	0.0										
						Ley 2/59	0.0										
						AICAS	0.0										
						ZPRNRMA re	0.0										
						Bosques de I	0.0										
						POMCAS	0.0										
						Bancos de H	0.0										
						SIPTA EN EV	0.0										
						Licencias API	46.0										
						3-Sectorial											
						Titulo Vigent	6.0										
						Solicitudes V	1.0										
						Zona Minera	0.0										
						Mapa de Tie	23.0										
						Gasodutos y	2.0										
						Subestaciones	1.0										

Ilustración 10. Pruebas de funcionamiento exportar determinantes. Fuente: Imagen elaborada por ADA.

- En la sección indicadores socioeconómicos se puede seleccionar las variables que se necesiten del listado a través de los botones de selección (puede seleccionar todas a la vez). En este ejemplo se seleccionaron actividades terciarias, población rural y población urbana.
- Una vez seleccionados los indicadores socioeconómicos, se presiona el botón “buscar”.
- Se muestra la tabla “Que desea graficar”, se seleccionan tres graficas simultáneas.
- El sistema muestra las gráficas correctamente.

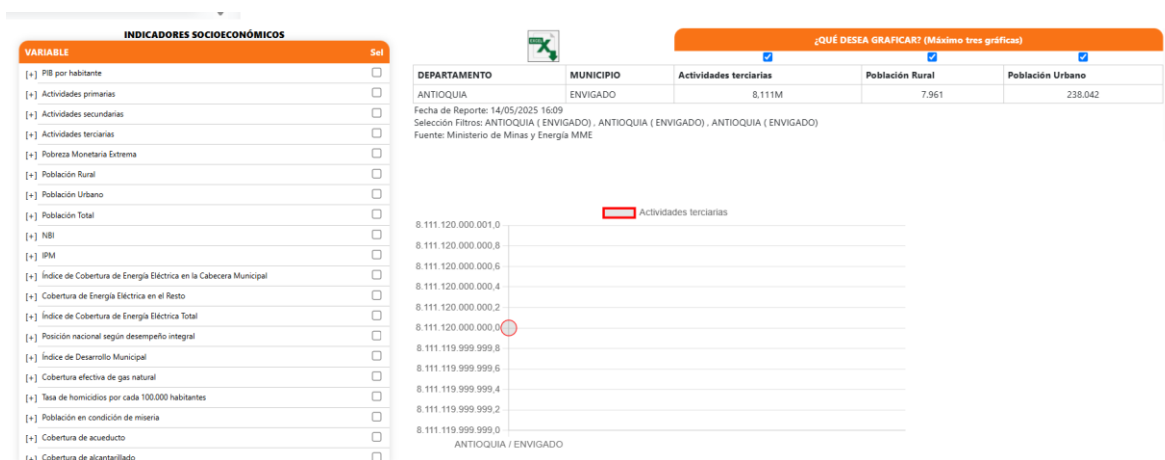
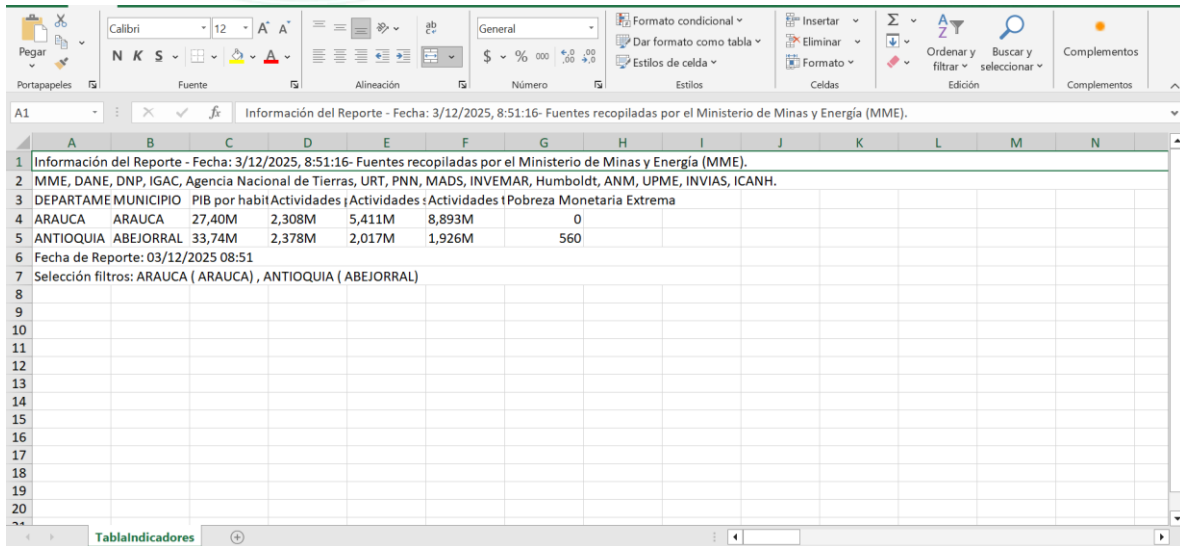


Ilustración 11. Pruebas de funcionamiento indicadores. Fuente: Imagen elaborada por ADA.

- Se exportó la información en formato .xlsx a través del botón con el logo de Excel.



DEPARTAMENTO MUNICIPIO	PIB por habitante	Actividades	Actividades	Pobreza Monetaria	Extrema
ARAUCA ARAUCA	27,40M	2,308M	5,411M	8,893M	0
ANTIOQUIA ABEJORRAL	33,74M	2,378M	2,017M	1,926M	560

Ilustración 12. Pruebas de funcionamiento exportar indicadores. Fuente: Imagen elaborada por ADA.

10. Se realizó prueba del botón “imprimir mapa”, el sistema muestra correctamente la información solicitada incluyendo los metadatos.

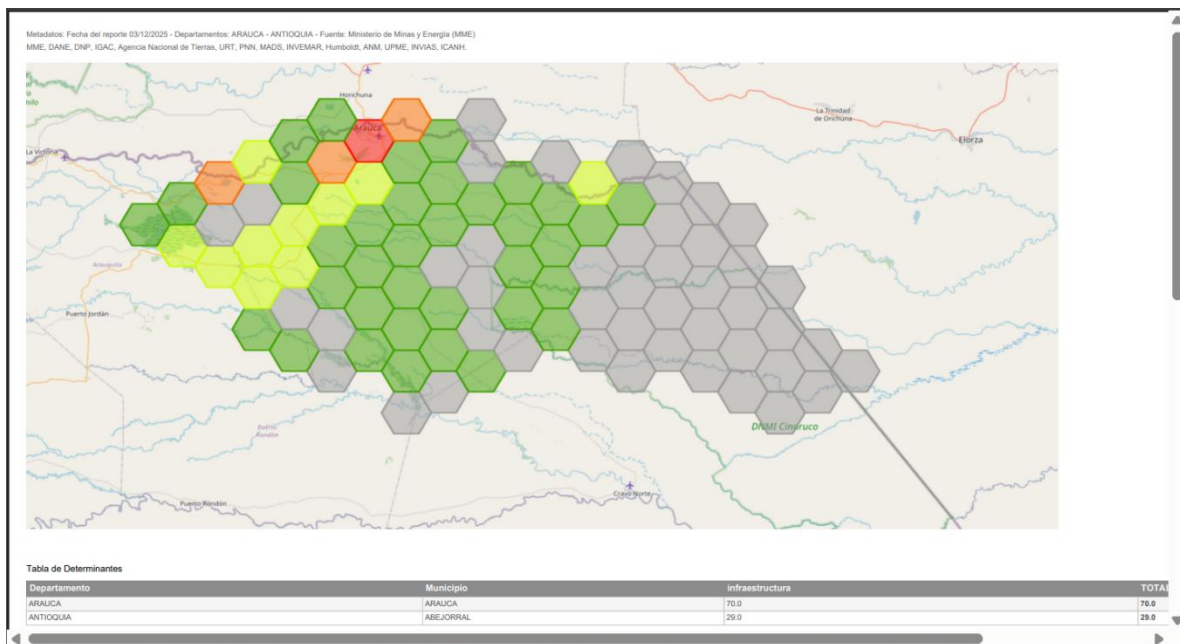


Ilustración 13. Pruebas de funcionamiento exportar mapa en PDF. Fuente: Imagen elaborada por ADA.

11. Se presionó el botón “limpiar” en la sección “indicadores”, se borraron correctamente la tabla y las gráficas.

INDICADORES SOCIOECONÓMICOS	
VARIABLE	Sei
[+] PIB por habitante	☐
[+] Actividades primarias	☐
[+] Actividades secundarias	☐
[+] Actividades terciarias	☐
[+] Pobreza Monetaria Extrema	☐
[+] Población Rural	☐
[+] Población Urbana	☐
[+] Población Total	☐
[+] HDI	☐
[+] IPM	☐
[+] Índice de Cobertura de Energía Eléctrica en la Cabecera Municipal	☐
[+] Cobertura de Energía Eléctrica en el Resto	☐
[+] Índice de Cobertura de Energía Eléctrica Total	☐
[+] Posición nacional según desempeño integral	☐
[+] Índice de Desarrollo Municipal	☐
[+] Cobertura efectiva de gas natural	☐
[+] Tasa de homicidios por cada 100.000 habitantes	☐
[+] Población en condición de miseria	☐
[+] Cobertura de acueducto	☐
[+] Cobertura de alcantarillado	☐

Indicadores Compuestos	
VARIABLE	Sei
Año de reporte de la base de datos	☐
Porcentaje en el municipio	☐
Porcentaje en el departamento	☐
Posición en el departamento	☐

Ilustración 14. Pruebas de funcionamiento indicadores botón limpiar. Fuente: Imagen elaborada por ADA.

12. Se presionó el botón “limpiar” debajo del desplegable en la sección “determinantes”, se borró correctamente la información del mapa.

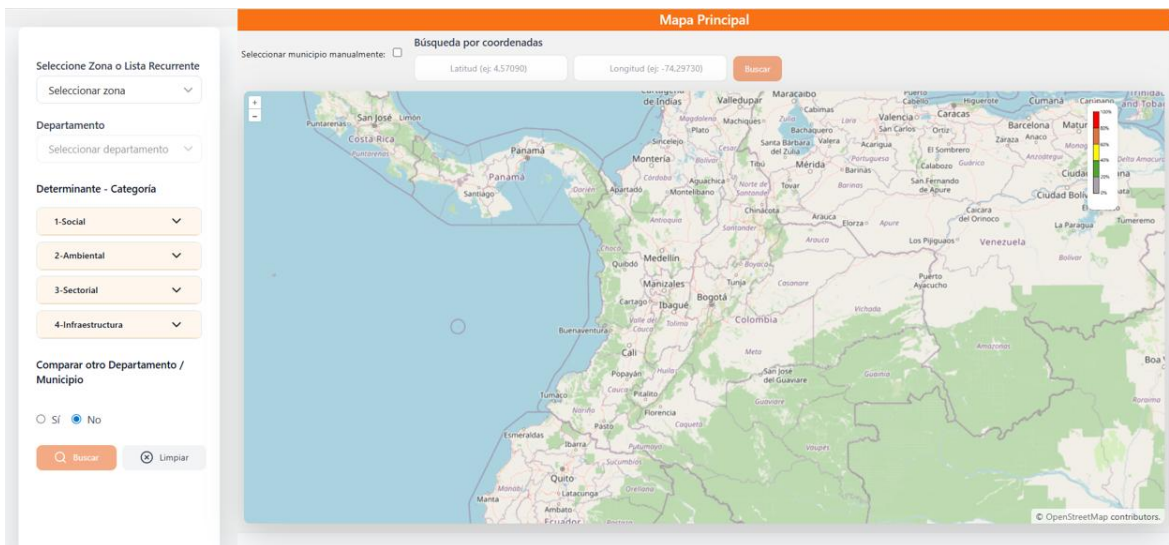


Ilustración 15. Pruebas de funcionamiento determinantes botón limpiar. Fuente: Imagen elaborada por ADA.

