

Gestión inteligente de insumos:

Diseño de un agente de IA aplicado a la gestión
de stock en una clínica privada



Alumna: Rosario Assis

Mail: rosarioassis12@gmail.com

Tutor: Marcelo Medina Galván

Año 2025



Índice

Resumen	3
Introducción	4
Situación Problemática	5
Preguntas de Investigación	5
Objetivo General	5
Objetivos Específicos	6
Marco Metodológico	6
Marco Teórico	7
Aplicación.....	15
a. Descripción del proceso actual de seguimiento, consolidación y control de stock	15
b. Análisis de las características del proceso actual	19
c. Requerimientos para la elaboración del agente de Inteligencia Artificial	20
d. Diseño del primer prototipo de agente de IA	20
e. Diseño del prototipo final del agente de IA.....	22
e.1.Flujo de automatización en n8n para consolidación de datos	24
e.2.Configuración de herramienta MCP Server	27
e.3.Creación del agente de IA conectado mediante chat	29
Recomendaciones.....	31
Conclusiones	32
Apéndice	33
Referencias.....	42

Resumen

La gestión de insumos es un elemento crítico en las instituciones de salud que ofrecen tratamientos de alta complejidad. En este sentido, contar con materiales disponibles de manera oportuna y confiable asegura la continuidad de los tratamientos y contribuye a mantener la calidad de la atención, evitando demoras y dificultades en la planificación de las actividades clínicas.

El presente trabajo tiene como objeto de estudio al “Centro HD”, un centro privado de hemodiálisis de la provincia de Tucumán. El objetivo general del estudio consiste diseñar un agente de inteligencia artificial capaz de automatizar la integración, consolidación y actualización de los datos de inventario en el Centro HD. Entre los objetivos específicos se incluyen: describir el proceso actual de gestión de inventario, identificar los factores que inciden en su eficiencia, y caracterizar el entorno organizacional actual, así como sus limitaciones para la implementación del agente.

El estudio se desarrolla bajo un enfoque metodológico mixto, empleando un diseño de caso único con intervención tecnológica. La recolección de información combina observación de los procesos administrativos y análisis de los registros de consumo y compras de insumos.

Con base en esta evidencia, se implementa una solución tecnológica compuesta por tres elementos: un flujo de automatización en n8n para procesar diariamente los registros enviados desde Farmacia, una base de datos relacional en Supabase que reemplaza al archivo de Excel usado originalmente y un agente de inteligencia artificial conectado mediante chat, capaz de responder consultas en lenguaje natural a través del uso de herramientas estructuradas que ejecutan consultas sobre la base de datos.

Los resultados muestran que la automatización parcial del proceso mejora significativamente el ordenamiento del flujo de datos, reduce tiempos de consolidación, disminuye errores derivados de la intervención manual y aumenta la disponibilidad inmediata de información confiable. Asimismo, el análisis revela la importancia de acompañar la implementación tecnológica con prácticas organizacionales orientadas a la estandarización del registro y a la adopción progresiva de nuevas herramientas por parte del personal. En conjunto, la solución desarrollada sienta las bases para avanzar hacia un modelo de gestión más eficiente, basado en datos y escalable, alineado con las necesidades operativas del Centro HD.

Palabras claves: gestión de insumos, inteligencia artificial, automatización, agente inteligente.

Introducción

Según la Encuesta Nacional de Nutrición y Salud (ENNyS), realizada en 2019, el 12,7% de la población argentina padece algún grado de enfermedad renal crónica (ERC), lo que equivale aproximadamente a uno de cada ocho habitantes. A pesar de esta alta prevalencia, la mayoría de los afectados desconoce su condición hasta que la enfermedad alcanza estadios avanzados. Actualmente, se estima que unas 30.000 personas en el país requieren diálisis, mientras que otras 15.000 se encuentran en seguimiento para un trasplante renal. En la provincia de Tucumán, según datos del Centro Único Coordinador de Ablación e Implante de Tucumán (CUCAITUC) correspondientes al año 2025, existen alrededor de 1.255 pacientes en tratamiento de diálisis y 99 personas en lista de espera para un trasplante renal.

En este contexto, este trabajo tiene como objeto de estudio al **“Centro HD”**, un centro de hemodiálisis de la provincia de Tucumán que, desde hace más de veinte años, se dedica a la atención integral de pacientes con enfermedades renales. La clínica se especializa en brindar tratamientos de hemodiálisis, control médico y acompañamiento profesional, constituyéndose en un referente provincial en la materia.

La motivación de este estudio radica en la importancia de la gestión eficiente de insumos para asegurar la continuidad de los tratamientos de hemodiálisis. Dado que se trata de una práctica vital que los pacientes deben recibir varias veces por semana, cualquier interrupción en la provisión de materiales y recursos podría afectar directamente la calidad del servicio y la seguridad de los pacientes. Por ello, comprender y mejorar los procesos de abastecimiento se vuelve un aspecto esencial en la administración de este tipo de instituciones sanitarias.

Situación Problemática

La gestión del inventario de insumos en el Centro HD constituye una actividad fundamental para garantizar la continuidad de los tratamientos de hemodiálisis. En la práctica, el área de Farmacia registra diariamente la recepción de insumo provenientes de compras y los consumos realizados por los distintos sectores. Esta información se envía posteriormente al área de Administración, que se encarga de consolidarla en un archivo maestro y realizar su análisis.

Actualmente, los datos de consumos y compras se manejan de manera fragmentada en distintos registros y archivos, y la consolidación depende completamente del trabajo manual del personal administrativo. Como consecuencia de esto, se observan retrasos en la actualización de la información, dificultad para conocer con precisión los niveles de stock y necesidad de verificaciones constantes por parte del personal. Estas condiciones dificultan la planificación oportuna de reposiciones y limitan la capacidad de anticipar faltantes o excesos de insumos.

La situación afecta principalmente al personal de Farmacia y Administración, así como a la operación general del Centro, ya que la disponibilidad organizada y actualizada de insumos es esencial para la planificación y eficiencia de los tratamientos de hemodiálisis.

Preguntas de Investigación

1. ¿Cómo se lleva a cabo actualmente el proceso de gestión y consolidación de datos de inventario de insumos en la clínica?
2. ¿Qué aspectos del proceso actual de gestión de inventario están relacionados con la eficiencia y la calidad de la información?
3. ¿Qué características debe tener un agente de inteligencia artificial para automatizar la integración y actualización de los datos de inventario?
4. ¿Cuáles son las características del entorno organizacional actual y sus limitaciones para la implementación de un agente de inteligencia artificial en la gestión de insumos?

Objetivo General

Diseñar un agente de inteligencia artificial que automatice la integración, consolidación y actualización continua de los datos relacionados con el inventario de insumos en la clínica privada objeto de estudio, mejorando la disponibilidad, precisión y oportunidad de la información, contribuyendo así a una gestión más eficiente y fundamentada en datos confiables para la toma de decisiones estratégicas.

Objetivos Específicos

1. Describir el proceso actual de integración, consolidación y actualización de datos de inventario.
2. Identificar y caracterizar factores de proceso actual que podrían incidir en la eficiencia y calidad de la gestión del inventario.
3. Definir las características y funcionalidades necesarias para el diseño de un agente de inteligencia artificial orientado a automatizar la integración y actualización de datos de inventario.
4. Caracterizar el entorno organizacional actual e identificar las limitaciones que presenta para la implementación de un agente de inteligencia artificial en la gestión de insumos.

Marco Metodológico

El proyecto se realiza utilizando un enfoque mixto, combinando técnicas cualitativas y cuantitativas para abordar tanto la comprensión del proceso actual de gestión de inventarios como el análisis de datos necesarios para el desarrollo del agente de inteligencia artificial.

Los métodos mixtos o híbridos representan un conjunto de procesos sistemáticos, empíricos y críticos de investigación, e implican la recolección y el análisis de datos tanto cuantitativos como cualitativos, así como su integración y discusión conjunta para realizar inferencias producto de toda la información recabada (denominadas metainferencias) y lograr un mayor entendimiento del fenómeno bajo estudio (Hernández-Sampieri & Mendoza, 2008). En la ruta mixta, se utiliza evidencia de datos numéricos, verbales, textuales, visuales, simbólicos y de otras clases para comprender problemas en las ciencias (DeCuir-Gunby & Schutz, 2017; Creswell, 2013a; Lieber & Weisner, 2010). Chen (2006) señala que los métodos híbridos pueden presentarse de manera pura, conservando las estructuras originales de los métodos cuantitativo y cualitativo, o modificada, adaptando y sintetizando los métodos según las necesidades del estudio. Asimismo, su implementación puede seguir distintas secuencias: cuantitativo primero y cualitativo después, cualitativo primero y cuantitativo después, simultáneamente o fusionados a lo largo de todo el proceso de investigación.

El estudio de caso, por su parte, constituye un diseño de investigación que permite examinar un fenómeno dentro de su contexto real, especialmente cuando los límites entre fenómeno y entorno no son claramente distinguibles. Según Hernández-Sampieri, este enfoque posibilita analizar estructuras, procesos y comportamientos de manera integral, integrando diversas fuentes de información como observación, entrevistas y revisión documental, favoreciendo la comprensión de situaciones complejas y la generación de conocimiento contextualizado.

Se utiliza un diseño de estudio de caso único con intervención tecnológica, centrado en una clínica privada, lo que permite un análisis detallado y la aplicación directa de la solución. La recolección de datos incluye la observación de los procesos administrativos, entrevistas semiestructuradas con el personal involucrado y revisión documental de los archivos digitales



existentes, en los cuales se registran los datos de consumo y compras de insumos. Para el análisis, se emplean técnicas de modelado y procesamiento de datos para estructurar la información, junto con el desarrollo del agente inteligente que automatice la integración y actualización de los datos.

