Retos de ciudad

Bogotá tiene 5 en Innovación

Reto de ciudad 1: Datos con historia

Etapa de Implementación

1. **Documento técnico de la comprensión del problema y el detalle técnico de la solución a implementar.**

Marzo 2025

Versión 2

OFICINA DE LABORATORIO DE CIUDAD-SECRETARÍA DISTRITAL DE PLANEACIÓN
CORPORACIÓN MALOKA DE CIENCIA, TECNOLOGÍA E INNOVACIÓN
IGG-INVERSIONES GUTIÉRREZ GARCÍA

#5 EN INNOVACIÓN

RETO DE CIUDAD 1

DATOS CON HISTORIA

1. Introducción **3**

2. Análisis del problema **4**

2.1. Oficina de Participación y Diálogo Ciudadano (OPDC) 6

2.2. Subsecretaría de Políticas Públicas y Planeación Social y Económica (SPPPSE) 6

2.3. Subsecretaría de Planeación de la Inversión (SPI) 6

3. Solución del problema **7**

3.1. Arquitectura de la solución 7

3.2. Stack Tecnológico 9

3.3. Procesamiento y gestión de datos 10

3.4. Flujo de procesamiento 11

3.5. Seguridad con los datos 13

4. Metodología de trabajo **13**

4.1. Cronograma 14

4.2. Backlog de requerimientos 15

4.3. Matriz de riesgos técnicos 17

4.4. Plan de pruebas de usuario 17

5. Anexos **18**

6. Glosario **18**

7. Referencias **21**

1. Introducción

El presente documento, es el primero de tres informes, que documentan el desarrollo de la Etapa de implementación; la última etapa de la estrategia de Innovación abierta, cuyo alcance es la construcción de un producto mínimo viable (PMV) asociado a la solución al Reto planteado en el marco de esta estrategia. Se construyó en conjunto entre los equipos de Maloka e IGG Inversiones Gutiérrez García, equipo cuya propuesta de solución al Reto fue seleccionada en la etapa de Prototipado. **Este documento presenta la información base de la etapa de implementación de la solución propuesta**, desde el análisis de la problemática, descripción detallada de la solución y los aspectos técnicos asociados, hasta la metodología de desarrollo e implementación del producto propuesto.

La estrategia de innovación abierta *Retos de Ciudad* invitó a personas de la academia, de organizaciones y de la comunidad en general, para presentar propuestas de desarrollo tecnológico basado en ciencia de datos que respondan al reto: *¿Cómo automatizar la recolección, análisis y categorización de la información cualitativa, gestionada por las diferentes dependencias de la Secretaría Distrital de Planeación (SDP), a través de un desarrollo tecnológico, para que esta información se pueda integrar efectivamente en la toma de decisiones y optimizar la planificación urbana?*

Esta estrategia se desarrolló en el marco del Convenio Especial de Cooperación No. 997-2024 suscrito entre la Oficina de Laboratorio de Ciudad de la Secretaría Distrital de Planeación y la Corporación Maloka de Ciencia, Tecnología e Innovación, cuya metodología se divide en tres momentos: Convocatoria, Prototipado e Implementación.

2. Análisis del problema

Las dependencias de la Secretaría Distrital de Planeación enfrentan múltiples desafíos relacionados con el manejo de información cualitativa, clave para el cumplimiento de sus funciones misionales. Estas funciones incluyen la elaboración y seguimiento de instrumentos y planes estratégicos como el POT – Plan de Ordenamiento Territorial, el PDD – Plan Distrital de Desarrollo, la evaluación de políticas públicas, la medición de impactos y el control de la inversión realizada. Sin embargo, el volumen, la diversidad y las características no estructuradas de la información representan obstáculos importantes para lograr resultados eficientes y efectivos. Uno de los principales retos es la gran cantidad de documentos y datos no estructurados que los funcionarios deben revisar manualmente, lo que requiere un tiempo significativo y prolonga el cumplimiento de tareas críticas. La falta de estandarización en los formatos de información y la presencia de errores o inconsistencias dificultan su análisis, incrementando la carga operativa de los equipos.

Además, la SDP recibe información de múltiples fuentes (estructuradas y no estructuradas, incluyendo documentos técnicos, hojas de cálculo y otros formatos) y entidades distritales, cada una con niveles de detalle y formatos diferentes. Esta diversidad añade complejidad a la consolidación y análisis integral de los datos, ya que los funcionarios deben validar y depurar la información manualmente para garantizar su coherencia y alineación con los objetivos estratégicos. Otro desafío crítico es la falta de herramientas tecnológicas adecuadas para automatizar los procesos de validación y análisis de información cualitativa. Los mecanismos actuales, aunque funcionales, no son suficientes para abordar de manera eficiente la identificación de discrepancias, la alineación temática o la evaluación de la relevancia y precisión de los datos. Esto afecta tanto la evaluación de productos como la medición de impactos y procesos participativos, donde la claridad y precisión de la información son esenciales. Uno de los principales obstáculos radica en la necesidad de revisar manualmente grandes cantidades de documentos, lo cual consume tiempo y recursos significativos. Actualmente, la revisión manual de seis políticas públicas puede tomar entre una y dos semanas para una persona. Esta situación retrasa el cumplimiento de tareas misionales y limita la capacidad de los equipos para enfocarse en actividades estratégicas.

En este contexto, la gestión de la información cualitativa enfrenta problemas relacionados con su calidad, extensión y diversidad, los cuales generan una mayor carga de trabajo para los funcionarios, afectando los tiempos y la eficiencia de las tareas misionales de la SDP.

1. **Falta de Estandarización y Calidad de los Datos**

El análisis de información cualitativa es complejo debido a la falta de estandarización en los formatos de datos y a la presencia de errores, inconsistencias y datos incompletos. Esto afecta la coherencia y la alineación de la información con los objetivos estratégicos de la organización. Adicionalmente, la diversidad de fuentes y niveles de detalle en la información incrementa la carga de trabajo para la validación manual, lo que impacta negativamente en los tiempos y la calidad de los resultados.