6.4.2 Escenario 2: Consulta de determinantes e indicadores con comparación.

1. Se seleccionaron los filtros (zona, departamento, municipio), y los determinantes que se van a consultar (social, ambiental, sectorial, infraestructura), el sistema permite seleccionar todos los determinantes y también de manera individual.

2. Se selecciona el radio botón “SI” en la sección comparar departamento/municipio.

3. El sistema habilita dos campos adicionales “departamento comparación” y “municipio comparación”, en los cuales para el ejercicio se seleccionó Cundinamarca, Chía. Adicionalmente el sistema habilitó de manera correcta un mapa adicional a la derecha de la pantalla donde se muestra la información del departamento/municipio a comparar.
4. Se da clic en el botón “Buscar”. El sistema cargará la información solicitada en el mapa que se visualiza a la derecha de la pantalla.

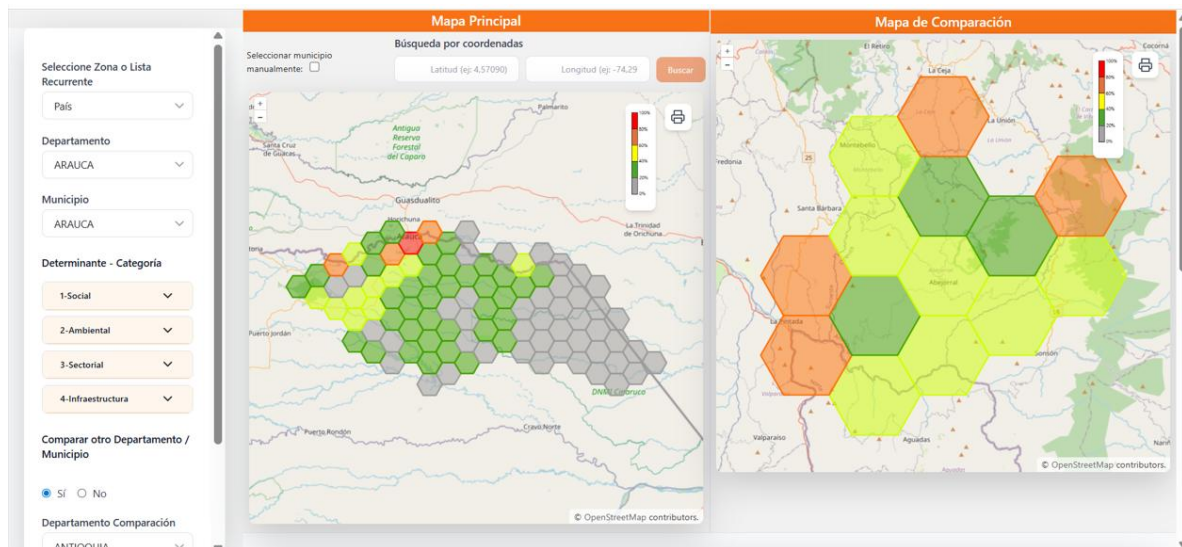


Ilustración 16. Pruebas de funcionamiento determinantes. Fuente: Imagen elaborada por ADA.

5. En la sección indicadores socioeconómicos se puede seleccionar las variables que se necesiten del listado a través de los botones de selección (puede seleccionar todas a la vez). En este ejemplo se seleccionaron actividades terciarias, población rural y población urbana.
6. Una vez seleccionados los indicadores socioeconómicos, se presiona el botón “buscar”.
7. Se muestra la tabla “Que desea graficar”, se seleccionaron tres graficas simultáneas. Para este ejemplo el sistema muestra la información en la tabla de los dos municipios seleccionados previamente.
8. El sistema muestra las gráficas correctamente.

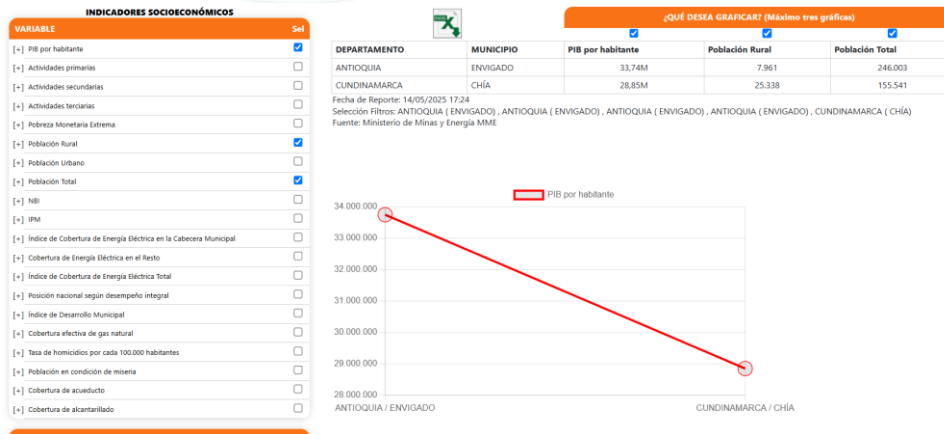
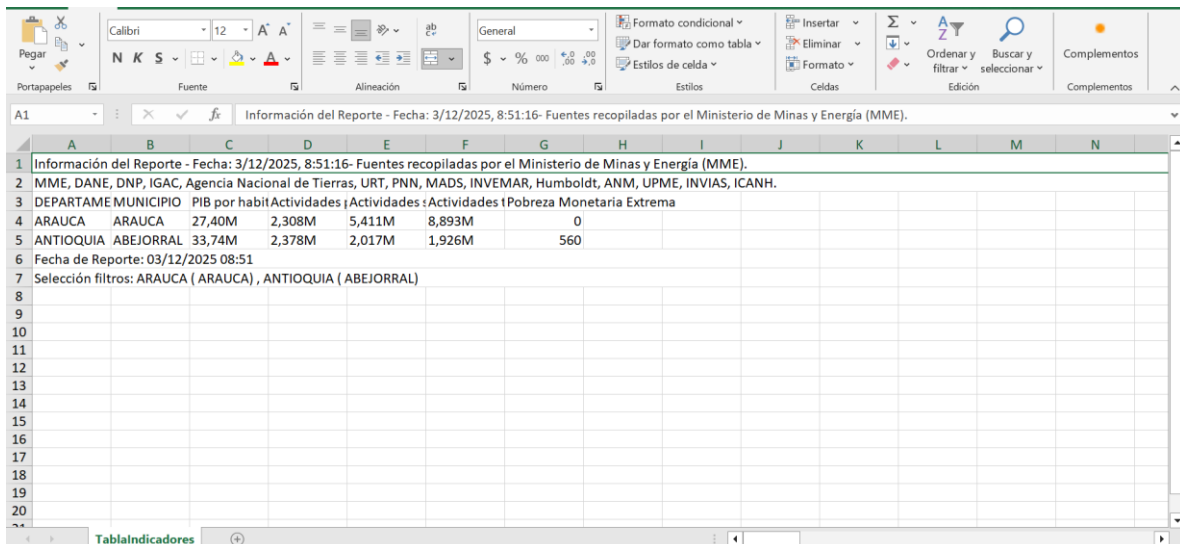


Ilustración 17. Pruebas de funcionamiento indicadores. Fuente: Imagen elaborada por ADA.

9. Se exportó la información en formato .xlsx a través del botón con el logo de Excel.



Información del Reporte - Fecha: 3/12/2025, 8:51:16- Fuentes recopiladas por el Ministerio de Minas y Energía (MME).						
Información del Reporte - Fecha: 3/12/2025, 8:51:16- Fuentes recopiladas por el Ministerio de Minas y Energía (MME).						
MME, DANE, DNP, IGAC, Agencia Nacional de Tierras, URT, PNN, MADS, INVEMAR, Humboldt, ANM, UPME, INVIAS, ICANH.						
DEPARTAME	MUNICIPIO	PIB por habit	Actividades	Actividades t	Pobreza Monetaria	Extrema
ARAUCA	ARAUCA	27,40M	2,308M	5,411M	8,893M	0
ANTIOQUIA	ABEJORRAL	33,74M	2,378M	2,017M	1,926M	560
Fecha de Reporte: 03/12/2025 08:51						
Selección filtros: ARAUCA (ARAUCA), ANTIOQUIA (ABEJORRAL)						

Ilustración 18. Pruebas de funcionamiento exportar indicadores. Fuente: Imagen elaborada por ADA.

10. Se presionó el botón “limpiar” en la sección “indicadores”, se borraron correctamente la tabla y las gráficas.

INDICADORES SOCIOECONÓMICOS

VARIABLE	Sei
[+] PIB por habitante	☐
[+] Actividades primarias	☐
[+] Actividades secundarias	☐
[+] Actividades terciarias	☐
[+] Pobreza Monetaria Extrema	☐
[+] Población Rural	☐
[+] Población Urbana	☐
[+] Población Total	☐
[+] HDI	☐
[+] IPM	☐
[+] Índice de Cobertura de Energía Eléctrica en la Cabecera Municipal	☐
[+] Cobertura de Energía Eléctrica en el Resto	☐
[+] Índice de Cobertura de Energía Eléctrica Total	☐
[+] Posición nacional según desempeño integral	☐
[+] Índice de Desarrollo Municipal	☐
[+] Cobertura efectiva de gas natural	☐
[+] Tasa de homicidios por cada 100.000 habitantes	☐
[+] Población en condición de miseria	☐
[+] Cobertura de acueducto	☐
[+] Cobertura de alcantarillado	☐

Indicadores Compuestos

Indicadores Compuestos	Sei
Año de reporte de la base de datos	☐
Porcentaje en el municipio	☐
Porcentaje en el departamento	☐
Posición en el departamento	☐

¿QUÉ DESEA GRAFICAR? (Máximo tres gráficas)

Ilustración 19. Pruebas de funcionamiento indicadores botón limpiar. Fuente: Imagen elaborada por ADA.