Marco Teórico

El estudio de procesos administrativos y de tecnologías aplicadas a la gestión de insumos en organizaciones de salud requiere apoyarse en diversos marcos conceptuales. Estos fundamentos permiten comprender cómo se estructuran, se analizan y se optimizan los procesos, cómo se gestionan los recursos críticos, qué rol cumplen los datos en la toma de decisiones y bajo qué principios operan los sistemas de información y las tecnologías emergentes como la automatización y la inteligencia artificial. A continuación, se desarrollan los conceptos teóricos esenciales que sustentan esta investigación:

Procesos administrativos. Cursosgramas

En primer lugar, para comprender los fundamentos que sustentan el diseño y análisis de sistemas operativos dentro de organizaciones de salud, resulta necesario abordar la noción de proceso.

Un proceso es un conjunto de actividades interrelacionadas o que interactúan entre sí, destinadas a transformar entradas en salidas y a generar resultados previstos de manera eficaz y eficiente (ISO 9001:2015). La gestión adecuada de los procesos es esencial para garantizar la calidad, eficiencia y consistencia en la operación de cualquier organización, incluyendo instituciones de salud y clínicas privadas.

Características de los procesos según la norma ISO 9001 2015:

- **Entradas y salidas definidas:** cada proceso debe contar con insumos claros que se transforman en productos o resultados específicos.
- **Secuencia e interacción:** las actividades deben organizarse de manera lógica y conectarse con otros procesos relacionados.
- **Criterios y métodos de control:** establecer mecanismos que aseguren la efectividad y eficiencia del proceso.
- **Recursos necesarios:** identificar y disponer de los recursos humanos, tecnológicos y materiales requeridos.
- **Responsabilidades y autoridades:** definir quién es responsable de cada actividad y su alcance.



- **Evaluación y mejora continua:** implementar mecanismos para supervisar, medir y optimizar el proceso.

En este contexto, los cursogramas constituyen herramientas gráficas que permiten visualizar la secuencia de actividades, decisiones y flujos de información dentro de un proceso (Harrington, 1991). Permiten identificar redundancias, cuellos de botella y oportunidades de mejora en la operación organizacional.

Según Betancourt (2016) los cursogramas facilitan la comprensión de los procesos al mostrar cómo se relacionan las distintas actividades y recursos, permitiendo planificar intervenciones más eficientes. Klein (2016) distingue entre cursogramas sinópticos, que ofrecen una visión general del proceso, y analíticos, que detallan cada actividad, decisión y control, siendo útiles para estandarizar procedimientos y mejorar la eficiencia.

Gestión de suministros médicos

Tras comprender los conceptos de procesos y su representación, es posible avanzar hacia un área clave en organizaciones sanitarias: la gestión de insumos y abastecimiento. La gestión de suministros médicos comprende la planificación, adquisición, almacenamiento, distribución y uso racional de medicamentos e insumos esenciales.

Según Management Sciences for Health (2002), un sistema integral de suministros garantiza disponibilidad, calidad y eficiencia en la atención, optimizando recursos y reduciendo costos.

El ciclo de gestión incluye:

- **Selección:** consiste en identificar los insumos y medicamentos necesarios según las necesidades de salud de la población y los niveles de atención. Una selección adecuada permite optimizar recursos, mejorar la eficiencia del proceso de adquisición y garantizar la disponibilidad de productos esenciales.
- **Adquisición:** comprende la cuantificación de necesidades, la elección de modalidades de compra, la negociación con proveedores y la evaluación de calidad de los insumos. La eficiencia en esta etapa impacta directamente en los costos y en la continuidad del servicio.
- **Almacenamiento:** busca preservar la calidad de los insumos durante su permanencia en depósito o farmacia, asegurando custodia, vigilancia y control de condiciones ambientales y administrativas.
- **Distribución:** incluye la entrega de insumos a los distintos puntos de atención, garantizando seguridad, trazabilidad y oportunidad en la provisión.
- **Uso racional:** abarca diagnóstico, prescripción, dispensación y consumo adecuado de los insumos, asegurando que se empleen de manera eficiente y segura.

La integración de todas las funciones permite reducir desabastecimientos, minimizar costos y mejorar la calidad de la atención.

Gráfico 1¹: Esquema de un modelo integral de sistema de suministro de medicamentos esenciales.



Fuente: Guía Práctica para la Planificación de la Gestión del Suministro de Insumos Estratégicos. Organización Panamericana de la Salud

Ciencia de datos

Una vez comprendida la lógica e importancia de los procesos de gestión de suministros, surge la necesidad de interpretar la información generada por esas operaciones. Aquí adquiere relevancia la ciencia de datos. Esta disciplina se define como la extracción de conocimiento accionable a partir de datos (Provost & Fawcett, 2013). Comprende:

1. **Procesamiento y limpieza de datos:** garantizar que la información sea consistente, organizada y libre de errores, asegurando la validez de los análisis.
2. **Análisis y modelado:** aplicación de técnicas estadísticas, minería de datos y algoritmos de aprendizaje automático para descubrir patrones, relaciones y generar predicciones.
3. **Interpretación y comunicación:** traducción de los resultados en conclusiones claras y accionables, que faciliten la comprensión y la aplicación de la información.

Entre sus características principales destacan:

- Interdisciplinaria: combina estadística, informática, matemáticas y conocimiento del área de aplicación.
- Basada en evidencia: las conclusiones derivan de datos verificables y análisis rigurosos.
- Orientada a la acción: los hallazgos deben poder aplicarse a la resolución de problemas concretos.



- Iterativa: requiere ciclos continuos de prueba, evaluación y refinamiento para mejorar la calidad de los resultados.

Sistemas de bases de datos

Dado que la ciencia de datos requiere estructuras sólidas para almacenar y organizar información, resulta pertinente introducir los fundamentos de las bases de datos. El modelo relacional, propuesto por Edgar F. Codd (1970), establece que los datos deben organizarse en tablas relacionadas mediante valores comunes, permitiendo operaciones lógicas y garantizando consistencia. Codd afirma que “las relaciones entre datos pueden representarse mediante tablas formadas por filas y columnas, permitiendo manipulación mediante operadores matemáticos y lógicos”.

Elmasri y Navathe (2017) definen una base de datos como “una colección de datos relacionados, diseñada para ser compartida por múltiples usuarios y aplicaciones, minimizando redundancia e inconsistencias” (*Fundamentos de Sistemas de Bases de Datos*, Pearson). Entre los elementos estructurales del modelo relacional se destacan:

- **Clave primaria (Primary Key):** atributo que identifica de manera única cada registro.
- **Clave foránea (Foreign Key):** atributo que referencia a una clave primaria en otra tabla, asegurando integridad referencial.
- **Integridad de datos:** reglas que garantizan consistencia, ausencia de duplicados y validez de valores.

Software de código abierto

En este tipo de infraestructuras de datos, los sistemas abiertos se vuelven especialmente relevantes. El software de código abierto se caracteriza por permitir que el código fuente sea estudiado, modificado y redistribuido libremente. La Open Source Initiative define el código abierto como un modelo que garantiza acceso al código fuente, posibilitando su uso, modificación y distribución sin restricciones propietarias (OSI, 2007). Esta definición constituye el marco formal que orienta la práctica y la regulación del movimiento open source.

Desde una perspectiva teórica, el autor Richard Stallman sostiene que este tipo de software otorga a los usuarios un control real sobre las herramientas que utilizan, reduce la dependencia de proveedores únicos y fomenta la mejora continua a través de comunidades activas de desarrollo (Stallman, 2002).

PostgreSQL como sistema de gestión de bases de datos

PostgreSQL es un sistema de gestión de bases de datos relacional de código abierto ampliamente reconocido por su estabilidad, solidez y adhesión a los estándares del modelo relacional. Su documentación oficial lo describe como “un sistema objeto-relacional con integridad transaccional completa, amplio soporte para SQL y un alto nivel de extensibilidad”

(PostgreSQL Global Development Group, 2024). Desde sus desarrollos iniciales, PostgreSQL fue concebido para manejar grandes volúmenes de información y permitir reglas automáticas que faciliten la gestión de datos.

Entre sus características principales se destacan:

- **Soporte para triggers y funciones almacenadas**, que permiten ejecutar acciones automáticas ante la inserción, modificación o eliminación de registros, reforzando la consistencia de la base de datos.
- **Integridad referencial**, que se sostiene a través de claves primarias y claves foráneas, asegurando que las relaciones entre tablas sean válidas y estén correctamente vinculadas.
- **Escalabilidad y alto rendimiento**, lo que permite que la base de datos crezca sin comprometer la velocidad o la estabilidad.
- **Extensibilidad**, dado que permite incorporar nuevos tipos de datos, operadores o funciones según las necesidades del sistema.

Inteligencia Artificial

Tras definir cómo se gestionan los datos, es posible abordar la disciplina que permite interpretarlos y operar sobre ellos de manera autónoma: la Inteligencia Artificial.

La IA es una rama de la informática dedicada al diseño y estudio de agentes inteligentes, entendidos como entidades capaces de percibir su entorno y tomar decisiones que maximicen sus posibilidades de éxito (Russell & Norvig, 2020). En este sentido, busca desarrollar sistemas que puedan razonar, aprender y actuar de manera autónoma frente a problemas complejos.

Russell y Norvig (2020) identifican cuatro enfoques principales de la IA, combinando las dimensiones de comportamiento vs. pensamiento y humano vs. racional:

1. **Enfoque de actuar humanamente:** Este enfoque se centra en crear sistemas que reproduzcan el comportamiento humano observable. Se basa en la idea de que un agente es inteligente si sus acciones son indistinguibles de las de un ser humano, como se evalúa en el Test de Turing (Russell & Norvig, 2020). El objetivo no es necesariamente entender los procesos internos del pensamiento humano, sino generar un comportamiento convincente desde la perspectiva externa.
2. **Enfoque de pensar humanamente:** En este caso, la IA intenta replicar los procesos cognitivos humanos. Los investigadores estudian cómo piensan y razonan las personas mediante introspección, experimentos psicológicos y técnicas de neuroimagen, para luego traducir esos modelos a programas de computadora (Russell & Norvig, 2020). La meta es que el sistema no solo actúe como un humano, sino que también piense de manera análoga a los procesos mentales humanos.