2. Carencia de Herramientas Tecnológicas Adecuadas

Los procesos actuales carecen de herramientas tecnológicas que permitan automatizar la validación y el análisis de información cualitativa. Los mecanismos existentes son funcionales pero insuficientes para abordar de manera eficiente tareas como la identificación de discrepancias, la alineación temática y la evaluación de la relevancia de los datos. Esto limita la capacidad de generar reportes confiables y dificulta la toma de decisiones basadas en evidencia.

En este proyecto, nos centramos en la extracción de información de fuentes específicas proporcionadas. Para SPI la matriz de Excel con la información de avances y retrasos para cada uno de los indicadores del plan de acción del Plan de Desarrollo Distrital (PDD), Para la SPPPSE, la matriz de Excel con información cualitativa y todo lo relacionado con enfoques para cada uno de los productos de las Políticas Públicas. Para la OPDC, las actas de diferentes instrumentos generados en formato PDF siguiendo los lineamientos acordados, estructura a nivel del acta, tabla de aportes, criterios y objetivos y finalmente archivo en Excel con matriz de carga de aportes ciudadanos. Estas fuentes serán la base para los procesos de analítica necesarios para abordar los retos identificados.

3. Procesamiento de información

Una vez que los datos han sido extraídos, se han implementado procesos avanzados como el reconocimiento de entidades nombradas. Este método permitirá identificar conceptos clave como nombres de personas, entidades, lugares, fechas y otros datos relevantes para los objetivos del proyecto. La información estructurada y sus metadatos serán almacenados en bases de datos especializadas, asegurando su disponibilidad para análisis posteriores.

En relación con los retos planteados, se diseñaron estrategias generales que permiten maximizar la eficiencia y los resultados:

- Procesamiento de políticas públicas: Actualmente, la revisión manual de 6 políticas toma entre 1 y 2 semanas para una persona. Esta solución reducirá este tiempo en al menos un 60%, lo que liberará recursos para actividades estratégicas.
- Alineación con necesidades de los usuarios: Se abordaron las expectativas de los diferentes actores mediante un enfoque analítico robusto que optimice los recursos disponibles, sin incluir inicialmente integraciones complejas como autenticación LDAP, visualización en ArcGIS o PowerBI. Esto asegura que las soluciones entregadas sean viables dentro del alcance establecido.
- Periodicidad de actualización de la información: Se acuerda que al menos la información de los planes, actas de seguimientos y demás documentos involucrados se estará actualizando al menos dos veces al año.

De manera específica para cada área se plantea:

2.1. Oficina de Participación y Diálogo Ciudadano (OPDC)

- Análisis de actas y documentos clave: Se automatiza la extracción y estructuración de información contenida en actas y documentos relevantes, permitiendo su visualización y análisis desde perspectivas más funcionales y orientadas a los objetivos del PDD.

- Entendiendo la importancia de la participación ciudadana y el impacto que tiene en la confianza pública, tal como se propuso con el flujo de procesos implementado, se pretende optimizar la recolección, trazabilidad e intercambio de información mediante un sistema que permite administrar la depuración de aportes irrelevantes y evalúa dimensiones clave como relevancia estratégica, impacto, factibilidad, innovación y sostenibilidad. Así, se facilita el seguimiento de los aportes ciudadanos, garantizando su relación con áreas específicas de trabajo, todo a través de una interfaz gráfica que simplifique la visualización categorizada y permita a los usuarios apropiarse del área de su interés.

2.2. Subsecretaría de Políticas Públicas y Planeación Social y Económica (SPPPSE)

- Reconociendo la importancia de los datos para las políticas públicas de la ciudad y los desafíos asociados a su aprovechamiento, se ha plantea una solución innovadora que facilita la categorización de aportes en variables estratégicas y la generación automatizada de contribuciones alineadas con el Plan Distrital de Desarrollo y los ODS. Este sistema de analítica permite identificar errores de ortografía, información incompleta y duplicidad de aportes, definiendo la categoría correspondiente, evaluando su impacto, relacionándolo con indicadores y resultados y generando reportes que evidencian alertas, logros y acciones de mejora.

2.3. Subsecretaría de Planeación de la Inversión (SPI)

- Se incluye un sistema de análisis de texto, impulsado por inteligencia artificial, para identificar tendencias claves en la información recolectada por las entidades distritales, clasificándose por temáticas estratégicas. Este sistema implementa un control de calidad basado en criterios definidos que validan coherencia, precisión y alineación temática, detectando inconsistencias y generando reportes sobre tendencias, alertas y recomendaciones que aseguren datos útiles y confiables para la toma de decisiones.

3. Solución del problema

3.1. Arquitectura de la solución

La arquitectura de la solución desde el punto de vista funcional está basada en módulos. Cada módulo está compuesto por uno o más contenedores basados en Docker. Esta solución utiliza tecnologías modernas y permite correr la solución on-premise o en cualquier proveedor de nube basado en contenedores. Esta arquitectura permite escalar de manera independiente cada uno de los módulos y/o de sus diferentes componentes de manera independiente. Entre la mayoría de las componentes principales se implementan colas de mensajería, lo que permite realizar comunicación asíncrona y procesamiento basado en eventos. Los diferentes módulos se describen a continuación. Desde el punto de vista de tecnología la arquitectura de la solución contendría los siguientes elementos:

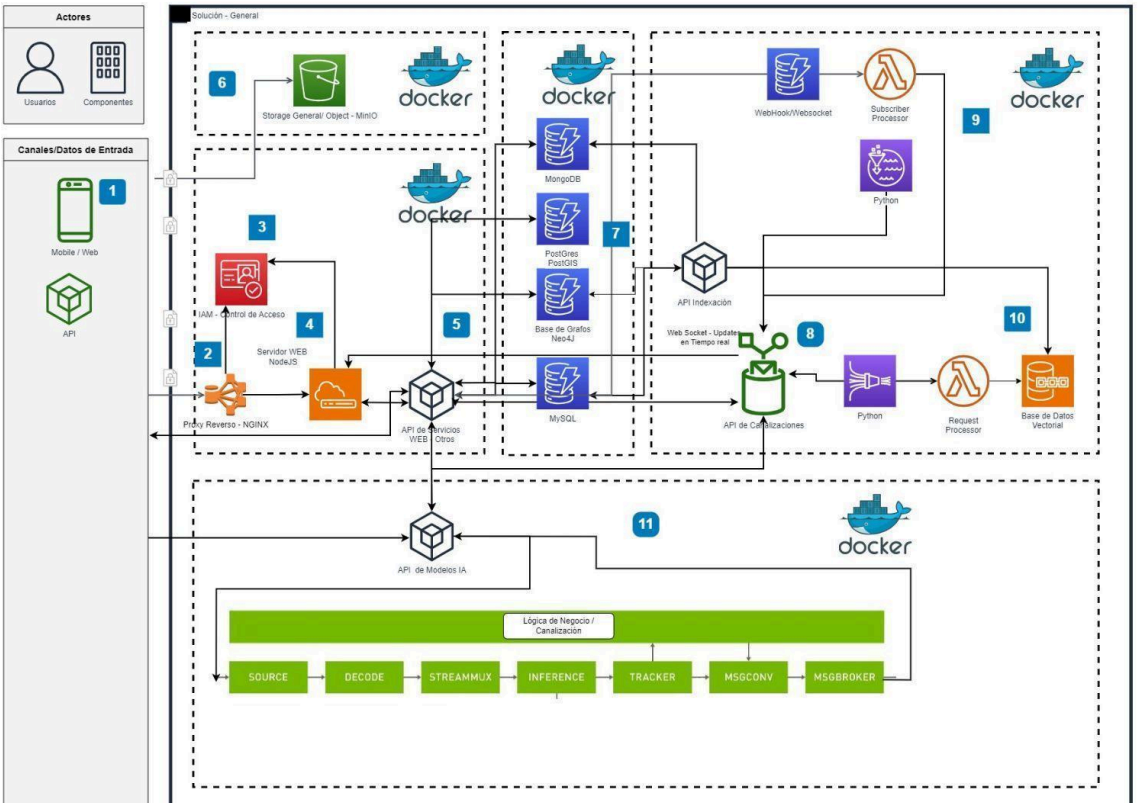


Imagen 1. Diagrama de la arquitectura propuesta. Se anexa la imagen a este documento.

Módulo Front-End y Capa Media:

1. Acceso por un servidor web local que tiene un proxy reverso basado en Apache Nginx. Todos los otros dispositivos se comunican por medio de una API y/o un componente del contenedor del servidor web y/o del contenedor que tiene el servicio del Colas de mensajería abre una conexión

remota.

2. Proxy reverso que permite escalar varios de los componentes de la solución planteada utilizando Apache Nginx. Control de Acceso a las diferentes aplicaciones / OAuth2 - JWT.
3. Los datos de autenticación estarán en la base de datos propia de KeyCloak.
4. Servicio WEB que tiene una aplicación basada en node.js que tendrá todos los casos de uso como autenticación, un CRUD básico de procesos de carga y transformación de archivos basado en formato PDF y/o EXCEL. También se implementarán todos los otros casos de uso necesarios para las 3 áreas de la Secretaría Distrital de Planeación y todos sus diferentes roles.
5. API de servicios implementada usando SpringBoot y java para atender todos los servicios de capa media necesarios por el servidor web, extraer los datos desde algún otro componente de la solución en caso de necesidad.

Módulo Almacenamiento:

6. Servicios de almacenamiento de objetos basado en un protocolo compatible con S3 y/o similar. Se usó la herramienta de código abierto MinIO. En caso de que en el futuro se desee migrar a una solución de nube, se puede cambiar este servicio sin problema. Este servicio permite la replicación.

Módulo de Bases de Datos:

7. Servicios de base de datos. La solución contempla una base de datos no estructurada basada en MongoDB. También se contempla una base de datos postgres para persistir información de tipo geo-espacial (extensión postgis) y cualquier otro tipo de información estructurada y relacional necesaria como MySQL. También se contempla una base de datos basada en grafos como Neo4J o similares.

Módulo de Indexación de Documentos:

8. Servicio de API basada en protocolo REST que permite escalar todo el proceso de indexación con uno o más nodos respectivamente para realizar todos los trabajos de indexación de los diferentes documentos y/o casos de uso en batch. Este servicio es fundamental para implementar toda la lógica de negocio de los diferentes pipelines y/o canalizaciones de procesamiento de modelos de Inteligencia Artificial.
9. Componentes para poder hacer procesamiento. Este componente permitirá realizar búsquedas híbridas. Se fundamenta en la implementación de uno o más requerimientos y/o pasos en python de todas las canalizaciones de datos. Este componente permitirá notificar eventos de interés a la API del backend.
10. En esta sección se tiene el servicio de la base de datos vectorial basado en una tecnología como MongoDB y/o similar. Este motor se encargará de persistir los sumergimientos (embeddings - por su nombre en inglés). Esto le permitirá a la solución realizar búsquedas semánticas, implementar varios temas de similitud semántica, entre otros. Este servicio está disponible por medio de la API visualizada en el contenedor y/o conectado directamente a las colas de mensajería

Módulo de Modelos de Inteligencia Artificial:

11. Contenedor Docker que implementa una solución basada en la tecnología de nvidia, python y otros servicios para exponer uno o más modelos de Inteligencia Artificial de una manera sincrónica y/o asincrónica. Este módulo cuenta con un servicio de cola de mensajería para hacer replicación y/o atender solicitudes en batch de manera asíncrona. Servicios de canalización y flujos de trabajo para encadenar uno o más modelos en un flujo de trabajo con capacidad para escalar cada componente del módulo de manera independiente. Integración con mensajería basadas en MQTT y/o KAFKA en caso de necesidad.
12. Contenedor Docker dedicado a correr triton-server y/o tensro_rt y/o servidor de inferencia con tecnologías similares, para exponer modelos de inteligencia artificial, procesamiento de lenguaje natural en múltiples formatos como ONNX y pytorch

3.2. Stack Tecnológico

Los elementos tecnológicos que se consideran en la solución actual de la SDP, y que aparecen en el diagrama anterior serían los siguientes:

Categoría	**Tecnología(s) empleadas**	**Justificación** (Explica en una o dos frases por qué se eligió esa tecnología, destacando su alineación con los objetivos del proyecto y ventajas técnicas o funcionales)
Framework Web	Angular y/o Vue (Nuxtjs)	Frameworks para implementación de frontend modernos. Hay muchos componentes necesarios para acelerar el desarrollo de una aplicación web responsiva y con buenos principios de diseño visual.
Lenguaje de Programación	TypeScript, Java, Python y/o C, C++	typeScript es un lenguaje moderno que permite implementar frontend y backend acelerando los tiempos de implementación. Java es un lenguaje empresarial seguro y con muchas librerías para diferentes usos en el backend. Python es el lenguaje por excelencia para implementar temas de Inteligencia Artificial. C y C++ se usan para lograr el mejor desempeño para realizar inferencia de modelos profundos.
Base de Datos	Mongo, Postgres, Neo4J, MySQL	Mongo es una base de datos no relacional que permite modelos de documentos y sus diversas relaciones. Hay múltiples drivers para su uso en todos los lenguajes. Postgres es una base de datos relacional con varias extensiones entre las que se encuentran PostGIS (Datos Geográficos). Tiene la opción de almacenar documentos en formato JSON (java script object notation – por su nombre en inglés). Neo4J es una base de datos para persistir grafos que pueden representar bases de conocimiento. ElasticSearch es un servidor que permite almacenar documentos

		en texto crudo permitiendo indexación y búsqueda basada en palabras clave con oportunidad de integración con servicios de búsqueda semántica y basada en vectores. Milvus es uno de los motores de base de datos vectoriales más utilizados en la actualidad. Puede operar en nube u on-premise. Implementa el almacenamiento de vectores que representan información de datos no estructurados con una oportunidad de escalar a billones de vectores en caso de necesidad
Modelos de IA/NLP	Modelos Entrenados por Inversiones Gutiérrez Garcia en reconocimiento de entidades nombradas, clasificación basada en vectores, extracción de relaciones, preguntas extractivas y generativas y modelos de lenguaje basados en transformadores como BERT, NOMIC (varias de sus variantes), Gliner, Modelos de Lenguaje de Gran Escala basados en Llama 3 y sus variantes.	Modelos de procesamiento de lenguaje que implementan el estado del arte en tareas como reconocimiento de entidades nombradas, extracción de relaciones, desambiguación de entidades, entre otros. Estas tareas son fundamentales para la solución implementada. Modelos de clasificación basados en surgimientos y/o embeddings (por su nombre en inglés). Estos vectores representan características de una palabra, frase, párrafo o documento según sea necesario. Implementación personalizada basada en afinamiento teniendo ejemplos de textos y documentos en Colombia. Modelos de preguntas extractivas y generativas fundamentales para resolver parte de los retos identificados. Modelos de Lenguaje de gran escala afinados para seguir instrucciones en data sets customizados para retos similares.
Otra(s)	Apache Camel	Tecnologías modernas que implementan protocolo de suscripción y producción. Permiten implementar comunicación asíncrona entre microservicios. Kafka Connect tiene una gran cantidad de conectores de fuente y destino para implementar cualquier comunicación entre diferentes fuentes de datos. Redis implementa cachés en memoria para acelerar procesamiento basado en eventos y también permite suscripción y producción asíncrona. Apache Camel es una librería para realizar integraciones e implementación de ETL's (extracción, transformación y Carga, por sus siglas en inglés)

3.3. Procesamiento y gestión de datos

El flujo de información inicia con la carga de un archivo en formato PDF o EXCEL en el que se encuentre la información en la capa web. Seguidamente se guarda el archivo en el componente de almacenamiento (contenedor Docker – MinIO). Luego se le notifica al usuario quien puede generar el evento para iniciar el procesamiento del archivo. Este procesamiento se implementa en un flujo en Apache Camel que tiene la funcionalidad de procesar los archivos en EXCEL, validar su estructura y aplicar una o más reglas de negocio según corresponda para los diferentes tipos de formatos. Luego de que las validaciones de los diferentes

archivos se cumplen, se persiste en batch en la base de datos de Mongo principalmente. Se implementó un proceso en batch asociado al Mongo para extraer, analizar y procesar los eventos de inserción, actualización y eliminación de una o más de las colecciones que representan los diferentes modelos de documentos propuestos en la solución. Se utiliza un conector especial para este fin dentro del componente de uno o varios scripts en python. Este conector persiste en memoria los diferentes eventos en formato JSON (Java Script Object Notation). Cada paso de las diferentes canalizaciones tiene implementada una lógica de negocio para uno o más flujos de trabajo de manera respectiva.

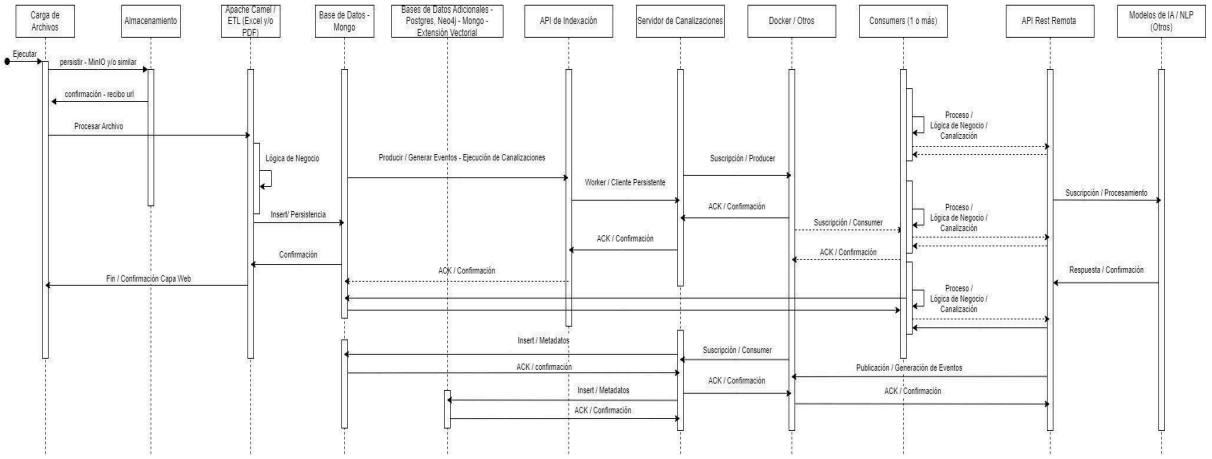
Algunos ejemplos de estos flujos de trabajo corresponden a realizar la generación de "chunks" o "fragmentación de textos" (en caso de que aplique), generación de surgimientos utilizando uno o más modelos de procesamiento de lenguaje natural, extracción de entidades nombradas, extracción de relaciones, extracción de preguntas generativas y/o extractivas, clasificación de textos, construcción de una o más bases de conocimiento, entre otras tareas. Para poder independizar las inferencias de todos los modelos, nuevamente se utilizan colas de mensajería para poder realizar el mejor procesamiento en batch para cada uno de los diferentes modelos.