6.4.3 Escenario 3: Consulta manual.

1. Se selecciona el check de “Seleccionar municipio manualmente”, el cual se encuentra ubicado en la parte superior de la pantalla. El sistema carga la información en polígonos de todo el país (Colombia).
2. Se da clic en cualquier zona del mapa de Colombia y allí se visualiza tooltip con la información del polígono del mapa seleccionado, con datos como las coordenadas X y Y (latitud y longitud), id del municipio, id departamento, etc.

Seleccione Zona o Lista Recurrente

Seleccionar zona

Departamento

Seleccionar departamento

Determinante - Categoría

1-Social

2-Ambiental

3-Sectorial

4-Infraestructura

Comparar otro Departamento / Municipio

☐ Sí

☒ No

Buscar

Limpiar

Seleccionar municipio manualmente: ☒

Búsqueda por coordenadas

Latitud (eg. 4.57090)

Longitud (eg. -74.29730)

Buscar

Mapa de Colombia

BOGOTÁ D.C.

Latitud: 4.673881790000053

Longitud: -74.15495270499997

idMunicipio: 11001

Municipio: BOGOTÁ D.C.

idDepartamento: 11

Departamento: BOGOTÁ D.C.

Consultar Municipio

Ilustración 20. Pruebas de funcionamiento Selección Manual. Fuente: Imagen elaborada por ADA.

3. Se valida que al dar clic en el botón “consultar municipio” de la ventana emergente, se carga automáticamente el panel de la izquierda con la información del polígono seleccionado.

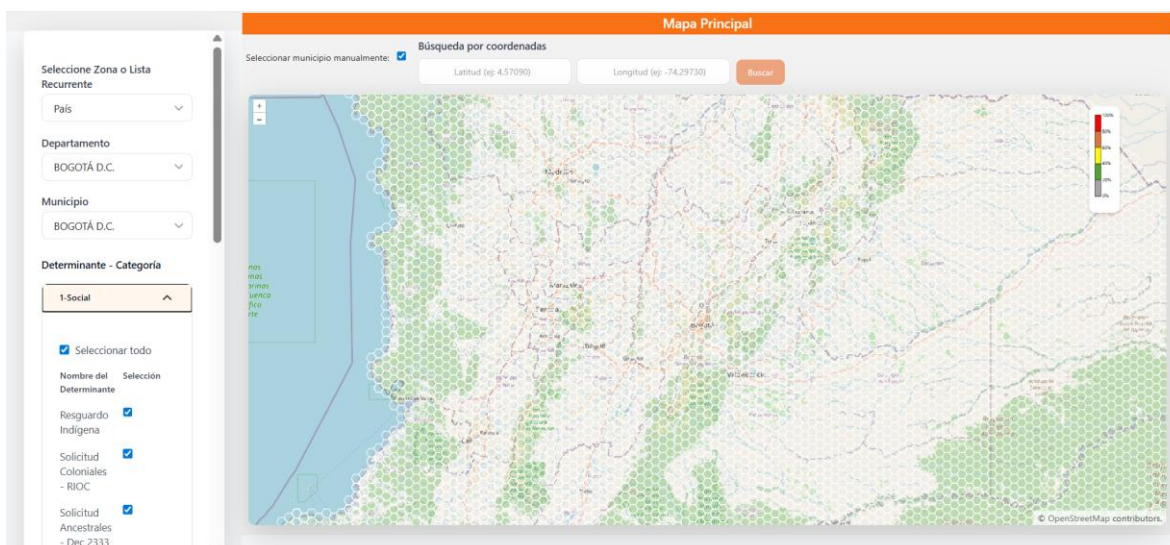


Ilustración 21. Pruebas de funcionamiento Selección Manual. Fuente: Imagen elaborada por ADA.

6.4.4 Escenario 4: Consulta por coordenadas.

El aplicativo ofrece la posibilidad al usuario de realizar consultas por coordenadas latitud y longitud, para ello se dispuso en el diseño de la interfaz con dos campos ubicados encima del mapa junto con el botón “Buscar”. De este modo el sistema mostrará automáticamente en el mapa la ubicación de la coordenada geográfica.

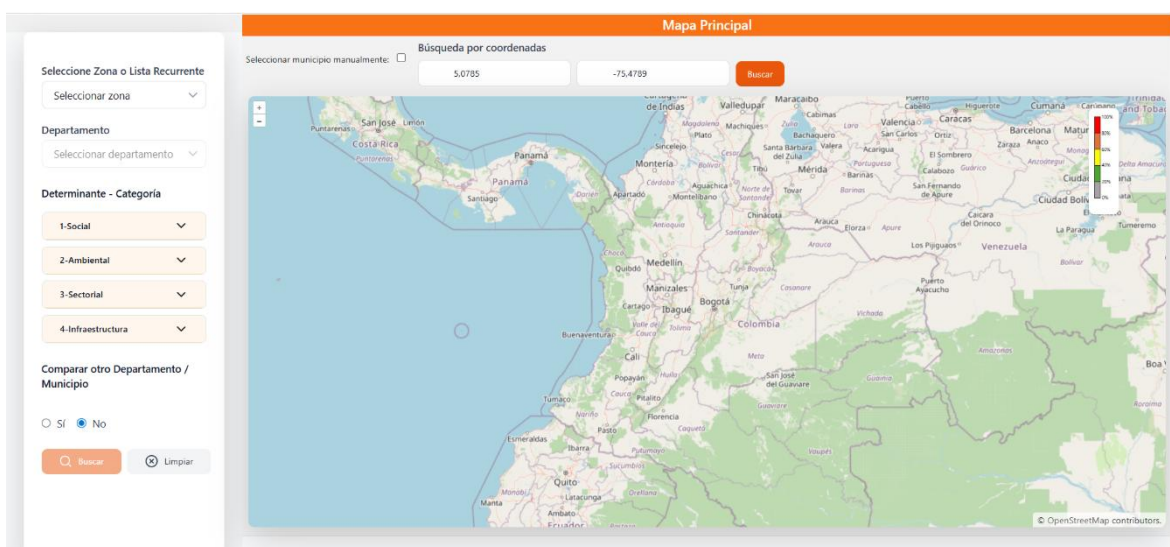


Ilustración 22. Funcionalidad X, Y. Fuente: ADA

7. Validación de Requisitos

La validación de requisitos es un proceso crucial para garantizar que el sistema cumpla con los estándares de funcionalidad, rendimiento y seguridad esperados. Este proceso involucra múltiples pruebas y verificaciones que abarcan diversos aspectos del aplicativo web, desde su accesibilidad hasta la correcta exportación de datos, pasando por la visualización de la información y la generación de reportes. A continuación, se describen en detalle las pruebas y validaciones específicas para cada uno de estos aspectos.

7.1 Acceso al Sistema

La validación del acceso al sistema se lleva a cabo mediante pruebas en el ambiente de desarrollo suministrado por el Ministerio de Minas y Energía, con el objetivo de asegurar que el aplicativo sea accesible, confiable y seguro en su entorno de producción.

7.2 Visualización de Información

La validación de la visualización de datos se centra en asegurar que los usuarios puedan ver la información de manera clara, precisa y organizada, tanto en tablas como en gráficos, sin distorsiones o errores. Se realizan pruebas funcionales para verificar que cada tipo de dato se visualice correctamente, adaptándose a las necesidades específicas del usuario.

- **Casos de prueba por tipo de variable:**
 - **Variables numéricas:** Se validan los datos de tipo numérico, como los indicadores, para asegurarse de que se muestren correctamente en los gráficos y tablas. Se comprueba que los valores se calculen y presenten con precisión, permitiendo a los usuarios interpretar la información sin dificultades.
 - **Variables categóricas:** Se revisan las variables categóricas, como el tipo de determinante, para asegurar que se visualicen adecuadamente. En este caso, las opciones o categorías deben aparecer correctamente segmentadas y fácilmente comprensibles.
 - **Variables geoespaciales:** Permite realizar consultas basadas en punto coordenada geográfica, para identificar la data de un territorio específico (por ejemplo, dando clic a uno de los polígonos pertenecientes a Leticia en el departamento del Amazonas). Este tipo de validación asegura que la información espacial se refleje correctamente, facilitando la interpretación de los datos a los usuarios.

7.3 Reportes Generados

Los reportes generados por el sistema deben ser precisos, completos y estar alineados con los filtros y parámetros seleccionados por el usuario. Se realiza una validación de la generación de reportes para garantizar que los resultados sean fieles a las expectativas y necesidades del usuario final.

- **Revisión de contenido descargado:** Se comprueba que los reportes descargados, ya sea en formato PDF o Excel, coincidan con los filtros aplicados por el usuario. Esta revisión garantiza que los datos presentados en los reportes estén actualizados y sean relevantes para el contexto de la consulta.
- **Verificación de la integridad de los datos exportados:** Se valida que los datos exportados mantengan su integridad durante el proceso de descarga. Esto incluye comprobar aspectos como los encabezados de las tablas, los formatos de fecha y la correcta codificación de caracteres, para evitar problemas de interpretación de los datos en otros sistemas o herramientas.
 - **Ejemplo de prueba automatizada:**
 - Se valida que el nombre del archivo exportado contenga la información correspondiente al reporte:
`expect(exportedFile.name).toContain("reporte_meta");`
 - Además, se verifica que los datos exportados contengan al menos una fila, asegurando que el reporte no esté vacío:
`expect(exportedData.rows.length).toBeGreaterThan(0);`

7.4 Exportación de Datos

La exportación de datos es un proceso fundamental que permite a los usuarios obtener la información en formatos más accesibles, como Excel y PDF, para su análisis y posterior uso. La validación de este proceso incluye tanto pruebas funcionales como pruebas de rendimiento, para garantizar que la exportación se realice correctamente, incluso cuando se manejen grandes volúmenes de datos.

- **Exportación sin errores:** Se valida que el sistema sea capaz de exportar los datos correctamente en los formatos establecidos (Excel y PDF), sin errores ni pérdida de información. Esta validación asegura que la funcionalidad de exportación sea confiable y eficiente.
- **Pruebas de estrés para grandes volúmenes de datos:** Se realizan pruebas de rendimiento utilizando herramientas como JMeter para simular la exportación de grandes volúmenes de datos. Este tipo de prueba permite identificar posibles cuellos de botella o limitaciones en el sistema que puedan afectar su capacidad para manejar grandes cargas de trabajo. Ver anexo 4. Prueba de estrés aplicativo web.

- **Compatibilidad con herramientas estándar:** Se comprueba que los archivos exportados sean compatibles con las herramientas más utilizadas en la industria, como Excel, LibreOffice y Google Sheets. Esto garantiza que los usuarios puedan abrir y trabajar con los datos exportados sin problemas, independientemente de la herramienta que utilicen.

8. Arquitectura y Reglas del Sistema

La arquitectura del aplicativo web está diseñada de manera que se aprovechen principios modernos de desarrollo de software, implementando una estructura modular, escalable y desacoplada. Esta configuración se basa en una arquitectura de tres capas: la capa de presentación, que es responsable de la interfaz de usuario (frontend), la capa de lógica de negocio, que gestiona el procesamiento de datos a través de la API del backend, y la capa de persistencia de datos, que se encarga del almacenamiento seguro y eficiente de la información mediante una base de datos geoespacial. Esta separación clara de responsabilidades permite una gestión óptima de cada componente del sistema, lo que a su vez facilita su mantenimiento y escalabilidad.