3. **Enfoque de pensar racionalmente:** Se centra en la lógica y las leyes del pensamiento correcto. Inspirado en la tradición logicista, este enfoque busca que los agentes lleguen a conclusiones válidas a partir de premisas correctas, utilizando razonamiento formal y principios matemáticos (Russell & Norvig, 2020). La inteligencia se define aquí como la capacidad de inferir de manera correcta, independientemente de la imitación del comportamiento humano.
4. **Enfoque de actuar racionalmente:** En este enfoque, la inteligencia de un agente se mide por su capacidad para tomar las mejores decisiones posibles dadas las circunstancias. El agente racional persigue objetivos específicos, se adapta a cambios, percibe el entorno y actúa para maximizar el éxito esperado (Russell & Norvig, 2020). Este paradigma, conocido como el modelo estándar de agentes racionales, es el más utilizado en la investigación y aplicación práctica de la IA.

Automatización de Procesos con Inteligencia Artificial

A partir de estas capacidades, emergen modelos híbridos que combinan IA con técnicas de automatización, especialmente RPA. La combinación de Inteligencia Artificial (IA) y Robotic Process Automation (RPA) ha transformado significativamente la forma en que se gestionan los procesos administrativos, permitiendo que tareas que tradicionalmente requerían intervención humana puedan ejecutarse de manera autónoma y eficiente (Willcocks & Lacity, 2016). Mientras que la RPA se centra en la automatización de tareas estructuradas, repetitivas y de bajo nivel, como el ingreso de datos o la ejecución de procedimientos rutinarios, la IA agrega capacidades de comprensión, análisis y toma de decisiones, posibilitando que los sistemas identifiquen patrones, clasifiquen información y realicen predicciones de manera autónoma.

Asimismo, el uso de tecnologías complementarias, como OCR (Reconocimiento Óptico de Caracteres) y NLP (Procesamiento de Lenguaje Natural), permite extraer información de documentos no estructurados y convertirla en datos procesables por los sistemas automatizados, ampliando el alcance de la automatización más allá de los procesos estrictamente repetitivos.

Entre los principales beneficios de esta integración se destacan el aumento de la eficiencia operativa, la reducción de errores asociados a la intervención manual y la mejora en la trazabilidad y control de los procesos administrativos. En conjunto, estas herramientas representan un avance hacia la gestión inteligente de procesos, donde la tecnología no solo ejecuta tareas, sino que también apoya la toma de decisiones y la optimización de los flujos de trabajo.

Plataformas de automatización de procesos

Para implementar dicha automatización, se emplean plataformas capaces de modelar, ejecutar y supervisar flujos de trabajo. Su función principal es reemplazar tareas manuales repetitivas por procedimientos automáticos, asegurando mayor eficiencia, uniformidad y control en la operación.

Según Dumas, La Rosa, Mendling y Reijers (2018), estas plataformas integran tres capacidades fundamentales:

1. Modelar procesos en un lenguaje estructurado,
2. Automatizarlos mediante reglas y componentes reutilizables, y
3. Supervisar su desempeño para facilitar la mejora continua.

Agentes inteligentes

El concepto de agentes permite conectar los principios anteriores con sistemas capaces de actuar autónomamente. Un agente es cualquier entidad capaz de percibir su entorno mediante sensores y actuar sobre él utilizando actuadores (Russell & Norvig, 2020). La percepción consiste en recibir información del entorno en cualquier instante, y la acción se refiere a la respuesta del agente frente a esas percepciones. El comportamiento de un agente está determinado por su función del agente, que proyecta cada secuencia de percepciones en una acción específica. Esta función puede implementarse mediante un programa del agente, ejecutado sobre la arquitectura del sistema, diferenciándose así de la descripción abstracta que representa la función del agente.

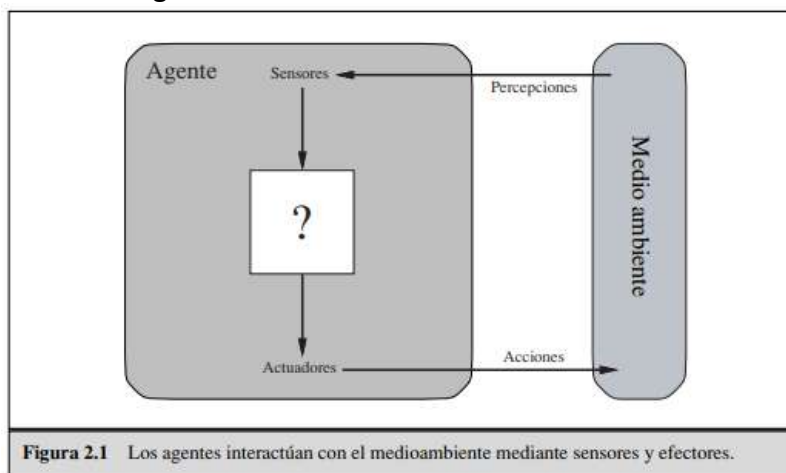


Figura 2.1 Los agentes interactúan con el medioambiente mediante sensores y efectores.

Fuente: Russell, S. J., & Norvig, P. (2020). Artificial Intelligence: A Modern Approach (4th ed.). Pearson.

Un agente inteligente típicamente posee las siguientes características (Russell & Norvig, 2020):

- Autonomía: Operan sin supervisión constante, gestionando sus acciones y estados internos.
- Percepción: Reciben información del entorno para tomar decisiones informadas.
- Reactividad: Responden rápidamente a cambios o estímulos externos.
- Proactividad: Toman iniciativas para cumplir sus objetivos.
- Capacidad de aprendizaje: Mejoran su comportamiento con la experiencia.

Estos agentes pueden clasificarse según su complejidad y arquitectura en:

- Agentes simples de reflejo: Operan bajo un paradigma estímulo-respuesta, sin modelos internos complejos
- Agentes basados en modelos: Poseen una representación interna del entorno. Permiten planificar acciones futuras en función del conocimiento del contexto y son los más adecuados para entornos parcialmente observables.
- Agentes basados en objetivos: Poseen metas explícitas que guían su toma de decisiones y utilizan técnicas de planificación para seleccionar acciones que maximicen la probabilidad de alcanzar sus objetivos.
- Agentes basados en utilidad: Evalúan las posibles acciones según una función de utilidad que refleja preferencias y riesgos. Son capaces de tomar decisiones óptimas en situaciones con múltiples alternativas y resultados inciertos.

Gestión del cambio organizacional

Finalmente, toda incorporación tecnológica requiere un abordaje sistemático de la gestión del cambio para su adopción efectiva. Según Chiavenato (2011), el cambio organizacional implica “modificaciones en personas, estructuras, procesos o tecnologías”, y su implementación exitosa depende de la capacidad de la institución para aprender y adaptarse. En este sentido, el cambio no se limita a introducir una herramienta nueva, sino que requiere transformar prácticas, creencias y rutinas existentes.

Robbins y Judge (2017) destacan que la adopción tecnológica suele enfrentar múltiples fuentes de resistencia, entre ellas los hábitos arraigados, el temor a la obsolescencia profesional, la incertidumbre sobre los resultados y la comodidad asociada a los procedimientos tradicionales. Estas resistencias pueden manifestarse de manera individual o estructural, y deben ser anticipadas mediante estrategias de comunicación, participación y capacitación.

Por su parte, Kotter (1996), sostiene que la adopción de nuevas prácticas requiere generar sentido de urgencia, construir una coalición que impulse el cambio, comunicar de manera clara la visión y eliminar obstáculos que dificulten su implementación. Estas etapas permiten que



los equipos comprendan el propósito de la innovación, disminuyen la incertidumbre y fortalecen la aceptación.

En conjunto, estos aportes subrayan que la introducción de tecnologías no depende solo del componente técnico, sino de la capacidad organizacional para acompañar el proceso con alineamiento, comunicación y aprendizaje colectivo.

Aplicación

a. Descripción del proceso actual de seguimiento, consolidación y control de stock

A continuación, se describe el proceso mediante el cual el Centro HD gestiona y realiza el seguimiento de los insumos utilizados en sus operaciones. Esta descripción se basa en la observación directa de las tareas diarias y el análisis de los registros empleados para controlar los materiales.

Diariamente, el proceso inicia en el área de Farmacia, donde se registran los movimientos correspondientes a los consumos que se realizan en los tres turnos del día (esto debido a que la preparación de los insumos para los mismos se realiza el día anterior). Para ello, se utiliza un archivo denominado **“Control de Stock”**, organizado en tres hojas principales:

Hoja Stock: Contiene la tabla matriz del sistema de control de stock, la cual contiene 6 columnas

1. Código: Asigna un código a cada producto del stock
2. Rubro: Clasifica el producto de acuerdo a su rubro , el mismo puede ser:
 - ACCESOS VASCULARES
 - DESCARTABLES SALA DIÁLISIS
 - MEDICAMENTOS
 - DESCARTABLES LIMPIEZA Y ALIMENTOS
 - DESCARTABLES PLANTA DE AGUA
3. Producto: Contiene el nombre del producto
4. Compras: Calcula el total de compras por cada producto
5. Consumos: Calcula el total de consumos por cada producto
6. Stock: Realiza el cálculo de Compras- Consumos por cada insumo

Hoja “Consumos” (10 columnas):

1. Id: Identifica cada una de las filas de la hoja “consumos” con una clave única

2. Fecha: Indica la fecha en la cual se produce el consumo
3. Mes: Indica mediante formula TEXTO el mes al que hace referencia la celda “fecha”
4. Código: Indica el código asignado al insumo que se está dando de baja
5. Rubro: Indica el rubro al que pertenece el insumo, el mismo se asigna automáticamente al rellenar la celda código mediante una fórmula que hace referencia a la hoja “stock”.
6. Insumo: Indica el nombre del insumo que se consume , siguiendo la misma lógica que la celda “rubro”.
7. Cantidad.
8. Solicitante/retira: Indica quien es la persona que solicita ese insumo y que empleado del área farmacia es el que hace la entrega del mismo.
9. Sector: Indica a que sector va dirigido ese insumo (Por ejemplo: Sala de diálisis, recepción, limpieza).
10. Observaciones: Se detallan datos relevantes de ese movimiento (Por ejemplo: En el caso del consumo de antibióticos, resulta importante especificar a que paciente va dirigida su aplicación).