Al terminar todas las inferencias, se generan las respuestas y/o eventos respectivos en otra cola. Posterior y nuevamente utilizando todos los componentes de la arquitectura de Kafka Connect, se consumen dichos eventos y se persisten en una o más de las colecciones finales con todos los metadatos necesarios para dar solución a todos los retos de las diferentes áreas de la Secretaría de Planeación. En caso de necesidad, algunos de los conectores también persistirán parte de los datos en el servidor de Postgres y/o Neo4J y/o Mongo. Todos estos servicios serán consumidos de manera posterior cuando el usuario realice la visualización de los diferentes análisis implementados en los diferentes flujos de trabajo.

3.4. Flujo de procesamiento

El flujo de información inicia con la carga de un archivo en formato PDF o EXCEL en el que se encuentre la información en la capa web. Seguidamente se guarda el archivo en el componente de almacenamiento (contenedor Docker – MinIO). Luego se le notifica al usuario quien puede generar el evento para iniciar el procesamiento del archivo. Este procesamiento se implementa en un flujo en Apache Camel que tiene la funcionalidad de procesar los archivos en EXCEL, validar su estructura y aplicar una o más reglas de negocio según corresponda para los diferentes tipos de formatos.

A continuación, se presenta un diagrama de secuencia simplificado que ilustra el flujo de procesamiento general de los datos en la solución.



Luego de que las validaciones de los diferentes archivos se cumplen, se persiste en batch en la base de datos de Mongo principalmente. Se implementará el patrón de captura de Datos (Change Data Capture) asociado al log binario de la base de datos en Mongo para extraer, analizar y procesar los eventos de inserción, actualización y eliminación de una o más de las colecciones que representan los diferentes modelos de documentos propuestos en la solución. Se utiliza un conector especial para este fin dentro del componente de Kafka Connect. Este conector persiste en memoria en un servidor Kafka y/o un servidor RabbitMQ y/o otra cola de mensajería los diferentes eventos en formato JSON (Java Script Object Notation). Posteriormente estos diferentes eventos presentes en una o más colas diferentes, serán consumidos por uno o más clientes que continuarán el flujo de procesamiento.

Cada uno de esos consumidores tienen implementada una lógica de negocio para uno o más flujos de trabajo de manera respectiva. Algunos ejemplos de estos flujos de trabajo corresponden a realizar la generación de "chunks" o "fragmentación de textos" (en caso de que aplique), generación de sumergimientos utilizando uno o más modelos de procesamiento de lenguaje natural, extracción de entidades nombradas, extracción de relaciones, extracción de preguntas generativas y/o extractivas, clasificación de textos, construcción de una o más bases de conocimiento, entre otras tareas.

Para poder independizar las inferencias de todos los modelos, nuevamente se utilizan colas de mensajería para poder realizar el mejor procesamiento en batch para cada uno de los diferentes modelos. Al terminar todas las inferencias, se generan las respuestas y/o eventos respectivos en otra cola. Posteriormente y nuevamente utilizando todos los componentes de la arquitectura de Kafka Connect, se consumen dichos eventos y se persisten en una o más de las colecciones finales con todos los metadatos necesarios para dar solución a todos los retos de las diferentes áreas de la Secretaría de Planeación. En caso de necesidad, algunos de los conectores también persistirán parte de los datos en el servidor de Postgres y/o Neo4J, ElasticSearch y/o Milvus. Todos estos servicios serán consumidos de manera posterior cuando el usuario realice la visualización de los diferentes análisis implementados en los diferentes flujos de trabajo.

El Mínimo Producto Viable, no tendrá integración con ningún sistema existente de la SDP. En algún momento se comentó la oportunidad de integrar el directorio activo de la entidad, pero claramente ese requerimiento excede el alcance del reto de innovación.

3.5. Seguridad con los datos

A nivel de seguridad de datos, el Prototipo Mínimo Viable que se implementará, utiliza el protocolo https/tls para cifrar las comunicaciones enviadas desde el cliente al servidor y viceversa. Se utiliza el cifrado desde el proxy reverso incluso para protocolos como WebSocket. Internamente la comunicación de los servicios no se cifra pues se entiende que hay una red privada virtual y un firewall perimetral que ayuda con la seguridad. Internamente las máquinas virtuales en las que se despliega la solución tendrán abiertos exclusivamente los puertos necesarios para su funcionamiento.

Se implementa el estándar Oauth2 y la implementación disponible en el servicio de IAM (Identity and Access Management) en la herramienta KeyCloak. Este servicio utiliza buenas prácticas para la persistencia de las contraseñas de los usuarios y dicho servicio está encargado de generar todos los tokens que se utilizan en la solución. A nivel de base de datos no se implementa ningún tipo de servicio de encripción. No hay políticas de back establecidas, pero dado que gran parte de la arquitectura de solución propuesta se encuentra implementada en tecnología de contenedores, se puede realizar backups con facilidad.

4. Metodología de trabajo

SCRUM

La metodología usada para el desarrollo fue SCRUM. Se realizan sprints de una (1) semana por el tiempo tan corto del proyecto. En la parte de modelos y cuando haya lugar a afinamientos y/o ajustes, se usará la metodología CRISP-.DM (Cross-Industry Standard Process for Data Mining, por su nombre en inglés).