El sistema se despliega sobre una infraestructura web avanzada que habilita el uso de tecnologías de vanguardia. Utiliza contenedores Docker, que aseguran la portabilidad y la consistencia del entorno de ejecución, además de bases de datos geoespaciales, específicamente PostgreSQL con la extensión PostGIS, que optimiza la gestión de datos espaciales. Esta infraestructura permite que el flujo de consulta y transformación de datos, tanto espaciales como estadísticos, sea lo más eficiente posible, asegurando que el sistema responda de manera rápida y precisa ante las solicitudes de los usuarios.

8.1 Infraestructura de Bases de Datos

El sistema emplea PostgreSQL 14 como su sistema gestor de bases de datos, potenciado con la extensión PostGIS 3.4, lo que permite manejar datos espaciales con una eficiencia notable. Las tablas principales en la base de datos están estructuradas de la siguiente manera:

- **Indicadores:** Esta tabla tiene como clave primaria una combinación compuesta de los campos territorio_id, año y variable_id, lo que permite almacenar de forma única cada indicador por territorio.
- **Territorios:** Esta tabla incluye una columna Geom que almacena la geometría de los territorios, además de información relevante como el nombre y el tipo del territorio (municipio o departamento).

- **Variables:** Esta tabla contiene la información meta informacional de cada indicador o determinante, permitiendo una descripción detallada de los datos.

Para optimizar las consultas geográficas, se implementan índices espaciales en las columnas geométricas. Además, el sistema asegura conexiones seguras a la base de datos mediante roles autenticados y el uso de certificados SSL, lo que garantiza la integridad y la seguridad de la información.

8.2 Transformación de Datos

El proceso de carga de datos está diseñado para ser eficiente y flexible, gestionando archivos provenientes de diversas fuentes, como Excel o Shapefiles. Este proceso sigue una arquitectura ETL (Extracción, Transformación y Carga), en la que se realizan las siguientes fases:

- **Extracción:** Se utilizan herramientas como pandas (Python) y GDAL para la lectura de los archivos, lo que permite una integración fluida de los datos desde diferentes formatos y estructuras.
- **Transformación:** Durante esta etapa, se normalizan los formatos, los nombres de las columnas, las codificaciones y las unidades de medida, para garantizar la consistencia y compatibilidad de los datos a lo largo del sistema.
- **Carga:** Los datos se insertan en la base de datos a través de SQLAlchemy o mediante el comando COPY de PostgreSQL, lo que permite una inserción eficiente de grandes volúmenes de datos, este proceso se realizó mediante instrucciones SQL topo batch por consola.

El proceso de validación es una parte crucial para garantizar la calidad de los datos. Durante este proceso, se eliminan los valores nulos o incongruentes, y se detectan y eliminan registros duplicados mediante claves compuestas, asegurando que solo los datos más precisos y fiables sean cargados en el sistema.

8.3 Representación de Información

El sistema permite la visualización de las variables a través de mapas vectoriales interactivos. Para ello, se utiliza la biblioteca Leaflet.js junto con capas GeoJSON, lo que proporciona una interfaz de usuario intuitiva y altamente interactiva. Además, se implementa la técnica de "lazy loading" para cargar solo las entidades visibles en el mapa, lo que optimiza la carga y mejora el rendimiento general del sistema.

La interacción con el backend se realiza mediante endpoints RESTful, como el siguiente ejemplo:

```
bash
```

```
/divterritorialmuni_determinantes?mpio_cdpmp=in.(05001,05002,05004)
```

Ilustración 23. Representación de Información. Fuente: Imagen elaborada por ADA

Este enfoque permite que el sistema sea altamente flexible y eficiente al momento de consultar y visualizar la información. En cuanto a la representación visual de los datos, se utiliza una codificación de valores basada en escalas de colores (tipo choropleth) y leyendas dinámicas, lo que facilita la comprensión de los datos y permite a los usuarios interpretar rápidamente los valores representados en los mapas.

8.4 Configuración de Reglas

Las reglas de negocio se configuran de manera sencilla y flexible mediante archivos JSON externos, lo que permite modificar las reglas sin necesidad de recompilar el código del sistema. Este enfoque facilita la actualización y personalización de las reglas sin interrumpir el funcionamiento del sistema.

Un ejemplo de una regla configurada podría ser el siguiente:

```
json

{
  "indicador": "cobertura_energia",
  "umbral_critico": 40,
  "accion": "destacar en rojo"
}
```

Ilustración 24. Configuración de Reglas. Fuente: Imagen elaborada por ADA

Además de estas reglas, el sistema es capaz de manejar reglas geográficas, lo que permite generar alertas visuales en función de valores específicos. Por ejemplo, si el valor de un municipio está por debajo de un umbral crítico nacional, el sistema generará una alerta visual para destacar esta área en los mapas. Estas reglas son procesadas en el backend antes de la generación de reportes o mapas, asegurando que las decisiones visuales y estadísticas estén siempre basadas en los criterios definidos por el usuario.

Este enfoque flexible y eficiente en la configuración de reglas asegura que el sistema pueda adaptarse a distintos contextos y requerimientos, proporcionando una herramienta poderosa y fácil de personalizar para los usuarios finales.

Nota: Dirigirse a la sección 12 del presente documento para conocer el detalle del diccionario de datos.

9. Procesamiento y Visualización

La administración de los datos dentro de la plataforma sigue principios clave como la integridad, eficacia y escalabilidad, lo que garantiza un manejo robusto y confiable de la información a lo largo de todo su ciclo de vida. En el corazón del sistema, los datos se almacenan en una base de datos relacional, diseñada para soportar un alto rendimiento y una gestión eficiente de grandes volúmenes de información. Para asegurar una estructura eficiente y escalable, la base de datos emplea PostGIS, una extensión de PostgreSQL que permite gestionar datos espaciales, integrando tanto información tabular como geométrica.

Los procesos de ingestión, validación y transformación de los datos se automatizan para maximizar la eficiencia operativa. Estos procesos son esenciales para asegurar que los datos que ingresan al sistema sean consistentes, precisos y adecuados para su posterior utilización. A través de un sistema de procesos automáticos, la plataforma valida, transforma y adapta los datos según las necesidades específicas del usuario y el contexto del sistema.

El almacenamiento en la plataforma no se limita únicamente a los datos tabulares convencionales, sino que también incluye la gestión de geometrías. Esto permite realizar operaciones espaciales avanzadas, como análisis geoespaciales y visualizaciones interactivas de los datos. Gracias a esta capacidad, los usuarios pueden realizar filtraciones complejas que abarcan múltiples variables simultáneamente, lo que les permite obtener insights profundos y específicos de los datos almacenados.

9.1 Descripción del almacenamiento

9.1.1 Base de datos de almacenamiento primario

El sistema utiliza PostgreSQL con la extensión PostGIS para almacenar tanto datos tabulares como geoespaciales. Esta combinación de tecnologías asegura un almacenamiento eficiente y flexible, adaptado a las necesidades específicas de la plataforma.

9.2 Estructura de los datos

9.2.1 Tablas normalizadas

Los datos se organizan en tablas que siguen principios de normalización, lo que facilita su manejo y optimiza las consultas. Las relaciones entre las tablas se definen mediante claves

foráneas (FOREIGN KEY), y se implementan restricciones de integridad, como claves de verificación (CHECK) y campos obligatorios (NOT NULL), para garantizar la calidad y consistencia de la información.

9.2.2 Separación de datos operativos y de referencia

Los datos de referencia, tales como catálogos y listas de valores predefinidos, se almacenan por separado de los datos operativos para mejorar el rendimiento y la flexibilidad del sistema.

9.2.3 Almacenamiento de archivos fuente

Los archivos originales, como documentos Excel y shapefiles, se almacenan en un sistema de almacenamiento de objetos como S3 o MinIO. Esta solución no solo permite una gestión más eficiente de los archivos, sino que también facilita su trazabilidad, garantizando que se pueda hacer un seguimiento preciso de cada archivo desde su origen hasta su utilización.

9.3 Mecanismo de versionado de datasets

La plataforma implementa un mecanismo robusto de versionado de datasets, lo que permite un seguimiento preciso de los cambios y asegura la integridad de los datos a lo largo del tiempo. Cada conjunto de datos cargado en el sistema está asociado a una serie de metadatos que incluyen:

9.3.1 Fecha de carga

Permite conocer el momento exacto en que los datos fueron ingresados en el sistema.

9.3.2 Origen de los datos

Se registra el origen de cada conjunto de datos, asegurando la trazabilidad y la fiabilidad de la información.

9.3.3 Usuario responsable

Cada carga de datos se asocia con el usuario que ejecutó el proceso, lo que asegura la trazabilidad y responsabilidad en la gestión de los datos.

9.3.4 Hash de los archivos

Se calcula un valor hash para cada archivo cargado, lo que permite verificar su integridad en el futuro. Si se detecta cualquier alteración, el sistema puede restaurar el archivo original para mantener la coherencia de los datos.

9.4 Detalle control de Versionamiento herramientas usadas para el aplicativo web

Nombre del software	Versión	Playwright	Cypress	Proveedor	Fecha de instalación	Entorno
Cliente: Docker Engine	28.1.1	N/A	N/A	Docker Inc.	18-abr-25	Pruebas MME
Servidor: Docker Engine	28.0.4	N/A	N/A	Docker Inc.	25-mar-25	Pruebas MME
Python	3.11 slim	1.43.0	N/A	Python software Foundation	15-may-25	Pruebas MME
React	18.3.1	N/A	N/A	Meta Platforms, Inc	15-may-25	Pruebas MME
PostgreSQL	16.8	N/A	N/A	PostgreSQL Global Development Group	15-may-25	Pruebas MME
PostGIS	15 - 3.3	N/A	N/A	Refractions Research	15-may-25	Pruebas MME

Ilustración 25. Control de versionamiento. Fuente: imagen elaborada por ADA

10. Procesamiento y Visualización

El proceso de tratamiento y presentación de los datos en el aplicativo web se organiza mediante una lógica eficiente que se apoya principalmente en operaciones asincrónicas. Estas operaciones permiten gestionar y procesar grandes volúmenes de información sin que los usuarios experimenten demoras o bloqueos en la interfaz. Además, se optimiza la consulta a bases de datos tanto relacionales como geoespaciales, asegurando que los resultados se obtengan de forma rápida y precisa.

En cuanto a la visualización, esta ocurre en tiempo real, lo que significa que a medida que los datos se procesan, los resultados se reflejan inmediatamente en la interfaz del usuario. Para ello, se utilizan componentes dinámicos que realizan el rendering (o representación visual) de diversas formas de presentación de datos, como tablas, gráficos y mapas. Estos elementos visuales se actualizan constantemente según los datos que se procesan, brindando al usuario una experiencia fluida e interactiva.