Hoja “Compras”(10 columnas):

1. Id: Identifica cada una de las filas de la hoja “compras” con una clave única
2. Fecha: Indica la fecha en la cual se produce el ingreso
3. Código: Indica el código asignado al insumo que se está dando de alta
4. Rubro: Indica el rubro al que pertenece el producto
5. Producto: Indica el nombre del producto que ingresa al stock
6. Cantidad
7. Numero de remito
8. Proveedor
9. Fecha de vencimiento
10. Observaciones: Se detalla datos relevantes de ese movimiento (Por ejemplo, quién recepcionó el insumo, quién controla la calidad del mismo, quien autorizó a recibir un insumo de características distintas a la orden de compra, etc.)

Una vez completado el registro diario, el archivo se envía por correo electrónico al personal del área de Administración, cuyo rol consiste en consolidar y organizar la información para el control y seguimiento de los insumos. Para ello, los datos de Farmacia se integran a un archivo de Excel llamado “Control de Stock Dinámico”, que mantiene la misma estructura de hojas y además contiene un dashboard que permite visualizar el consumo mensual por insumo.

Paralelamente, el personal de Administración mantiene en otro archivo llamado “Días de cobertura stock” un registro histórico mensual del consumo de cada insumo, que incluye una columna con el consumo estándar mensual de cada producto. Este registro permite organizar la información de manera que se puedan calcular promedios y analizar la evolución de los consumos, sirviendo de base para la planificación de próximos pedidos.

A partir de este registro histórico, se calcula el promedio de consumo mensual, teniendo en cuenta el consumo de los últimos dos meses para cada insumo. Luego, este promedio se divide entre 26, que representan los días en que el Centro HD trabaja en el mes (de lunes a sábado), obteniendo así el consumo diario promedio. Finalmente, se determina la cantidad de días de cobertura mediante la relación:

$$\text{Días de cobertura} = \text{Stock disponible} / \text{Consumo diario promedio}$$

Luego, se compara si los días de cobertura son suficientes para cubrir el consumo hasta la fecha estimada de llegada del pedido mensual, que generalmente se recibe el día 20 de cada mes:

- Si los días de cobertura alcanzan para cubrir hasta esa fecha, el proceso continúa con normalidad.
- Si los días de cobertura no son suficientes, se analiza la necesidad de realizar un pedido de urgencia para garantizar la disponibilidad de insumos hasta la llegada del pedido.

Para complementar la descripción realizada, se presenta el flujograma del proceso actual:

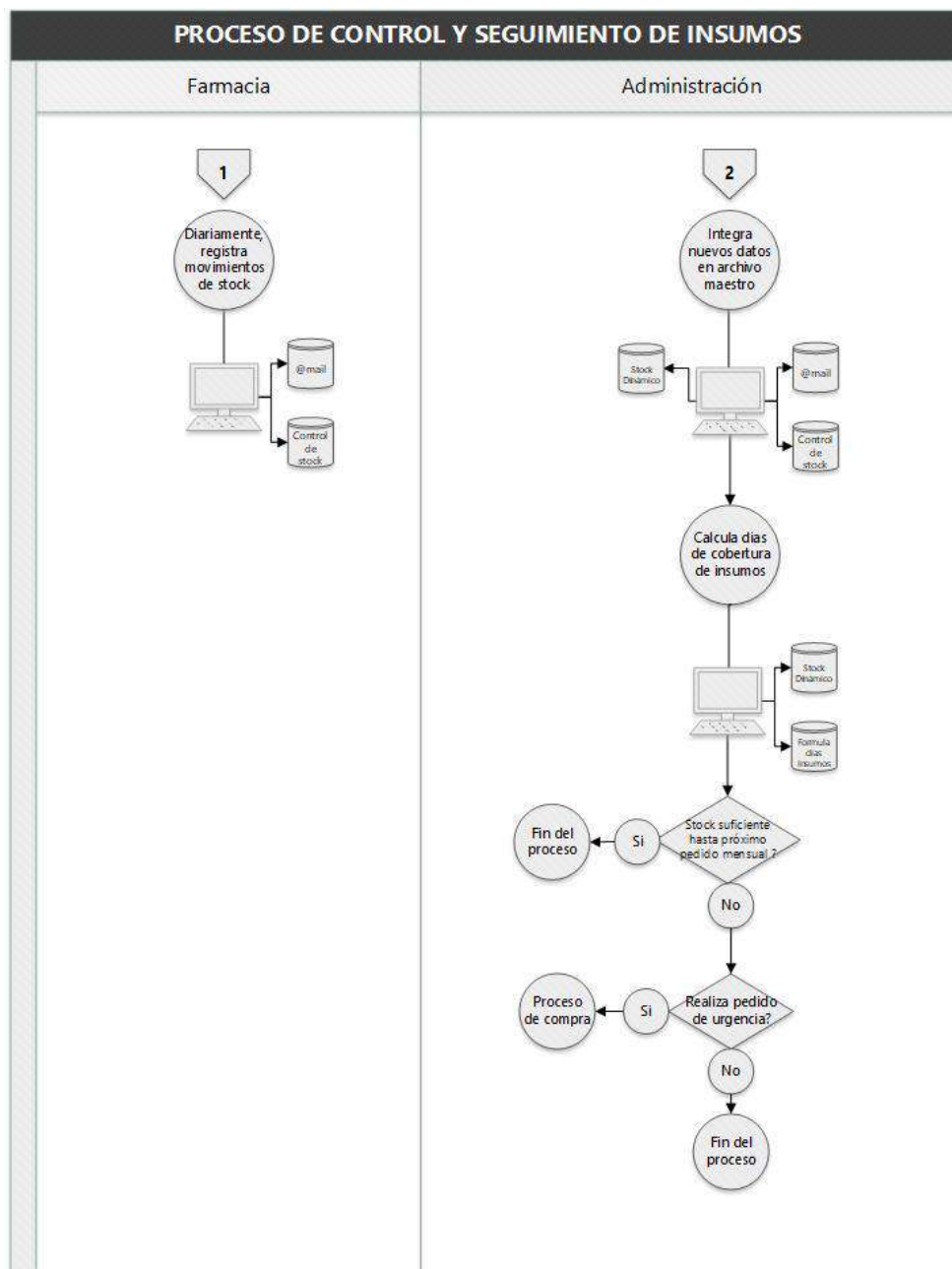


Gráfico N°1: Proceso de seguimiento y control de insumos

Fuente: Elaboración Propia

b. Análisis de las características del proceso actual

Este análisis del proceso vigente de registro, consolidación y actualización del inventario en el Centro HD permite identificar una serie de características que impactan directamente en la eficiencia del flujo de trabajo y en la calidad de los datos registrados:

1. Falta de una metodología estandarizada de registro:

El proceso de carga de movimientos de stock no se realiza siguiendo una secuencia formalizada. No existen protocolos que unifiquen criterios entre el personal. Esto genera variabilidad en la forma de registrar la información y ocasiona diferencias en tiempos, detalles y precisión.

2. Uso de registros manuales previos a la carga digital:

Antes de cargar los datos en el archivo de Excel, el personal de Farmacia anota los movimientos en un cuaderno o anotador. Este paso intermedio introduce riesgos adicionales, como errores de transcripción, omisiones, confusión entre anotaciones similares y pérdida de información si el anotador no está organizado.

3. Participación de múltiples personas en la carga final:

Con frecuencia, la persona que realiza el registro manual no es la misma que después transfiere esos datos al archivo de Excel. Esto implica interpretaciones subjetivas, confusiones al leer nombres o abreviaturas y discrepancias entre lo que se consumió y lo que finalmente se carga. La falta de un responsable único por turno o por tipo de movimiento afecta la trazabilidad y aumenta las inconsistencias.

4. Demora en la carga de consumos extraordinarios o “sorpresivos”:

Los consumos inesperados (por ejemplo, antibióticos o insumos utilizados en situaciones de urgencia) no siempre se registran en el momento en que ocurren. En muchos casos se anotan para cargar “más tarde”, lo que puede generar desactualización del stock teórico, diferencias con el stock físico y posterior necesidad de verificaciones manuales adicionales.

5. Ausencia de validaciones automáticas y controles de calidad del dato:

El sistema actual, basado en planillas, depende casi exclusivamente del criterio humano para asegurar la calidad de los datos. No existen mecanismos automáticos que validen:

- Códigos incorrectos o inexistentes.
- Cantidades negativas o improbables.

- Fechas inconsistentes.
- Duplicación de registros

c. Requerimientos para la elaboración del agente de Inteligencia Artificial

Teniendo en cuenta las limitaciones identificadas, se delinean las capacidades esenciales que debe incorporar un agente de inteligencia artificial para responder adecuadamente a las necesidades operativas del Centro HD. El diseño del agente se estructura considerando los principios propios de los agentes inteligentes (Russell & Norvig, 2020), y las buenas prácticas de gestión de inventarios en instituciones de salud.

1. Arquitectura general del agente: El agente debe operar bajo una arquitectura basada en modelos, capaz de mantener una representación interna del estado del inventario, interpretar entradas provenientes de distintos orígenes (archivos de farmacia, formularios, api u otros), y ejecutar acciones orientadas a mantener la información actualizada en tiempo real.

2. Funcionalidad de integración automática de datos: Se deberá unificar la información proveniente de registros de consumos y compras, para ello debe incorporarse mecanismos de lectura automatizada de archivos Excel o formularios digitales, capacidad para identificar nuevas filas o actualizaciones, emparejamiento por código de producto y fecha, detección de inconsistencias entre fuentes.