Fases Principales y Actividades:

Análisis y Diseño:

- Definición de modelos de datos.
- Diseño de flujos de trabajo para procesar datos.

Implementación:

- Desarrollo del Front-End, integración de APIs y configuración de servicios de almacenamiento.
- Creación de canalizaciones de datos para procesamiento y transformación.

Pruebas y Validación:

- Pruebas de carga y rendimiento.
- Validación de criterios de aceptación con usuarios finales.

CRISP-DM (Cross-Industry Standard Process for Data Mining)

Esta metodología será utilizada en la parte de modelado y afinamiento de algoritmos de IA. Incluye:

- Comprensión de los Datos
- Preparación de Datos
    - Limpieza, etiquetado y transformación de conjuntos de datos.
- Modelado y Evaluación
    - Entrenamiento y validación de modelos con datos reales del dominio.
    - Ajuste de hiper parámetros para mejorar la precisión y el rendimiento.

Se realizarán reuniones de trabajo con las diferentes dependencias para resolver dudas puntuales y se realizarán reuniones semanales con representantes de las diferentes dependencias para ir mostrando el desarrollo de la solución y con ello evitar que se presenten inconvenientes que detengan la normal ejecución del proyecto y pongan en peligro la implementación de la misma, además que asegura que la solución cumple con las expectativas de los usuarios.

4.1. Cronograma

Actividad	**Mes 1**	**Mes 2**	**Mes 3**
General - Análisis y Diseño final de los Modelos de Datos.	X		
General - Implementación del Modelo de Datos – Pruebas de Escritorio	X		
General -Configuración y Aprovisionamiento de los Entornos de Desarrollo - Servicios	X		
General - Pruebas de Conectividad y/o Comunicación de Componentes	X	X	
Front-End - Desarrollo de Mockups	X		
Front-End - Implementación de todas las vistas y/o páginas de la solución	X		
Front-End - Integración con Api de Servicios / Api Gateway	X	X	
Front-End - Integraciones de Servicios Adicionales	X	X	
Front-End - Integración de Autenticación y/o Privilegios, Roles, entre otros temas de seguridad	X		
Front-End - Pruebas de Usuario / Aseguramiento de Calidad / Criterios de Aceptación de Componentes	X	X	X
Capa Media - Implementación del Api Gateway – Seguridad – Flujos de Comunicación necesarios	X	X	
Capa Media - Implementación de la API de Servicios Capa Web y Otros	X	X	X
Capa Media - Implementación de Integración con Otros Servicios	X	X	
Capa Media - Pruebas de Casos de Uso – Operaciones	X	X	X

Actividad	Mes 1	Mes 2	Mes 3
ETLs - Implementación de todos los flujos necesarios en Apache Camel para implementar lógica de negocio requerida para diferentes casos de uso.	X	X	X
ETLs - Pruebas de ETLs	X	X	X
ETLs - Pruebas de Carga Simple		X	X
Canalización de Datos - Análisis y Diseño de las diferentes canalizaciones de datos necesarias	X		
Canalización de Datos - Implementación de las diferentes canalizaciones de datos - conectores	X	X	X
Canalización de Datos - Pruebas de Flujos de Canalizaciones	X	X	X
Canalización de Datos - Pruebas de Carga		X	X
Modelos de Inteligencia Artificial - Etiquetado, Pruebas de Latencia, Criterios de Aceptación, Pruebas integrales según metodología CRISP-DM (Cross-Industry Standard Process for Data Mining, por su nombre en inglés)	X	X	
Modelos de Inteligencia Artificial -Construcción de Conjunto de Datos de Prueba	X		
Modelos de Inteligencia Artificial -Construcción de Conjunto de Datos de Evaluación	X		
Modelos de Inteligencia Artificial -Evaluación de Flujos de Trabajo	X	X	X
Modelos de Inteligencia Artificial -Pruebas de Carga		X	X
General - Pruebas Integrales		X	X
General - Validación de Criterios de Aceptación			X

4.2. Backlog de requerimientos

Dependencia	Perfil del Usuario	Necesidades específicas identificadas	Casos de uso
Subsecretaría de Políticas Públicas y Planeación Social y Económica Dirección de Formulación y Seguimiento de Políticas Públicas.	Especialista en políticas públicas / analista de datos.	La Subsecretaría busca una solución innovadora que permita: ● Clasificar los aportes en variables estratégicas alineadas con los objetivos del plan de desarrollo y los ODS. ● Automatizar la generación de contribuciones estratégicas. ● Corregir errores de ortografía. ● Detectar información incompleta. ● Eliminar duplicidades. ● Relacionar indicadores con resultados concretos. ● Identificar alertas, logros y áreas de mejora. ● Sugerir acciones para optimizar los resultados. ● Evidenciar el impacto de las políticas públicas.	● Login ● Inicio de sesión ● Visita de página inicial – reporte simplificado de tareas realizadas. ● Carga de Documentos, ● Visualización procesos de negocio – documentos cargados y procesados, ● Visualización de resultados.