La estructura del frontend se ha diseñado para ofrecer una experiencia de exploración óptima. Los usuarios pueden interactuar con los datos de manera eficiente, realizando filtrados y generando vistas jerárquicas que permiten explorar la información desde diferentes perspectivas, como territorio, año o variable. Este enfoque hace que la navegación sea intuitiva, permitiendo una comprensión detallada de los datos sin dificultades.

10.1 Descripción del Proceso

10.1.1 Backend (Node.js / Python)

1. El backend recibe las peticiones de los usuarios a través de parámetros de filtrado que se envían mediante la petición GET. Estos parámetros incluyen detalles como el año, el territorio y la variable a consultar, entre otros.
2. Luego, se ejecuta una consulta optimizada a las bases de datos relacionales, como PostgreSQL, y bases de datos geoespaciales, como PostGIS, utilizando un ORM (Object-Relational Mapping) como Prisma o SQLAlchemy. Estas consultas pueden incluir funciones de agregación como AVG (promedio), SUM (suma) y COUNT (conteo), lo que permite realizar cálculos sobre los datos de forma eficiente.
3. Después de obtener los datos de la base de datos, se aplica la lógica de negocio definida previamente, la cual establece las reglas y condiciones específicas que deben cumplirse para el procesamiento de los datos.
4. Finalmente, los resultados se estructuran en un formato JSON (JavaScript Object Notation) que contiene la información relevante para ser utilizada en el frontend. La estructura de la respuesta JSON se organiza de la siguiente manera:

```
json

{
  "territorio": "Boyacá",
  "anio": 2023,
  "indicadores": [
    { "nombre": "Cobertura Eléctrica", "valor": 97.5, "unidad": "%" }
  ]
}
```

Ilustración 26. Descripción del Proceso. Fuente: Imagen elaborada por ADA

Nota: Para conocer el detalle ver anexo 2. Documentación técnica del código Back End.

10.1.2 Frontend (React.js)

1. En el frontend, la visualización de los datos se organiza mediante componentes de visualización condicional, que se activan según el tipo de dato que se desea mostrar. Estos componentes incluyen VistaTabla, VistaGráfica y VistaMapa, que permiten al

usuario visualizar los datos en diferentes formatos dependiendo de la información seleccionada.

2. Se integran bibliotecas como **Chart.js** y **Recharts** para la creación de gráficos dinámicos, los cuales permiten representar de manera clara y precisa los datos en forma de líneas, barras, sectores, entre otros. Estos gráficos se ajustan de forma automática a las actualizaciones de los datos procesados, garantizando que el usuario siempre tenga la información más reciente.
3. Los mapas, otra parte fundamental de la visualización, se renderizan utilizando **Leaflet.js**. Estos mapas incluyen capas GeoJSON que se extraen del backend y se visualizan en tiempo real. Los mapas interactivos permiten al usuario explorar los datos geoespaciales con facilidad, utilizando herramientas como el zoom y la visualización de diferentes capas de información.
4. Además, se utilizan componentes de presentación reutilizables, como tarjetas e insignias, para mostrar estadísticas clave y otra información relevante de manera destacada. Estos componentes son visualmente atractivos y se diseñan para facilitar la comprensión de los datos, proporcionando una experiencia de usuario clara y eficiente.

Nota: Para conocer el detalle ver anexo 1. Documentación técnica del código Front End.

10.2 Ejemplo del flujo de visualización

El siguiente ejemplo muestra cómo funciona la carga y visualización de los datos en el frontend utilizando React.js:

```
jsx

useEffect(() => {
  fetch('/api/indicadores?anio=2023&departamento=Antioquia')
    .then(res => res.json())
    .then(data => setIndicadores(data));
}, []);
```

Ilustración 27. Ejemplo del flujo de visualización. Fuente: Imagen elaborada por ADA

En este ejemplo, se realiza una solicitud al servidor para obtener los datos correspondientes al año 2023 y al territorio de Antioquia. Una vez obtenidos los datos, se actualiza el estado

del componente con la información recibida, lo que desencadena una actualización en la interfaz de usuario con los nuevos datos.

10.3 Ejemplo Diagrama de Flujo Carga de Datos.

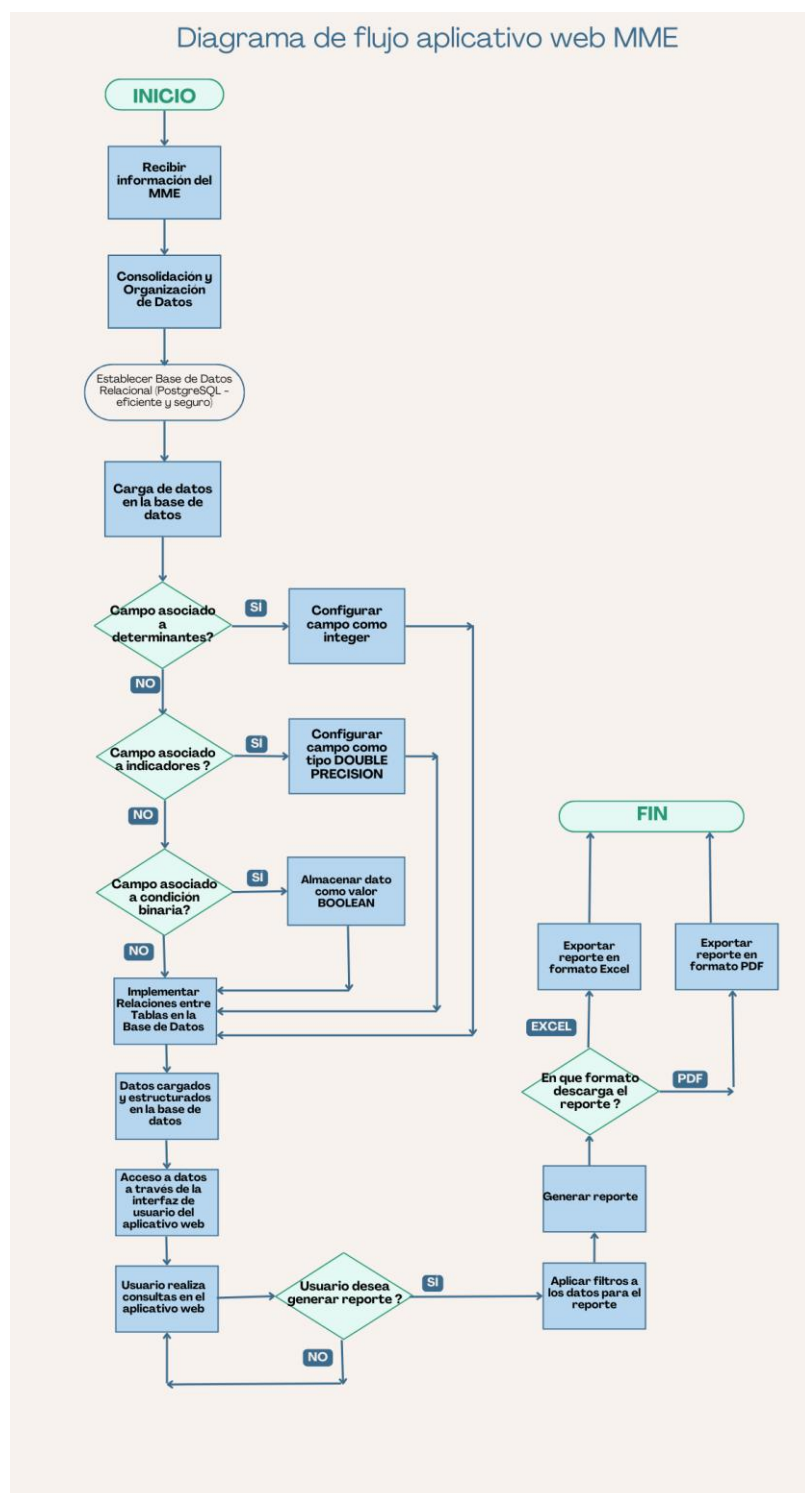


Ilustración 28. Diagrama de flujo carga de datos

10.4 Interacciones destacadas

El sistema también permite una serie de interacciones dinámicas que enriquecen la experiencia de usuario:

- **Escala de colores autoajustables:** Las escalas de colores de los mapas se ajustan de manera automática de acuerdo a los determinantes seleccionados, asegurando que la información sea coherente y fácil de comprender, el sistema maneja dicha escala en porcentajes de 0% (color verde) a 100% (color rojo), pasando por el 50% (color amarillo).
- **Sincronización entre filtros y contenido visual:** La selección de filtros por parte del usuario (como territorio, año, variable) se sincroniza instantáneamente con el contenido visual. Esto significa que la interfaz siempre refleja de forma precisa los datos que el usuario ha solicitado, mejorando la interacción y la usabilidad del sistema.

Este enfoque integral de procesamiento y visualización garantiza que los usuarios puedan explorar y analizar los datos de manera eficiente y con una experiencia fluida y atractiva.

11. Entrega y Despliegue del Sistema

El proceso de despliegue del sistema se organiza de manera eficiente mediante el uso de contenedores Docker, los cuales son gestionados y orquestados a través de Docker Compose. Este enfoque es utilizado en todos los entornos, incluyendo desarrollo, pruebas y producción, lo que asegura una consistencia en el manejo y la operación del sistema a lo largo de todas las etapas de su ciclo de vida.

El flujo del proceso de entrega está cuidadosamente diseñado para la verificación continua de la calidad del código y la ejecución de scripts que facilitan la migración de datos entre versiones. Esto no solo optimiza el tiempo de desarrollo, sino que también minimiza los riesgos asociados con errores manuales durante la entrega del sistema.

Es importante destacar que el sistema se despliega en una infraestructura proporcionada por el Ministerio de Minas y Energía (MME).

11.1 Requisitos mínimos para el correcto funcionamiento del aplicativo web.

Hardware: 4 Cores, 32 RAM, 500 ROM.

Software: SO Linux.

Navegadores: Google Chrome, Mozilla Firefox, Microsoft Edge y Apple Safari.

Antes de comenzar con cualquier instalación, resulta esencial validar que el sistema en el que se va a trabajar cumpla con una serie de prerequisites técnicos. Estos requisitos garantizan que los servicios a desplegar funcionen correctamente y que los contenedores Docker puedan operar sin restricciones a nivel de red, seguridad o permisos. A continuación, se detallan cada uno de los prerequisites necesarios:

11.2 Cuenta de Docker

Es indispensable disponer de una cuenta de Docker activa y vinculada a la máquina local desde la cual se realiza el despliegue. Esta cuenta permite autenticar operaciones como la descarga de imágenes desde el Docker Hub, el acceso a repositorios privados (en caso de ser necesario) y la gestión de contenedores a través de la CLI de Docker. En caso de que el usuario aún no posea una cuenta registrada, se proporciona más adelante una sección con instrucciones paso a paso para su creación y configuración.