3. Validación y control de calidad del dato: Una de las funciones centrales será reducir el margen de error humano.

4. Actualización continua del stock: El agente debe permitir calcular consumos totales, compras totales, stock disponible y variaciones respecto a días anteriores.

5. Generación de reportes dinámicos: El agente deberá ser capaz de producir reportes automáticos tales como consumo por insumo y por período, tendencias mensuales y cálculo automático de días de cobertura,

d. Diseño del primer prototipo de agente de IA

Una vez definidos los requerimientos necesarios, se procede a desarrollar un primer prototipo del agente de IA. Esta primera versión se centra exclusivamente en la lectura e interpretación del archivo diario enviado por Farmacia, con el objetivo de validar la capacidad del agente para recibir el archivo, procesarlo y responder consultas sobre su contenido.

Para el desarrollo del agente se utiliza **n8n**, una plataforma de automatización de procesos de código abierto que permite integrar aplicaciones, servicios y herramientas de manera visual. Su funcionamiento se basa en la construcción de flujos de trabajo donde cada acción está

representada por un nodo, es decir, un bloque que cumple una función específica: recibir información, procesarla, transformarla o enviarla hacia otra aplicación. De esta manera, el agente se estructura como una secuencia de nodos interconectados que reproducen, de forma automatizada, tareas que habitualmente se realizan de manera manual.

En este esquema, la automatización se arma a partir de tres componentes principales:

1. **Nodo de entrada (Trigger Node):** Define el evento que inicia el flujo. Puede activarse por la llegada de un correo, la modificación de un archivo, una fecha programada o incluso una API externa.
2. **Nodos de procesamiento:** Ejecutan acciones intermedias dentro del flujo, tales como leer archivos Excel, extraer datos específicos, transformar la información a otros formatos, aplicar filtros, comparaciones o condicionales.
3. **Nodos de salida:** Se encargan de enviar el resultado procesado hacia otra aplicación o usuario, pudiendo responder consultas, cargar datos en otro archivo, actualizar bases de datos, reenviar reportes o alertas por correo o mensajería.

Otro elemento fundamental del funcionamiento de n8n son las **credenciales**, que consisten en permisos de acceso configurados dentro de la plataforma para conectarse de manera segura con aplicaciones externas (por ejemplo, correo electrónico, Google Drive, bases de datos o sistemas en la nube). Estas credenciales garantizan que el agente interactúe con los servicios requeridos sin exponer contraseñas ni datos sensibles en el flujo de trabajo.

Sobre la base de esta arquitectura, el primer prototipo incorpora un nodo de entrada vinculado al correo electrónico, configurado para activarse cada vez que llega un mensaje con el asunto "stock". Una vez detectado, el flujo identifica el archivo adjunto, lee sus diferentes hojas (consumos, compras y stock) y organiza la información. Con estos datos, el agente se conecta a un modelo avanzado de lenguaje natural desarrollado por Google, capaz de comprender preguntas sobre el archivo y generar respuestas contextualizadas.

Cabe destacar que, en esta primera versión, no se integra todavía el archivo "Control de Stock Dinámico". El alcance se centra únicamente en la lectura e interpretación del archivo enviado desde Farmacia, constituyendo un primer paso en el análisis de la viabilidad de la automatización.

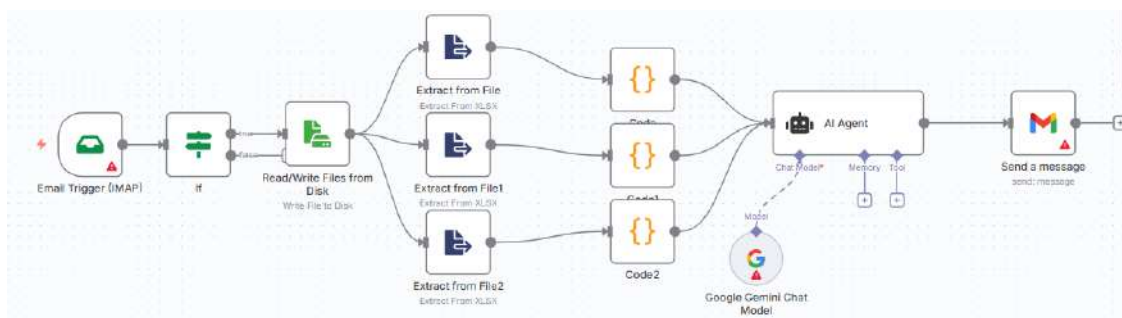


Gráfico n° 2: Primer prototipo de agente de IA para seguimiento y control de insumos
Fuente: Elaboración propia

e. Diseño del prototipo final del agente de IA

Luego de validar el primer prototipo, centrado en la lectura del archivo diario “Control de Stock”, se avanza hacia una segunda etapa orientada a automatizar la consolidación de datos en el archivo “Control de Stock Dinámico”, gestionado por el área de Administración. Este nuevo desarrollo busca eliminar la carga manual diaria de información e integrar automáticamente los registros generados en Farmacia al archivo histórico centralizado, manteniendo la integridad de las fórmulas existentes.

En esta fase, los objetivos principales se centran en automatizar la recepción del archivo enviado por Farmacia, extraer los datos de las hojas “Consumos” y “Compras”, actualizar el archivo dinámico incorporando únicamente los registros nuevos, resguardar copias de los archivos procesados para asegurar la trazabilidad y preservar las fórmulas originales del documento maestro.

Sin embargo, al diseñar este nuevo flujo de automatización surgen desafíos significativos que obligan a replantear la forma en que se organiza actualmente la información. El principal problema es el enorme volumen de datos acumulados dentro de los archivos de stock, que contienen registros desde mayo de 2023. Trabajar con archivos de Excel de gran tamaño genera demoras, bloqueos y fallas al intentar procesarlos mediante nodos de manipulación de planillas en n8n. Por este motivo, resulta inviable continuar utilizando exclusivamente archivos Excel como base del sistema automatizado.

Ante esta situación, y luego de analizar las herramientas compatibles con n8n, se decide transformar el archivo en una base de datos en Supabase. Esta plataforma permite crear y administrar bases de datos en la nube de forma sencilla, ofreciendo un entorno capaz de almacenar grandes volúmenes de información y conectarla fácilmente con otras aplicaciones. Supabase utiliza **PostgreSQL** como motor de base de datos, lo que garantiza estabilidad, rapidez en las consultas y una organización correcta de los datos. Además, la plataforma genera automáticamente los mecanismos necesarios para que sistemas externos puedan leer,

agregar o actualizar información sin necesidad de programar un servidor, facilitando así la integración con n8n.

Dentro de Supabase se crea un proyecto denominado “agente”, en el cual se establecen cuatro tablas principales que reemplazan la estructura del antiguo archivo “Control de Stock Dinámico” y también del archivo “Días de cobertura stock”. Las tablas creadas son: **stock**, **consumos**, **compras** y **días_cobertura_stock**. Las tres primeras reproducen la estructura del archivo original, manteniendo los mismos campos y organización, pero ahora bajo un esquema relacional que permite trabajar con mayor precisión y velocidad.

Para asegurar la coherencia entre las tablas y permitir que los campos “rubro” y “producto/insumo” se completen de manera automática cuando se cargan registros en “consumos” y “compras”, se define el campo “código” como **foreign key**. En una base de datos relacional, una foreign key (clave foránea) es un campo que referencia a un registro existente en otra tabla, garantizando que la información relacionada sea correcta y evitando ingresos de códigos inexistentes. Esto asegura que cada movimiento de consumo o compra esté vinculado a un producto válido de la tabla “stock”.

Asimismo, los campos “id” de las tablas “consumos” y “compras” se establecen como **primary key**. Una primary key (clave primaria) identifica de forma única cada registro dentro de una tabla y evita que existan dos registros con el mismo identificador. Esto permite mantener trazabilidad, evitar duplicados y facilitar la actualización selectiva de datos.

En la tabla “stock”, los campos “total_compras” y “total_consumos” se calculan de manera automática a través de un **trigger** y una función asociada. Un trigger es un mecanismo que se ejecuta de forma automática cuando ocurre un evento específico (por ejemplo, al insertar un nuevo consumo o compra). En este caso, cada vez que se agrega un registro en las tablas “compras” o “consumos”, el trigger actualiza las sumas acumuladas de cada producto y recalcula el campo “stock_actual”. De esta manera, el estado del inventario se mantiene sincronizado sin necesidad de realizar cálculos manuales o recurrir a fórmulas dentro de un archivo Excel.

La tabla “días_cobertura_insumos”, por su parte, almacena cálculos derivados del consumo histórico. Contiene registros de consumos mensuales por producto correspondientes al año 2025, así como columnas que calculan automáticamente el consumo promedio mensual tomando como referencia el promedio de los dos meses anteriores. También incluye una columna de consumo diario, obtenida al dividir el consumo mensual entre los 26 días operativos del Centro HD, y un campo que almacena el stock actualizado proveniente de la tabla “stock”. A partir de estos valores, la tabla permite calcular el indicador de días de cobertura y anticipar posibles faltantes de insumos, reproduciendo así de manera automatizada los cálculos que anteriormente debían realizarse manualmente desde el archivo Excel.

A fin de ilustrar la estructura y organización interna de las tablas que conforman la base de datos del agente, se presenta la visualización generada por Supabase. Esta representación permite observar de manera gráfica las relaciones entre las tablas, los campos principales definidos en cada una y la lógica relacional que sostiene el funcionamiento general del sistema.