Dependencia	Perfil del Usuario	Necesidades específicas identificadas	Casos de uso
		• Relacionar los resultados con actividades específicas. Todo esto mediante un sistema basado en analítica y tecnologías avanzadas para optimizar el análisis y el seguimiento de los objetivos estratégicos.	
Oficina de Participación y Diálogo de Ciudad	Facilitadores de participación ciudadana / gestores de información	La Oficina necesita un sistema que permita: • Mejorar la trazabilidad y el intercambio de datos relacionados con la participación ciudadana. • Ordenar, clasificar y agrupar los diferentes tipos de aportes según criterios definidos • Analizar dimensiones como relevancia estratégica, impacto, factibilidad, innovación y sostenibilidad. • Producir datos categorizados y medibles que faciliten el seguimiento ciudadano y su integración en áreas de trabajo específicas. • Permitir la visualización categorizada de la información. • Facilitar la apropiación de los datos por parte de los usuarios según sus áreas de interés. • Estas funcionalidades estarán respaldadas por análisis avanzados y algoritmos que evitarán sesgos, garantizando datos transparentes y útiles para la toma de decisiones estratégicas.	• Login • Inicio de sesión • Visita de página inicial – reporte simplificado de tareas realizadas. • Carga de Documentos, • Visualización procesos de negocio – documentos cargados y procesados, • Visualización de resultados. • Visualización de resultados detallados. • Exportar datos. • Cierre de sesión.
Subsecretaría de Planeación de la Inversión Dirección Distrital de Programación, Seguimiento a la Inversión y Plan de Desarrollo	Planificadores de inversión / analista de datos	La Subsecretaría de Planeación de la Inversión requiere una solución basada en inteligencia artificial para analizar y clasificar la información recolectada por las entidades distritales según temáticas estratégicas. El sistema deberá: • Identificar tendencias clave en la información y clasificarlas de manera estratégica. • Garantizar la calidad de los datos mediante controles automatizados que validen coherencia, precisión y alineación temática. • Detectar y reportar inconsistencias, generando alertas y recomendaciones para la toma de decisiones. • Producir reportes detallados sobre tendencias, alertas de incoherencia y acciones sugeridas.	• Login • Inicio de sesión • Visita de página inicial – reporte simplificado de tareas realizadas. • Carga de Documentos, • Visualización procesos de negocio – documentos cargados y procesados, • Visualización de resultados. • Visualización de resultados detallados. • Exportar datos • Cierre de sesión.

Dependencia	Perfil del Usuario	Necesidades específicas identificadas	Casos de uso
		La solución integrará análisis de datos cualitativos, categorización estratégica y herramientas de visualización, facilitando la evaluación de propuestas ciudadanas con base en criterios de relevancia estratégica, impacto social y factibilidad. Esto garantizará información confiable y útil para una toma de decisiones informada y efectiva en Bogotá.	

4.3. Matriz de riesgos técnicos

Este es el conjunto de riesgos que se han presentado o se podrían presentar en la solución una vez se realice el despliegue de la solución para el SDP, se deben considerar todos los elementos que se requieran para mantener la operación arriba.

RAG	DESCRIPCIÓN RIESGO O INCONVENIENTE	MITIGACIÓN	IMPACTO
	No iniciar en las fechas establecidas las entrevistas con los usuarios no nos permitirá avanzar con el entendimiento de la problemática y podría afectar el proceso de la solución	Se Acepta el hecho que los funcionarios claves no estuviesen presentes, se trajeron a la mesa personas con capacidades similares	HIGH
	No disponer de la infraestructura solicitada en la Nube	Se Acepta el impacto por parte del proveedor y dispuso de infraestructura a su propio costo para no detener el normal desarrollo de las actividades	VERY HIGH
	La infraestructura dispuesta no permita que se realice la comunicación correctamente entre sus diferentes elementos de la Arquitectura	Realizar el escalamiento del caso, para lograr la atención y resolución de los inconvenientes en el menor tiempo posible.	HIGH
	El proveedor de la nube no disponga de planes de acción en caso de fallas en su propia infraestructura que impacte a la SDP	Contar con Backups y acciones de despliegue rápido en otra nube.	HIGH
	Es necesaria la participación activa en las pruebas y el feedback de los usuarios finales es importante para la calidad del producto	Mantener la comunicación abierta con las diferentes áreas, y enviar el mensaje adecuado para evitar el rechazo al cambio que podría derivar en el fracaso de la implementación	HIGH

4.4. Plan de pruebas de usuario

- En la certificación de los usuarios finales se considera tener una tabla con la lista de las pruebas que se harán por cada Pipeline del proceso que se defina con los usuarios finales.
- De cada prueba los usuarios finales de las pruebas detalladas deben realizar videos que muestren

cómo se debe proceder y cómo los procesos están enlazados al Excel que tiene catalogadas todas las pruebas.
- En reuniones conjuntas con cada secretaría se deben revisar las observaciones y se tienen que identificar las acciones a tomar.

5. Anexos

- Anexo 1_Cronograma detallado.

https://docs.google.com/spreadsheets/d/1zN-yE77U23NCfHhMjPxw0oZ445i96CPd/edit?usp=drive_link&ouid=104920492087535799495&rtpof=true&sd=true

- Anexo 2_Diagrama de la arquitectura propuesta

https://drive.google.com/file/d/1dux4yxF4Sy4yb5KWS29N9bzpPgvhse4b/view?usp=drive_link

6. Glosario

A

- API (Interfaz de Programación de Aplicaciones): Conjunto de reglas y protocolos que permiten que diferentes aplicaciones se comuniquen entre sí.
- Apache Camel: Framework de integración que facilita la implementación de patrones de enrutamiento y manejo de mensajes entre sistemas.
- Apache Kafka: Plataforma de mensajería distribuida que permite procesar flujos de datos en tiempo real.
- Autenticación OAuth2 - JWT: Mecanismo de autenticación basado en tokens seguros para gestionar el acceso a aplicaciones y servicios web.

B

- Backend: Capa media de una aplicación web, en la que se implementan operaciones a nivel de lógica de negocio y se implementan integraciones con servicios internos y/o externos.
- Base de Datos NoSQL: Tipo de base de datos diseñada para manejar grandes volúmenes de datos no estructurados, como MongoDB y Neo4J.
- Bases de Datos Vectoriales: Motores especializados en almacenar y recuperar datos en formato vectorial, optimizados para búsquedas semánticas.
- Batch: Proceso que se ejecuta de manera asíncrona y en el que se realizan operaciones sobre un conjunto de ítems de manera conjunta.

C

- Canalización de Datos: Flujo de transformación y/o ejecución de una o más operaciones sobre un conjunto de datos para lograr uno o más objetivos relevantes a nivel de lógica de negocio.

- Contenedores Docker: Tecnología que permite empaquetar aplicaciones y sus dependencias en entornos portátiles y escalables.
- Cross-Industry Standard Process for Data Mining (CRISP-DM): Metodología estándar para el desarrollo de proyectos de minería de datos.