11.3 Puertos de red habilitados

Para que los distintos servicios que conforman la aplicación web puedan comunicarse y estar accesibles tanto interna como externamente, es imprescindible que ciertos puertos estén habilitados en el archivo de configuración `docker-compose.yml`, así como en el firewall del servidor y cualquier otro dispositivo de red intermedio. A continuación, se enumeran los puertos que deben estar disponibles:

- **Puerto 80:** Canal estándar para tráfico HTTP. Permite el acceso público al sitio web sin necesidad de cifrado.
- **Puerto 3306:** Utilizado por el motor de base de datos MySQL. Necesario para la conexión entre los contenedores que manejan la capa de aplicación y la base de datos.
- **Puerto 8080:** Usado comúnmente para servidores de aplicaciones en modo desarrollo o pruebas. Puede estar asociado al backend de la aplicación.
- **Puerto 3030:** Habitualmente empleado por interfaces de usuario, paneles de administración o herramientas de visualización como Grafana o similares.

11.4 Muestra del proceso de despliegue.

Para conocer el detalle dirigirse al anexo 3. Evidencia Despliegue en ambiente MME.

```
C:\Windows\system32\cmd.exe
C:\Users\Jair.Lizcano\OneDrive\Escritorio\Frontend_Fenog>docker build -t frontend_fenog:latest .
[+] Building 1.99 (12/15) FPM3SHD
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 531B
=> WARN: FrontCasing: 'as' and 'FROM' keywords' casing do not match (line 1)
=> [internal] load metadata for docker.io/library/nginx:alpine
=> [internal] load metadata for docker.io/library/node:20-alpine
=> [internal] load .dockerignore
=> => transferring context: 4B
=> [stage-1 1/5] FROM docker.io/library/nginx:alpine@sha256:65645c7b6a661892a80b3b96d74320a18dd2f3f17a54ef4b76f8e2f2a18
=> => resolve docker.io/library/nginx:alpine@sha256:65645c7b6a661892a80b3b96d74320a18dd2f3f17a54ef4b76f8e2f2a18
=> [builder 1/6] FROM docker.io/library/node:20-alpine@sha256:49f6e7f5721729c58b0947244222f6af2876220c4d4d28e931c92c
=> => resolve docker.io/library/node:20-alpine@sha256:49f6e7f5721729c58b0947244222f6af2876220c4d4d28e931c92c
=> [internal] load build context
=> => transferring context: 19.43kB
=> CACHED [builder 2/6] WORKDIR /app
=> CACHED [builder 3/6] COPY package*.json ./
=> CACHED [builder 4/6] RUN npm ci
=> CACHED [builder 5/6] COPY . .
=> CACHED [builder 6/6] RUN npm run build
=> CACHED [stage-1 2/3] COPY --from=builder /app/dist /usr/share/nginx/html
=> CACHED [stage-1 3/3] COPY nginx.conf /etc/nginx/conf.d/default.conf
=> exporting to image
=> => exporting layers
=> => exporting manifest sha256:45ce10020c3823aef321b73c7f28a5a5fe0a999ff084feed18d23083fdd138
=> => exporting config sha256:13432238100b1f0750802b427a6d42edf8f4e9909fc282ce144ce
=> => exporting attestation manifest sha256:c78d83a0dc8f7a84dc39d93ad07c4f65ad19c55ad16550891ca2cb958eccbde
=> => exporting manifest list sha256:29cdc8fa6f37585214ff7c2827dae2d03925e6bac78a20794d755b6b779ffe1
=> => naming to docker.io/library/frontend_fenog:latest
=> => unpacking to docker.io/library/frontend_fenog:latest

View build details: docker-desktop://dashboards/build/desktop-linux/desktop-linux/1jgcw28pww54khyz6m0xxtt

1 warning found (use docker --debug to expand):
- FrontCasing: 'as' and 'FROM' keywords' casing do not match (line 1)

C:\Users\Jair.Lizcano\OneDrive\Escritorio\Frontend_Fenog>docker images
REPOSITORY          TAG         IMAGE ID      CREATED      SIZE
frontend_fenog      latest      29cdc8fa6f37  18 minutes ago  90.8MB
backend_fenog       latest      3d18efcf3b39  8 days ago    34MB
postgres            17.2       fe4efc6981dd  5 months ago  614MB

C:\Users\Jair.Lizcano\OneDrive\Escritorio\Frontend_Fenog>docker save -o frontend_fenog.tar frontend_
```

Ilustración 29. Proceso de despliegue. Fuente: Imagen elaborada por ADA.

11.5 Entrega del Software

La modalidad de despliegue del software se basa en el uso de contenedores Docker, lo que ofrece una forma flexible y aislada de gestionar las diferentes partes del sistema. Cada componente del sistema se despliega en su propio contenedor, lo que permite un control preciso sobre los recursos y facilita la escalabilidad y el mantenimiento.

Los detalles de la infraestructura del sistema de contenedores son los siguientes:

- **Frontend:** La interfaz de usuario se desarrolla utilizando **React.js**, un framework de JavaScript moderno y eficiente. Este frontend es servido por **Nginx**, un servidor web que actúa como proxy inverso, garantizando una entrega rápida y eficiente de los contenidos estáticos y optimizando la carga del sistema.
- **Backend:** La parte de servidor que gestiona la lógica de negocio está construida con **Node.js** y **Express**. En caso de que se decida implementar el backend en Python, se utilizará **FastAPI**, un framework altamente eficiente y rápido para crear aplicaciones API.
- **Base de Datos:** El sistema utiliza **PostgreSQL** como gestor de bases de datos relacional, con la extensión **PostGIS** para soportar funciones geoespaciales, lo que permite una gestión avanzada de datos geográficos dentro del sistema.

Para orquestar estos contenedores, se emplea el archivo **docker-compose.yml**, que facilita la gestión de las redes internas y la persistencia de los datos mediante volúmenes. Docker Compose asegura que todos los contenedores trabajen en conjunto de manera fluida y coordina su interacción de forma eficiente.

11.5.1 Protocolo de la Entrega Técnica

La entrega técnica del sistema sigue una serie de pasos bien definidos para garantizar que todas las partes del sistema se entreguen correctamente y estén listas para su implementación (Ver en la página 50 del presente documento el enlace que contiene el detalle del control de Versionamiento de las herramientas). Estos pasos incluyen:

1. **Empaquetado del sistema:** El software es empaquetado como imágenes Docker, lo que permite que se ejecute de manera consistente en cualquier entorno.
2. **Entrega de archivos clave:** El equipo del MME recibe el archivo **docker-compose.yml** y el fichero **.env**, los cuales contienen las configuraciones necesarias para la correcta ejecución del sistema en los entornos de pruebas y producción.
3. **Pruebas en entorno de pruebas:** Se realiza la ejecución de la entrega en un entorno de pruebas proporcionado por el MME. Esto incluye pruebas de validación funcional y pruebas de carga para asegurar que el sistema responda adecuadamente bajo condiciones reales.
4. **Firma del acta de entrega:** Una vez realizadas las pruebas de validación, se firma un acta de entrega técnica que incluye un checklist detallado de los requisitos de funcionalidades, asegurando que el sistema cumple con todas las especificaciones previamente acordadas.

11.5.2 Propuestas de CI/CD

Para garantizar un proceso de desarrollo ágil y sin errores, se propone el uso de **GitHub Actions**, que automatiza una serie de pasos en el ciclo de vida del software:

- **Linter y pruebas automáticas:** Cada vez que se realiza un commit en el código, se ejecutan linter y pruebas automáticas para verificar que el código cumple con los estándares de calidad y no introduce errores.
- **Construcción de imágenes Docker:** Las imágenes Docker se construyen automáticamente como parte del flujo de trabajo de CI/CD, lo que asegura que siempre se estén utilizando las versiones más actualizadas del software.
- **Publicación de imágenes:** Una vez construidas, las imágenes Docker se publican en un registro privado, como **GitHub Packages** o un **Docker Hub privado**, para que puedan ser utilizadas en los entornos de pruebas y producción de manera segura y controlada.

11.6 Requerimientos del Entorno de Despliegue (MME)

El sistema requiere de ciertos componentes de infraestructura para funcionar correctamente en los entornos proporcionados por el MME:

- **Sistema operativo:** Ubuntu Server 22.04 LTS, una distribución estable y ampliamente soportada, que garantiza un entorno de ejecución confiable y seguro.

- **Docker y Docker Compose:** Se requieren versiones **Docker $\geq 24.x$** y **Docker Compose $\geq 2.x$** , para asegurar que todas las funcionalidades de orquestación y contenedores sean compatibles y optimizadas.
- **Dominio institucional:** El sistema debe estar accesible a través de un **dominio institucional** o un **subdominio público**, para garantizar una dirección web oficial y coherente con la identidad del MME.

Con estos requisitos y el proceso de entrega bien definido, se asegura que el sistema esté preparado para su funcionamiento en los entornos de producción, con una base sólida para su mantenimiento y evolución.

12. Diccionario de datos

12.1 Tabla: `spatial_ref_sys`

La imagen proporciona una vista detallada de la definición de la tabla **`spatial_ref_sys`**, componente esencial del esquema espacial en PostGIS. Esta tabla actúa como catálogo de referencia para los sistemas de coordenadas utilizados en las operaciones geoespaciales, permitiendo la correcta georreferenciación y transformación de datos espaciales.

Cada columna de la tabla está definida con precisión para asegurar la integridad referencial:

- **`srid`** (INTEGER, NOT NULL): Identificador único del sistema de referencia espacial dentro de la base de datos.
- **`auth_name`** (VARCHAR(256), NOT NULL): Nombre de la autoridad que define el sistema de referencia, como por ejemplo *EPSG*.
- **`auth_srid`** (INTEGER, NOT NULL): Código del sistema de referencia proporcionado por la autoridad correspondiente.
- **`srtext`** (VARCHAR(2048), NOT NULL): Representación textual del sistema en formato **Well-Known Text (WKT)**.
- **`proj4text`** (VARCHAR(2048), NOT NULL): Representación textual del sistema en formato **PROJ4**, utilizado comúnmente en transformaciones cartográficas.

La obligatoriedad de todos los campos asegura que cada definición de sistema espacial esté completa y correctamente especificada, favoreciendo la interoperabilidad entre distintos sistemas y aplicaciones geoespaciales.

## Tabla: <code>spatial_ref_sys</code>			
Sistema de referencia espacial para PostGIS			
Columna	Tipo	Null	Descripción
<code>srid</code>	INTEGER	No	Identificador único del sistema de referencia espacial

auth_name	VARCHAR(256)	No	Nombre de la autoridad que define el sistema (ej. EPSG)
auth_srid	INTEGER	No	SRID definido por la autoridad
srttext	VARCHAR(2048)	No	Definición del sistema en formato WKT
proj4text	VARCHAR(2048)	No	Definición del sistema en formato PROJ4

Tabla 1 Tabla: spatial_ref_sys Fuente: ADA.

12.2 Tabla: municipios

La imagen proporciona una descripción estructurada de la tabla **municipios**, diseñada para almacenar la geometría de los polígonos que delimitan municipios, junto con la información complementaria necesaria para su correcta identificación y trazabilidad. Esta tabla es parte integral de un sistema de información espacial.