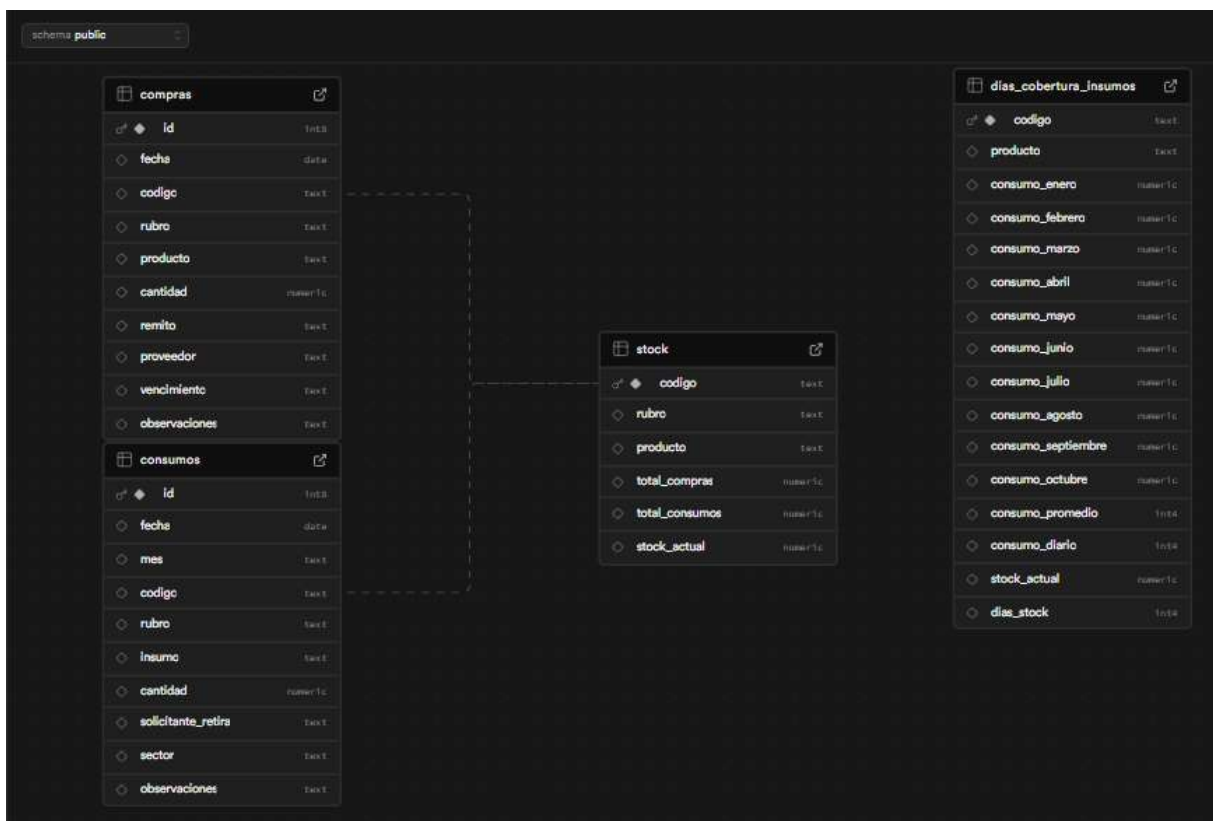


Gráfico n° 3: Estructura de la base de datos del Proyecto en Superbase
Fuente: Elaboración propia

Una vez creada la base de datos, es posible desarrollar en n8n el flujo de automatización destinado a consolidar los archivos diarios con la información almacenada en Supabase.

En esta etapa se diseña la arquitectura final del agente, organizada en tres componentes principales:

e.1.Flujo de automatización en n8n para consolidación de datos

Se construye un flujo que integra los archivos enviados diariamente desde Farmacia con la base de datos. El fin de este flujo es actualizar con los nuevos registros recibidos, la base de datos creada.

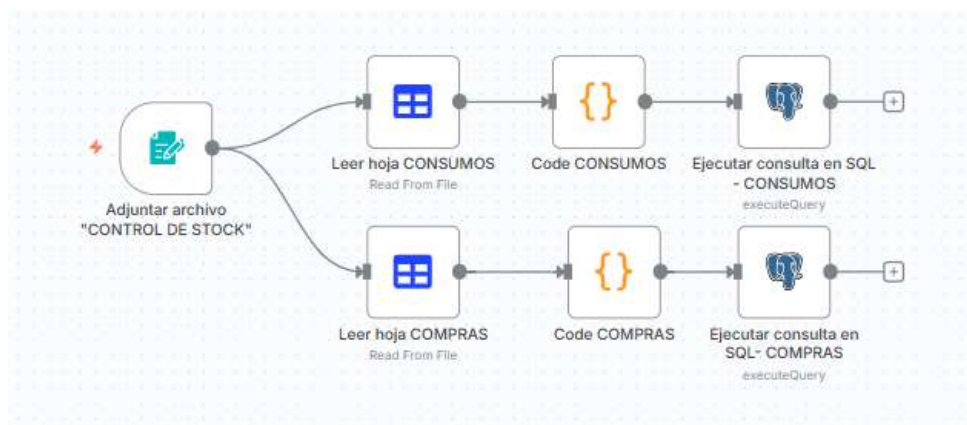


Gráfico N°4: Flujo de automatización para la integración de nuevos registros a la base de datos

Fuente: Elaboración propia

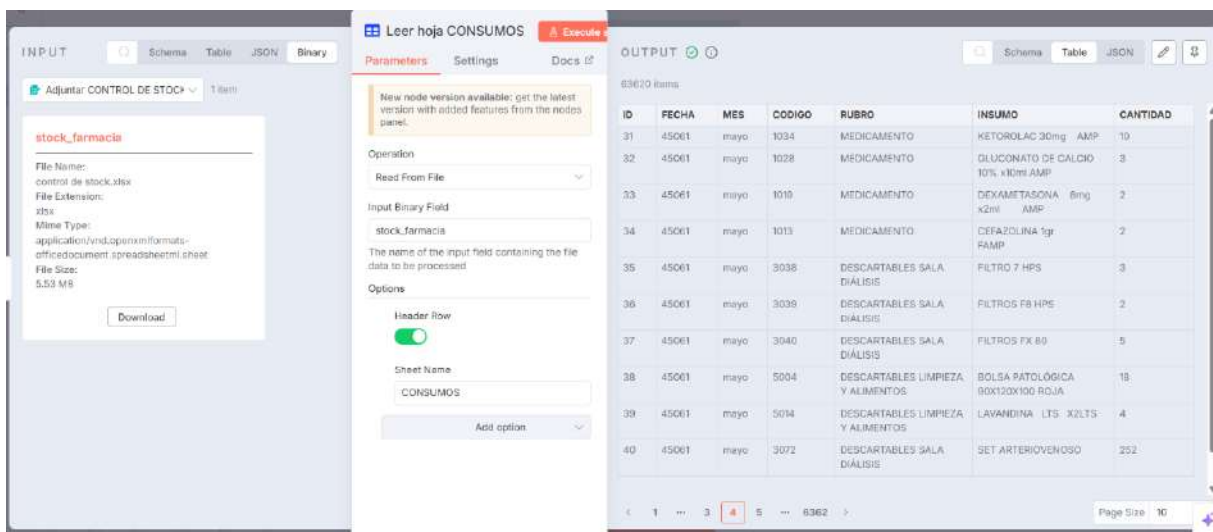
En primer lugar, se inserta un nodo de inicio de tipo **“Formulario”**, el cual permite que el usuario adjunte manualmente el archivo enviado desde el área de Farmacia. Este tipo de nodo actúa como un punto de entrada interactivo dentro de n8n: abre un formulario interno donde el usuario puede subir un archivo, escribir un texto o completar campos específicos antes de ejecutar el flujo. Se decide utilizar este nodo y no uno vinculado directamente al correo electrónico con el fin de evitar posibles retrasos o confusiones en la ejecución del flujo. Por ejemplo, si el archivo no llega con el asunto exacto configurado, o si existen variaciones en el modo en que se envía el correo, el flujo podría no activarse. En cambio, con el nodo Formulario se garantiza que el archivo se procese de manera segura, confiable y controlada, lo cual resulta especialmente importante durante la etapa inicial de implementación.



Gráfico N°5:Nodo inicial de tipo formulario para adjuntar control de stock

Fuente: Elaboración propia

A continuación, se insertan dos nodos de tipo **“Read from File”**. Este tipo de nodo permite leer el contenido de un archivo cargado previamente y extraer información desde hojas específicas en el caso de archivos Excel. En este flujo, cada nodo se configura para leer una hoja distinta del archivo: uno procesa la hoja **CONSUMOS** y el otro procesa la hoja **COMPRAS**. El resultado de esta lectura se convierte automáticamente al formato **JSON**, un formato estructurado y legible por n8n y esencial para permitir su procesamiento posterior.



The screenshot displays the n8n workflow editor. On the left, the 'INPUT' section shows the configuration for the 'Leer hoja CONSUMOS' node. The 'File Name' is 'control de stock.xlsx', the 'File Extension' is 'xlsx', and the 'File Size' is 5.53 MB. The 'Operation' is set to 'Read From File', and the 'Sheet Name' is 'CONSUMOS'. The 'Header Row' is checked. On the right, the 'OUTPUT' section shows a table with 63620 items. The table has columns: ID, FECHA, MES, CODIGO, RUBRO, INSUMO, and CANTIDAD. The data is organized into rows, with the first row showing ID 31, FECHA 45061, MES mayo, CODIGO 1034, RUBRO MEDICAMENTO, INSUMO KETOROLAC 30mg AMP, and CANTIDAD 10.