D

- Data Pipeline (Canalización de Datos): Secuencia de procesos que transforman y mueven datos de una fuente a un destino.
- Desambiguación de Entidades: Técnica de inteligencia artificial para identificar y diferenciar entidades con nombres similares en un texto.
- Docker: Tecnología propietaria para implementar el paradigma de contenedores a nivel de múltiples sistemas operativos. Esta tecnología permite realizar una abstracción a nivel de dependencias de un producto de software, para que se garantice un mínimo nivel de requisitos necesarios sin afectar el sistema operativo en el que se ejecuta uno o más de los contenedores mencionados.

E

- ETL (Extracción, Transformación y Carga): Proceso de integración de datos que implica extraer información de diferentes fuentes, transformarla y cargarla en un sistema de destino.
- Embeddings (Sumergimientos vectoriales): Representación matemática de palabras, frases o documentos en un espacio vectorial para facilitar la búsqueda y el análisis semántico.

F

- Framework: Conjunto de herramientas y librerías que facilitan el desarrollo de software, como Angular y SpringBoot.
- Frontend: Capa de interacción con los usuarios funcionales a nivel de una aplicación web. En esta capa, los usuarios pueden hacer flujos de negocio y/o utilizar uno o más casos de uso para lograr un fin importante en un dominio de negocio particular.

I

- IAM (Identity and Access Management): Sistema de gestión de identidades y accesos que permite administrar de manera segura el acceso de usuarios a sistemas, aplicaciones y datos. IAM garantiza que solo los usuarios autorizados puedan acceder a ciertos recursos, aplicando políticas de autenticación y control de permisos.
- Inferencia en Inteligencia Artificial: Proceso mediante el cual un modelo de IA hace predicciones o clasificaciones con base en datos de entrada.

K

- KeyCloak: Solución de código abierto para la gestión de identidad y acceso (IAM) que permite autenticación, autorización basada en roles y federación de identidades. Soporta protocolos estándar

como OAuth2, OpenID Connect y SAML, permitiendo la integración con múltiples aplicaciones y servicios.

M

- Microservicios: Arquitectura de software en la que una aplicación se compone de pequeños servicios independientes que se comunican entre sí.
- MinIO: Plataforma de almacenamiento de objetos compatible con el protocolo S3.
- Mockup: Diseño esquemático de una interfaz de usuario que permite identificar los requerimientos a nivel visual y a nivel de interacción para lograr un objetivo a nivel de negocio.

N

- Nginx: Servidor web de alto rendimiento utilizado como proxy reverso y balanceador de carga.
- Neo4J: Base de datos orientada a grafos que permite almacenar y analizar relaciones complejas entre datos.

O

- On-premise: Infraestructura de hardware que corresponde a máquinas en un centro de datos que es gestionado por alguien. No es un servicio en nube.

P

- PostGIS: Extensión de PostgreSQL que permite el almacenamiento y procesamiento de datos espaciales.
- Procesamiento de Lenguaje Natural (PLN): Rama de la inteligencia artificial que permite a las computadoras comprender, interpretar y generar texto humano.
- Proxy Reverso: Servidor que distribuye el tráfico entre múltiples servidores backend para mejorar la eficiencia y seguridad.

R

- Reconocimiento de Entidades Nombradas (NER): Técnica de PLN utilizada para identificar nombres de personas, organizaciones, ubicaciones y otros elementos clave en un texto.
- Redis: Sistema de almacenamiento en memoria utilizado para el almacenamiento en caché y la mensajería en tiempo real.

S

- Scrum: Marco de trabajo ágil para la gestión de proyectos que se centra en entregas incrementales y colaboración en equipos pequeños. Su objetivo es mejorar la eficiencia y adaptabilidad al facilitar la entrega de valor continuo al cliente.
- SpringBoot: Framework de Java para el desarrollo de aplicaciones backend con microservicios.
- Stack Tecnológico: Conjunto de herramientas, frameworks y tecnologías utilizadas para desarrollar

una solución de software.

T

- Triton Server: Servidor de inferencia desarrollado por NVIDIA que permite ejecutar modelos de IA optimizados para rendimiento.
- TypeScript: Lenguaje de programación basado en JavaScript que mejora la seguridad y escalabilidad del código.

7. Referencias

- Apache Software Foundation. (s.f.). Apache Kafka Documentation. Recuperado el 18 de marzo de 2025, de https://kafka.apache.org/documentation/
- Apache Software Foundation. (s.f.). Apache Camel Documentation. Recuperado el 18 de marzo de 2025, de https://camel.apache.org/manual/
- Docker Inc. (s.f.). Docker Overview. Recuperado el 18 de marzo de 2025, de https://docs.docker.com/get-started/overview/
- ISO/IEC. (2021). Information technology — Artificial intelligence — Overview of Natural Language Processing (NLP) technologies. ISO/IEC TR 24027:2021.
- Microsoft. (s.f.). *What is Identity and Access Management (IAM)?*. Recuperado el 19 de marzo de 2025, de https://learn.microsoft.com/en-us/security/zero-trust/identity-access-management
- MongoDB Inc. (s.f.). MongoDB Documentation. Recuperado el 18 de marzo de 2025, de https://www.mongodb.com/docs/manual/
- NVIDIA Corporation. (s.f.). Triton Inference Server Documentation. Recuperado el 18 de marzo de 2025, de https://docs.nvidia.com/deeplearning/triton-inference-server/
- Object Management Group (OMG). (s.f.). Unified Modeling Language (UML) Specification. Recuperado el 18 de marzo de 2025, de https://www.omg.org/spec/UML/
- PostgreSQL Global Development Group. (s.f.). PostgreSQL Documentation. Recuperado el 18 de marzo de 2025, de https://www.postgresql.org/docs/
- Red Hat. (s.f.). Spring Boot Documentation. Recuperado el 18 de marzo de 2025, de https://spring.io/projects/spring-boot
- Red Hat. (s.f.). *Keycloak Documentation*. Recuperado el 19 de marzo de 2025, de https://www.keycloak.org/documentation
- World Wide Web Consortium (W3C). (s.f.). OAuth 2.0 Authorization Framework. Recuperado el 18 de marzo de 2025, de https://datatracker.ietf.org/doc/html/rfc6749