Cada columna se define de la siguiente manera:

- **id** (UUID, NOT NULL): Identificador único del municipio, garantizando unicidad en el registro.
- **geom** (GEOMETRY(MultiPolygon, 4326), NOT NULL): Geometría espacial del municipio, almacenada como un objeto **MultiPolygon** bajo el sistema de referencia espacial **EPSG:4326**.
- **created_at** (TIMESTAMPTZ, NOT NULL): Marca temporal que registra la fecha y hora de creación del registro, en formato de timestamp con zona horaria.
- **id_grilla** (INTEGER, NOT NULL): Identificador que asocia el municipio a una grilla o partición espacial específica.

La definición de todos los campos como no nulos asegura la consistencia e integridad de la información almacenada, elementos fundamentales para procesos de análisis espacial, reportes y operaciones de consulta avanzadas.

## Tabla: municipios			
Almacena la geometría de los polígonos de determinantes y metadata de los municipios			
Columna	Tipo	Null	Descripción
id	UUID	No	Identificador único del municipio
geom	GEOMETRY(MultiPolygon, 4326)	No	Geometría del municipio en formato MultiPolygon
created_at	TIMESTAMPTZ	No	Fecha y hora de creación del registro
id_grilla	INTEGER	No	Identificador de la grilla

Tabla 2 Tabla: municipios, Fuente: ADA.

12.3 Tabla: div_territorial_municipios

La imagen muestra la estructura de la tabla **div_territorial_municipios**, utilizada para representar la división territorial de un país o región a nivel municipal. Esta tabla integra información espacial y administrativa, permitiendo un manejo preciso de los datos de delimitación geográfica y atributos asociados.

Las columnas que componen esta tabla son:

- **id** (UUID, NOT NULL): Identificador único del registro, actuando como clave primaria.
- **geom** (GEOMETRY(MultiPolygon, 4326), NOT NULL): Geometría espacial del municipio en formato **MultiPolygon**, basada en el sistema de referencia **EPSG:4326**.
- **created_at** (TIMESTAMPTZ, NOT NULL): Marca temporal que indica la fecha y hora de creación del registro, con información de zona horaria.
- **mpio_cdmp** (TEXT, NOT NULL): Código único asignado al municipio.
- **mpio_cnmb** (TEXT, NOT NULL): Nombre del municipio.
- **dpto_ccdgo** (TEXT, NOT NULL): Código único asignado al departamento o entidad territorial de nivel superior.
- **dpto_cnmb** (TEXT, NOT NULL): Nombre del departamento o entidad territorial de nivel superior.

Todos los campos son obligatorios (NOT NULL), garantizando la integridad y completitud de los datos almacenados. La combinación de atributos espaciales y alfanuméricos permite la ejecución de consultas geográficas avanzadas y el desarrollo de soluciones de análisis territorial y visualización cartográfica.

## Tabla: div_territorial_municipios			
División territorial a nivel municipal			
Columna	Tipo	Null	Descripción
id	UUID	No	Identificador único del registro (Primary Key)
geom	GEOMETRY(MultiPolygon, 4326)	No	Geometría del municipio en formato MultiPolygon
created_at	TIMESTAMPTZ	No	Fecha y hora de creación del registro
mpio_cdmp	TEXT	No	Código del municipio
mpio_cnmb	TEXT	No	Nombre del municipio
dpto_ccdgo	TEXT	No	Código del departamento
dpto_cnmb	TEXT	No	Nombre del departamento

Tabla 3 Tabla: div_territorial_municipios, Fuente: ADA.

12.4 Tabla: div_territorial_zonas

La tabla describe la estructura de “**div_territorial_zonas**”, diseñada para representar la división territorial de un país a nivel de zonas geográficas y estructuras administrativas. Esta tabla integra tanto información identificativa de los departamentos y municipios como su pertenencia a distintas regiones y programas estratégicos.

La definición de los campos es la siguiente:

- **nc** (INTEGER, NOT NULL): Número consecutivo que actúa como clave primaria del registro.
- **dpto_ccdgo** (TEXT, NOT NULL): Código del departamento.
- **dpto_cnmb** (TEXT, NOT NULL): Nombre del departamento.
- **mpio_cdmp** (TEXT, NOT NULL): Código del municipio.
- **mpio_cnmb** (TEXT, NOT NULL): Nombre del municipio.
- **id_grilla** (INTEGER, NOT NULL): Identificador asociado a la grilla territorial.
- **caribe, andina, amazonia, orinoquia, pacifica, insular** (BOOLEAN, NOT NULL): Indicadores de pertenencia del municipio a las respectivas regiones naturales.
- **estructura_1, estructura_2, estructura_3** (BOOLEAN, NOT NULL): Indicadores de pertenencia a estructuras territoriales específicas, según criterios administrativos o de planificación.
- **pais** (BOOLEAN, NOT NULL): Indicador de alcance nacional.
- **zomac** (BOOLEAN, NOT NULL): Indica si el municipio pertenece a una **Zona Más Afectada por el Conflicto Armado**.
- **pdet** (BOOLEAN, NOT NULL): Indica si el municipio es parte de los **Programas de Desarrollo con Enfoque Territorial (PDET)**.

Todos los campos son obligatorios (NOT NULL), garantizando la completitud y precisión de los datos. Esta tabla es fundamental para realizar análisis espaciales, regionales y socioeconómicos, facilitando la implementación de políticas públicas diferenciadas por región o estructura.

## Tabla: div_territorial_zonas			
División territorial por zonas			
Columna	Tipo	Null	Descripción
nc	INTEGER	No	Número consecutivo (Primary Key)
dpto_ccdgo	TEXT	No	Código del departamento
dpto_cnmb	TEXT	No	Nombre del departamento
mpio_cdmp	TEXT	No	Código del municipio
mpio_cnmb	TEXT	No	Nombre del municipio

id_grilla	INTEGER	No	Identificador de la grilla
caribe	BOOLEAN	No	Indica si pertenece a la región Caribe
andina	BOOLEAN	No	Indica si pertenece a la región Andina
amazonia	BOOLEAN	No	Indica si pertenece a la región Amazonia
orinoquia	BOOLEAN	No	Indica si pertenece a la región Orinoquia
pacifica	BOOLEAN	No	Indica si pertenece a la región Pacífica
insular	BOOLEAN	No	Indica si pertenece a la región Insular
estructura_1	BOOLEAN	No	Pertenencia a Estructura 1
estructura_2	BOOLEAN	No	Pertenencia a Estructura 2
estructura_3	BOOLEAN	No	Pertenencia a Estructura 3
pais	BOOLEAN	No	Indicador nacional
zomac	BOOLEAN	No	Zona Más Afectada por el Conflicto Armado
pdet	BOOLEAN	No	Programas de Desarrollo con Enfoque Territorial

Tabla 4 Tabla: div_territorial_zonas, Fuente: ADA.

12.5 Tabla: divterritorialdepto_determinantes

La imagen muestra un **diccionario de datos** de la tabla llamada divterritorialdepto_determinantes, que contiene información de **determinantes a nivel departamental**.

El diccionario presenta las siguientes columnas:

- **Columna:** nombre del campo en la tabla.
- **Tipo:** tipo de dato (principalmente INTEGER y un campo TEXT).
- **Null:** indica si el campo permite valores nulos (en este caso, todos son "No").
- **Descripción:** breve explicación del contenido de cada campo.

Algunos ejemplos de campos son:

- dpto_codigo: Código del departamento (clave primaria).
- dpto_cnabr: Nombre del departamento.
- suma_s_111: Seguridad Indígena.
- suma_s_141: Áreas arqueológicas protegidas.

- suma_s_223: Manglar.
- suma_s_311: Títulos Vigentes.
- suma_s_431: Casco Urbano.

Cada campo suma_s_xxx corresponde a un tipo específico de determinante geográfico, ambiental o legal.

## Tabla: divterritorialdepto_determinantes			
Determinantes a nivel departamental			
Columna	Tipo	Null	Descripción
dpto_ccdgo	INTEGER	No	Código del departamento (Primary Key)
dpto_cnmb	TEXT	No	Nombre del departamento
suma_s_111	INTEGER	No	Resguardo Indígena
suma_s_112	INTEGER	No	Solicitud Coloniales - RIOC
suma_s_113	INTEGER	No	Solicitud Ancestrales - Dec 2333 de 2014
suma_s_114	INTEGER	No	Solicitud legalización
suma_s_115	INTEGER	No	Linea Negra
suma_s_121	INTEGER	No	CC de Comunidades Negras
suma_s_122	INTEGER	No	Área pretendida en titulación Colectiva
suma_s_123	INTEGER	No	Aspiraciones
suma_s_131	INTEGER	No	Constituidas
suma_s_132	INTEGER	No	Proceso
suma_s_141	INTEGER	No	Áreas arqueológicas Protegidas
suma_s_151	INTEGER	No	Zonas de Frontera
suma_s_161	INTEGER	No	Individual
suma_s_162	INTEGER	No	Colectiva
suma_s_163	INTEGER	No	Microzonas
suma_s_164	INTEGER	No	Decreto 2007

suma_a_221	INTEGER	No	Páramos
suma_a_222	INTEGER	No	Bosque Seco tropical
suma_a_223	INTEGER	No	Manglar
suma_a_231	INTEGER	No	Ley 2/59
suma_a_241	INTEGER	No	AICAS
suma_a_242	INTEGER	No	ZPRNRMA resolución 125 de 2021
suma_a_243	INTEGER	No	Bosques de Paz
suma_a_244	INTEGER	No	POMCAS
suma_a_245	INTEGER	No	Bancos de Habitat ICDE resolución 1051 de 2017
suma_a_251	INTEGER	No	SIPTA EN EVALUACION
suma_a_252	INTEGER	No	Licencias APROBADAS
suma_a_211	INTEGER	No	RUNAP
suma_t_311	INTEGER	No	Titulo Vigentes
suma_t_312	INTEGER	No	Solicitudes Vigentes
suma_t_313	INTEGER	No	Zona Minera Etnica
suma_t_321	INTEGER	No	Mapa de Tierras
suma_t_322	INTEGER	No	Gasodutos y Polductos
suma_t_331	INTEGER	No	Subestaciones
suma_t_332	INTEGER	No	Lineas contruidas STN
suma_t_333	INTEGER	No	Generación Actual
suma_t_334	INTEGER	No	Generacion Prospectiva
suma_t_335	INTEGER	No	ZNI
suma_n_411	INTEGER	No	Vías Principal
suma_n_421	INTEGER	No	Casco Urbano
suma_n_431	INTEGER	No	Puertos

suma_n_441	INTEGER	No	Navegables
suma_n_412	INTEGER	No	Vias todas

Tabla 5 Tabla: divterritorialdepto_determinantes, Fuente: ADA.

12.6 Tabla: divterritorialmuni_determinantes

La tabla divterritorialmuni_determinantes contiene información de **determinantes territoriales** a nivel **municipal**. Cada registro corresponde a un municipio del país e incluye el mismo conjunto de determinantes que la tabla departamental, pero desagregado a nivel de municipio.