ID	FECHA	MES	CODIGO	RUBRO	INSUMO	CANTIDAD
31	45061	mayo	1034	MEDICAMENTO	KETOROLAC 30mg AMP	10
32	45061	mayo	1028	MEDICAMENTO	GLUCONATO DE CALCIO 10% x10ml AMP	3
33	45061	mayo	1010	MEDICAMENTO	DEXAMETASONA x2ml AMP	2
34	45061	mayo	1013	MEDICAMENTO	CEFAZOLINA 1gr FAMP	2
35	45061	mayo	3038	DESCARTABLES SALA DIALISIS	FILTRO 7 HPS	3
36	45061	mayo	3038	DESCARTABLES SALA DIALISIS	FILTROS FB HPS	2
37	45061	mayo	3040	DESCARTABLES SALA DIALISIS	FILTROS FX 80	5
38	45061	mayo	5004	DESCARTABLES LIMPIEZA Y ALIMENTOS	BOLSA PATOLÓGICA 80x120x100 ROJA	18
39	45061	mayo	5014	DESCARTABLES LIMPIEZA Y ALIMENTOS	LAVANDINA LTS X2LTS	4
40	45061	mayo	3072	DESCARTABLES SALA DIALISIS	SET ARTERIOVENOSO	252

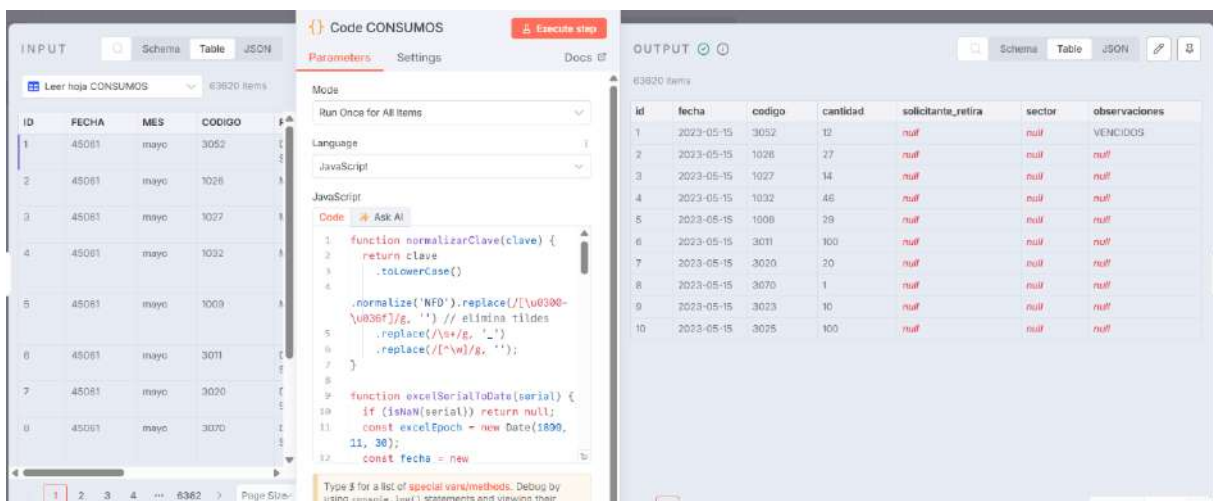
Gráfico N° 6: Output del nodo **“Read from file”** para hoja CONSUMOS

Fuente: Elaboración propia

Luego de esta lectura inicial, los datos son enviados a nodos de **código JavaScript**, un lenguaje de programación ampliamente utilizado dentro de n8n para transformar, limpiar y reorganizar información antes de integrarla a otros sistemas. Estos nodos permiten adaptar los registros al formato exacto que exige la base de datos, aplicando una serie de reglas y transformaciones específicas. Entre las principales modificaciones que se realizan se encuentran:

- Convertir las fechas al formato **Año-Mes-Día (YYYY-MM-DD)** requerido por la base.
- Estandarizar los nombres de los encabezados, dejando todos los campos en minúscula.
- Eliminar las columnas rubro y producto/insumo, dado que estos valores se completan automáticamente en Supabase a partir del código del producto mediante la relación establecida en la base de datos.

Estas transformaciones permiten garantizar que cada registro sea compatible con la estructura de Supabase y evitar errores al momento de insertarlo en las tablas correspondientes.



INPUT Schema Table JSON

Leer hoja CONSUMOS 63620 Items

ID	FECHA	MES	CODIGO
1	45081	mayo	3052
2	45081	mayo	1026
3	45081	mayo	1027
4	45081	mayo	1032
5	45081	mayo	1009
6	45081	mayo	3011
7	45081	mayo	3020
8	45081	mayo	3070

Code CONSUMOS Execute step

Parameters Settings Docs

Mode: Run Once for All Items

Language: JavaScript

JavaScript

```
Code Ask AI
1. function normalizarClave(clave) {
2.   return clave
3.     .toLowerCase()
4.     .normalize('NFD').replace(/[\u0300-
5.     \u036f]/g, '') // elimina tildes
6.     .replace(/ /g, '_')
7.     .replace(/[\^\w]/g, '');
8. }
9.
10. function excelSerialToDate(serial) {
11.   if (isNaN(serial)) return null;
12.   const excelEpoch = new Date(1899,
13.     11, 30);
14.   const fecha = new
```

OUTPUT Schema Table JSON

63620 Items

id	fecha	codigo	cantidad	solicitante_retira	sector	observaciones
1	2023-05-15	3052	12	null	null	VENCIDOS
2	2023-05-15	1026	27	null	null	null
3	2023-05-15	1027	14	null	null	null
4	2023-05-15	1032	46	null	null	null
5	2023-05-15	1009	29	null	null	null
6	2023-05-15	3011	100	null	null	null
7	2023-05-15	3020	20	null	null	null
8	2023-05-15	3070	1	null	null	null
9	2023-05-15	3023	10	null	null	null
10	2023-05-15	3025	100	null	null	null

Gráfico N° 7: Output del nodo “Code JavaScript” para hoja CONSUMOS

Fuente: Elaboración propia

Como último paso de esta sección del flujo se incorpora un nodo de consulta en **PostgreSQL**, el cual se conecta directamente a la base de datos alojada en Supabase. Este nodo ejecuta un fragmento de código SQL diseñado para insertar en la tabla correspondiente (ya sea **compras** o **consumos**) únicamente los registros nuevos provenientes del archivo procesado.

Para lograrlo, el nodo compara los **id** de cada registro con los que ya existen en la base de datos. Solo aquellos registros cuyo id no se encuentre previamente almacenado son insertados. De esta manera, se evita la duplicación de datos y se garantiza que la actualización del inventario se realice de forma ordenada, precisa y sin generar inconsistencias.

e.2. Configuración de herramienta MCP Server

Se implementa un nodo de inicio **MCP Server** (Model Context Protocol), un componente diseñado para permitir que un modelo de inteligencia artificial interactúe con herramientas externas de manera controlada, segura y estructurada. MCP es un protocolo que actúa como un intermediario: recibe las solicitudes del agente de IA, ejecuta funciones predefinidas y devuelve solo la información necesaria, evitando accesos directos a la base de datos y mejorando la seguridad y trazabilidad del sistema.

En este proyecto, el MCP Server expone una serie de funciones específicas que permiten realizar consultas sobre la base de datos alojada en Supabase. Estas funciones se implementan utilizando nodos de tipo **PostgreSQL** conectados directamente a la base de datos. Cada herramienta corresponde a una consulta SQL concreta que el agente puede ejecutar a través del protocolo MCP. Las herramientas configuradas son las siguientes:

2. **select_stock:** Esta función permite consultar el stock actual de un producto o recuperar la tabla completa de stock. Se basa en una instrucción SQL que selecciona los registros de la tabla “stock” y devuelve campos como código, producto, rubro, total de compras, total de consumos y stock actualizado.
3. **select_consumos:** Esta herramienta ejecuta una consulta sobre la tabla “consumos” y permite obtener el detalle de consumos filtrado por código, fecha o cualquier otro parámetro que solicite el agente. Es fundamental para responder preguntas sobre movimientos, historial o consumos específicos.
4. **select_compras:** Su función es recuperar la información de la tabla “compras”, permitiendo visualizar ingresos de insumos, cantidades compradas, fechas, proveedores y cualquier dato relevante asociado a los registros de compra.
5. **select_días_stock:** Esta herramienta consulta la tabla “días_cobertura_insumos”, la cual contiene cálculos de consumo mensual, consumo diario, stock actualizado y estimación de días de cobertura. Permite que el agente responda preguntas vinculadas a disponibilidad futura, proyecciones y posibles faltantes.

En un MCP Server, cada una de estas herramientas funciona como un tool: una función que el modelo de IA puede llamar cuando lo necesita. El nodo “Tool” en la configuración de PostgreSQL dentro del MCP Server permite definir qué consulta SQL debe ejecutarse, cuáles son los parámetros permitidos y qué información retorna. De esta manera, el agente puede realizar consultas complejas sobre la base de datos sin operar directamente sobre ella, recibiendo únicamente los resultados de cada función expuesta.

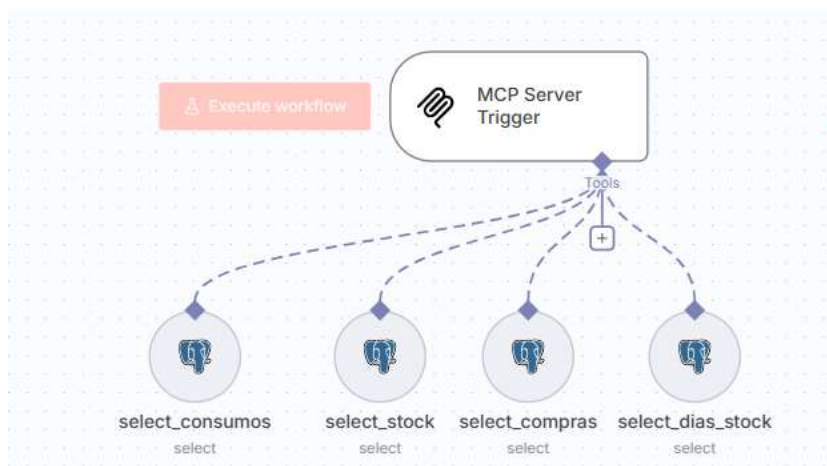


Gráfico N° 8: Trigger MCP Server y herramientas de PostgreSQL para consultas a la base de datos

Fuente: Elaboración propia

e.3. Creación del agente de IA conectado mediante chat

Finalmente, una vez configurados el flujo de automatización en n8n y las herramientas a través del MCP Server, se procede a la configuración del agente de inteligencia artificial que opera a través de una interfaz de chat. Este agente constituye la capa interactiva del sistema, permitiendo que un usuario formule consultas en lenguaje natural y reciba respuestas basadas en los datos almacenados en la base de datos.