## Tabla: divterritorialmuni_determinantes			
Determinantes a nivel municipal			
Columna	Tipo	Null	Descripción
mpio_cdmp	VARCHAR(10)	No	Código del municipio (Primary Key)
mpio_cnmb	VARCHAR(100)	No	Nombre del municipio
suma_s_111 a suma_n_412	INTEGER	No	Mismos campos que divterritorialdepto_determinantes pero a nivel municipal

Tabla 6 Tabla: divterritorialmuni_determinantes, Fuente: ADA.

12.7 Tabla: indicadores_territorial

La tabla indicadora territorial almacena **indicadores territoriales y de ranking** a nivel departamental y municipal. Los registros permiten evaluar múltiples dimensiones de desarrollo, infraestructura, servicios públicos, educación y desempeño territorial de cada unidad administrativa.

- Los tipos de datos utilizados son INTEGER y DOUBLE PRECISION, dependiendo si el valor es un conteo o un índice/ranking decimal.
- No se permiten valores nulos (NULL) en ningún campo para asegurar consistencia en los análisis.
- Se incluyen indicadores educativos, económicos, de infraestructura, sociales y de desarrollo humano.

## Tabla: indicadores_territorial			
Indicadores territoriales y rankings			
Columna	Tipo	Null	Descripción
dpto_ccdgo	INTEGER	No	Código del departamento

dpto_cnmb	VARCHAR(100)	No	Nombre del departamento
mpio_cdpmp	INTEGER	No	Código del municipio
mpio_cnmb	VARCHAR(100)	No	Nombre del municipio
pobl_rur	INTEGER	No	Población rural
anno_pobl_rur	INTEGER	No	Año del dato de población rural
pobl_urb	INTEGER	No	Población urbana
anno_pobl_urb	INTEGER	No	Año del dato de población urbana
pobl_tot	INTEGER	No	Población total
anno_pobl_tot	INTEGER	No	Año del dato de población total
nbi	DOUBLE PRECISION	No	Necesidades Básicas Insatisfechas
anno_nbi	INTEGER	No	Año del dato NBI
ipm	DOUBLE PRECISION	No	Índice de Pobreza Multidimensional
anno_ipm	INTEGER	No	Año del dato IPM
icee_cm	DOUBLE PRECISION	No	Índice de Cobertura de Energía Eléctrica - Cabecera Municipal
anno_icee_cm	INTEGER	No	Año del dato ICEE CM
icee_resto	DOUBLE PRECISION	No	ICEE - Resto del municipio
anno_icee_resto	INTEGER	No	Año del dato ICEE Resto
icee_tot	DOUBLE PRECISION	No	ICEE Total
anno_icee_tot	INTEGER	No	Año del dato ICEE Total
di_p_nal	DOUBLE PRECISION	No	Desempeño Integral - Puntaje Nacional
anno_di_p_nal	INTEGER	No	Año del dato DI Nacional
indesarrollo_mun	DOUBLE PRECISION	No	Índice de Desarrollo Municipal
anno_indesarrollo_mun	INTEGER	No	Año del dato Índice Desarrollo
cob_gres_efec	DOUBLE PRECISION	No	Cobertura de Gestión de Residuos Efectiva
anno_cob_gres_efec	INTEGER	No	Año del dato Cobertura Residuos
pibxhabitante	DOUBLE PRECISION	No	PIB per cápita
anno_pibxhabitante	INTEGER	No	Año del dato PIB per cápita

actividades_primarias	DOUBLE PRECISION	No	Valor actividades económicas primarias
anno_actividades_primarias	INTEGER	No	Año del dato actividades primarias
actividades_secundarias	DOUBLE PRECISION	No	Valor actividades económicas secundarias
anno_actividades_secundarias	INTEGER	No	Año del dato actividades secundarias
actividades_terciarias	DOUBLE PRECISION	No	Valor actividades económicas terciarias
anno_actividades_terciarias	INTEGER	No	Año del dato actividades terciarias
sumatoria_actividades	DOUBLE PRECISION	No	Suma total de actividades económicas
anno_sumatoria_actividades	INTEGER	No	Año del dato sumatoria actividades
proporción_para_el_dpto	DOUBLE PRECISION	No	Proporción respecto al departamento
anno_proporción_para_el_dpto	INTEGER	No	Año del dato proporción departamental
pobreza_monetaria_extrema	DOUBLE PRECISION	No	Índice de pobreza monetaria extrema
anno_pobreza_monetaria_extrema	INTEGER	No	Año del dato pobreza extrema
homicidios_x_cada_100000_habitan tes	DOUBLE PRECISION	No	Tasa de homicidios por 100,000 habitantes
anno_homicidios_x_cada_100000_h abitantes	INTEGER	No	Año del dato tasa homicidios
cobertura_de_alcantarillado	DOUBLE PRECISION	No	Porcentaje cobertura alcantarillado
anno_cobertura_de_alcantarillado	INTEGER	No	Año del dato cobertura alcantarillado
cobertura_de_acueducto	DOUBLE PRECISION	No	Porcentaje cobertura acueducto
anno_cobertura_de_acueducto	INTEGER	No	Año del dato cobertura acueducto
poblacion_en_condicion_de_miseria	DOUBLE PRECISION	No	Porcentaje población en miseria
anno_poblacion_en_condicion_de_ miseria	INTEGER	No	Año del dato población en miseria
di_p_nal_norm	DOUBLE PRECISION	No	Desempeño Integral Nacional Normalizado
pobl_rur_pct	DOUBLE PRECISION	No	Porcentaje población rural
pobl_rurt_pct	DOUBLE PRECISION	No	Porcentaje población rural total

pobl_urb_pct	DOUBLE PRECISION	No	Porcentaje población urbana
pobl_tott_pct	DOUBLE PRECISION	No	Porcentaje población total
di_p_dep	DOUBLE PRECISION	No	Desempeño Integral Departamental
a_km2	DOUBLE PRECISION	No	Área en kilómetros cuadrados
rank_dep *	INTEGER	No	Rankings departamentales de diversos indicadores
rank_nac *	INTEGER	No	Rankings nacionales de diversos indicadores

Tabla 7 Tabla: indicadores_territorial, Fuente: ADA.

13. Código Fuente del Desarrollo

El Código Fuente del desarrollo correspondiente al proyecto, será entregado como un documento adjunto para su revisión y análisis en formato txt. Dicho documento estará guardado en el servidor en la siguiente ruta:

/home/admgear/Documents/Instaladores/Codigo fuente/codigo fuente.txt

14. Garantía sobre el Aplicativo web

El proceso de garantía para el aplicativo web está diseñado para asegurar el correcto funcionamiento de la aplicación después de su entrega.

Este mantenimiento correctivo está enfocado exclusivamente en la solución de los errores de software que puedan surgir durante el uso de la aplicación, siempre y cuando dichos errores se detecten en las condiciones de operación que previamente se han validado.

La garantía cubre los defectos que afecten las funcionalidades originalmente desarrolladas conforme a las especificaciones establecidas en este documento técnico.

Quedan fuera de esta cobertura aquellas modificaciones o ampliaciones que impliquen mejoras, adición de nuevas funcionalidades o cambios sustanciales respecto al diseño inicial acordado.

14.1 Condiciones de la Garantía

14.1.1 Plazo de cobertura

La garantía tiene una duración de 30 días calendario, contados a partir de la formalización de la aprobación del aplicativo web por parte del Ministerio de Minas y Energía (MME). Este plazo permite un seguimiento adecuado de cualquier incidencia o error que pueda presentarse en el funcionamiento de la aplicación bajo las condiciones de uso inicialmente definidas.

14.1.2 Cobertura

La garantía cubre exclusivamente los siguientes aspectos:

- **Corrección de errores de funcionalidad (bugs):** Se solucionarán los problemas reproducibles en los ambientes de pruebas establecidos. Esto incluye fallos técnicos que afecten la operatividad de la aplicación tal como fue originalmente concebida.
- **Inconsistencias en la visualización o exportación de datos:** Si se presentan fallos en la representación o exportación de datos debido a errores derivados del código implementado, estos serán corregidos durante el periodo de garantía.
- **Fallas en los mecanismos de carga, filtrado, comparación, visualización y reporte de datos:** Cualquier fallo relacionado con la ejecución de las funcionalidades descritas en este documento técnico será abordado en el marco de la garantía.

14.2 Procedimiento para la Gestión de Incidencias

El proceso de gestión de incidencias está claramente definido para asegurar una respuesta rápida y eficaz a cualquier problema detectado en el funcionamiento del aplicativo web. A continuación, se detallan los pasos a seguir:

14.2.1 Reporte de incidentes

Cuando se detecta un problema, el equipo del Ministerio de Minas y Energía (MME) debe comunicarlo enviando un ticket al correo electrónico asignado (luis.gutierrez@ada). Este ticket debe incluir la siguiente información:

- **Descripción detallada de la incidencia:** Una explicación clara y precisa sobre el problema observado, para facilitar su identificación.
- **Capturas de pantalla:** Imágenes que muestren visualmente el error o problema, ayudando a la comprensión del contexto.

- **Datos de entrada empleados:** Información sobre los archivos o valores específicos utilizados al momento de experimentar la incidencia.
- **Resultado esperado vs. resultado observado:** Una comparación entre lo que se esperaba que ocurriera y lo que realmente sucedió, lo que permitirá identificar más fácilmente el origen del error.

14.2.2 Reproducción del error

Una vez recibido el reporte, el equipo de ADA procederá a intentar reproducir la incidencia en un entorno de pruebas controlado. Esto es fundamental para verificar que el error es reproducible y se encuentra dentro del alcance de la garantía. Solo de esta manera se podrá proceder con la identificación precisa del fallo.

14.2.3 Solución y entrega

Una vez que el error ha sido reproducido y comprendido, se agendará la corrección necesaria (fix). Este arreglo se entregará al Ministerio de Minas y Energía (MME), acompañado de instrucciones claras sobre cómo aplicar la solución, o bien, si se acuerda, se desplegará directamente en el entorno de pruebas para su validación.

14.3 Documentación de garantía

Cada corrección realizada será debidamente documentada, incluyendo el detalle de la solución implementada y la versión del código afectado. Esta documentación se actualizará de acuerdo con las modificaciones realizadas. Finalmente, el Ministerio de Minas y Energía (MME) deberá confirmar que el error ha sido resuelto correctamente para que el incidente se cierre formalmente.

Este proceso garantiza que el aplicativo web no solo esté alineado con las especificaciones iniciales, sino que también pueda mantenerse operativo a largo plazo, con una respuesta rápida ante cualquier incidencia que afecte su funcionalidad en el periodo de garantía.

Nota: Una vez cumplido el proceso anteriormente descrito, la fábrica tendrá 15 días hábiles para desarrollar los cambios o ajustes correspondientes al aplicativo.

15. Aprobación

A continuación, firman aprobando:

Nombre	Cargo	Firma	Fecha
Responsable de Área (MME)			
Adriana Patricia Cante Chaparro	Supervisora Contrato designada por FENOGE.		
María Fernanda Ramírez	Supervisor Convenio. MME-OAAS		