Para comprender su funcionamiento, resulta fundamental identificar los componentes principales que integran un agente dentro de n8n:

1. **Modelo de lenguaje:** Encargado de interpretar las preguntas del usuario;
2. **Memoria:** Almacena temporalmente las interacciones para mantener el contexto de la conversación.
3. **Herramientas (tools):** Son funciones externas que el agente puede ejecutar para obtener información o realizar acciones específicas.

En este proyecto se utiliza como modelo de lenguaje **Google Gemini**, perteneciente a la familia de modelos generativos desarrollados por Google. Este modelo es el encargado de interpretar las consultas en lenguaje natural, seleccionar la herramienta adecuada del MCP Server y generar respuestas contextualizadas en función de los datos recibidos.

Como sistema de memoria, se implementa la SQL Simple Memory, una forma de almacenamiento liviano que conserva temporalmente información relevante del diálogo para que el agente pueda sostener una conversación coherente sin perder el contexto a lo largo de una misma sesión.

En cuanto a las herramientas, el agente utiliza un MCP Client, configurado de modo que se conecte directamente al MCP Server desarrollado previamente. A través de este cliente, el agente accede a las funciones expuestas (`select_stock`, `select_consumos`, `select_compras` y `select_días_stock`), las cuales ejecutan consultas SQL sobre la base de datos en Supabase sin que el modelo acceda directamente a la base de datos. De esta manera, se establece una integración segura y estructurada entre el agente, el MCP Server y la información almacenada.

Este diseño permite que el agente responda preguntas complejas sobre el inventario, los consumos, las compras o los días de cobertura, combinando la comprensión del lenguaje natural con consultas en tiempo real a la base de datos, y garantizando así precisión, trazabilidad y actualización constante de la información.

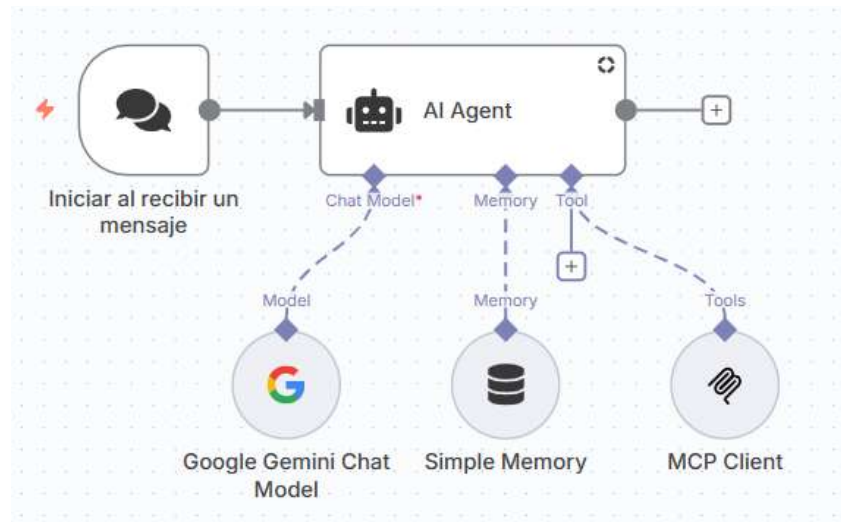


Gráfico N° 9: Estructura del agente de inteligencia artificial conectado mediante chat
Fuente: Elaboración propia

Con el fin de presentar de manera unificada la estructura completa del sistema desarrollado, a continuación se muestra el diseño final que integra los tres componentes principales del proyecto: el flujo de automatización encargado de la consolidación de datos, el MCP Server que expone las funciones de consulta a la base de datos, y el agente de inteligencia artificial que interactúa con el usuario mediante chat. Esta representación permite visualizar cómo cada una de estas partes se articula para conformar un sistema automatizado, coherente y operativo, capaz de actualizar información, ejecutar consultas estructuradas y responder en lenguaje natural a partir de los datos almacenados.

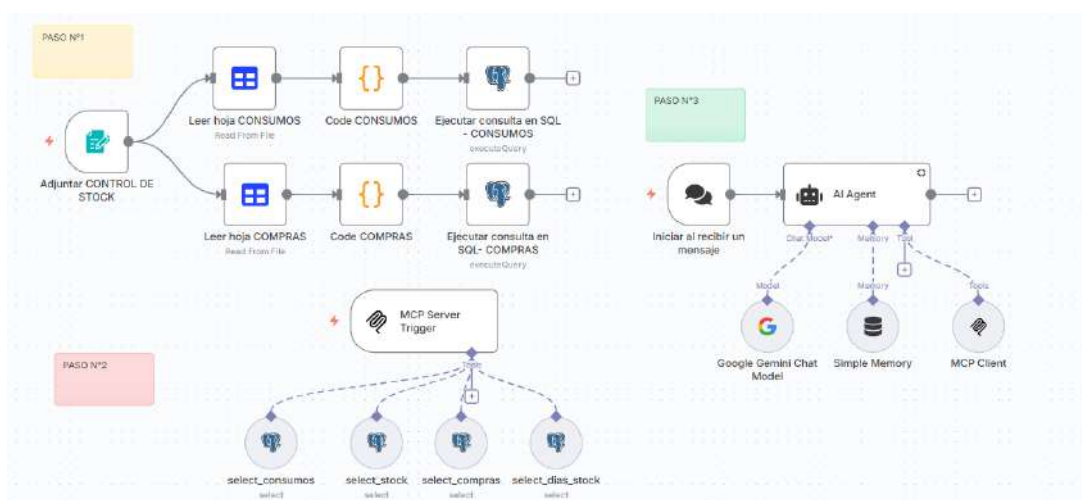


Gráfico N° 10: Diseño final del sistema integrado de automatización y agente de IA
Fuente: Elaboración propia

Recomendaciones

La incorporación de un agente de inteligencia artificial en la gestión de inventarios requiere no solo la implementación técnica del sistema, sino también un proceso de acompañamiento organizacional que garantice su integración efectiva en las rutinas de trabajo del Centro HD. A continuación, se presentan aspectos clave para facilitar la adopción exitosa de esta herramienta:

1. Estandarización y precisión en el registro de la información

Para que la automatización funcione de manera sostenible, es fundamental otorgar especial importancia a la precisión en el uso de la información y a la estandarización de los procedimientos, en particular aquellos vinculados al registro de datos:

- Es necesario promover hábitos de carga más rigurosos, evitando anotaciones informales, parciales o sin validación.
- La información debe registrarse en tiempo y forma, siguiendo criterios unificados entre turnos y sectores.

2. Acompañamiento al personal y actitud favorable al cambio

Asimismo, la transición hacia un sistema automatizado exige una actitud favorable al cambio y al aprendizaje.

- Es conveniente brindar explicaciones claras sobre el propósito del agente, los problemas que resuelve y los beneficios que aporta, especialmente en términos de reducción de tareas manuales y disminución de errores.
- Dar lugar a instancias de capacitación y espacios de consulta permiten que el personal incorpore las nuevas prácticas de manera gradual, reduciendo la sensación de imposición o ruptura.
- Una comunicación transparente favorece una actitud positiva hacia el cambio, indispensable para una adopción sostenible.

3. Mantenimiento básico y monitoreo del sistema

Finalmente, para asegurar la continuidad del funcionamiento del agente, es necesario establecer un esquema básico de mantenimiento:

- Revisar periódicamente la base de datos, verificar posibles inconsistencias
- Evaluar regularmente el desempeño del flujo de automatización, verificando que los procesos se ejecuten sin errores.
- Designar un responsable de monitoreo, encargado de supervisar el comportamiento del sistema y atender alertas o desvíos.

La automatización, entendida como un proceso dinámico y evolutivo, requiere cuidados mínimos pero constantes, que permiten consolidar sus beneficios y garantizar su aporte estratégico en la gestión de insumos del Centro HD.



Conclusiones

El desarrollo de este trabajo permitió constatar que la gestión de insumos en el ámbito sanitario, y particularmente en instituciones de alta complejidad como el Centro HD, constituye un proceso crítico cuya eficiencia depende en gran medida de la disponibilidad, precisión y oportunidad de la información. La observación y el análisis del circuito actual de registro y consolidación de datos evidenciaron la existencia de múltiples puntos de fricción asociados al uso de archivos fragmentados, la duplicación de tareas manuales y la escasa automatización en la actualización de inventarios. Estas condiciones no solo generan demoras en la toma de decisiones, sino también una pérdida potencial de trazabilidad y control sobre los recursos, lo que afecta de manera directa la calidad del servicio asistencial.

El diseño del agente de inteligencia artificial desarrollado en n8n representa un avance significativo hacia la transformación digital del proceso de gestión de inventario. Su estructura modular, basada en nodos de entrada, procesamiento y salida, permitió automatizar tareas rutinarias de lectura, clasificación y organización de datos provenientes de los registros de Farmacia, estableciendo un flujo de trabajo capaz de operar de manera autónoma. La migración del archivo “Control de Stock Dinámico” a una base de datos relacional en Supabase constituyó un paso central en este proceso, permitiendo manejar grandes volúmenes de información de forma más ordenada, confiable y escalable. La integración con modelos de lenguaje natural amplía además las posibilidades de consulta y análisis, posibilitando una interacción más fluida y accesible con la información disponible.

Desde el punto de vista técnico, la implementación del agente ofrece ventajas sustantivas en términos de eficiencia operativa, reducción de errores humanos y mejora en la trazabilidad del inventario. Asimismo, la incorporación de inteligencia artificial al circuito administrativo permitió avanzar hacia un enfoque de gestión basada en datos, en el cual la información se vuelve más confiable, más inmediata y más útil para el análisis cotidiano. El diagnóstico realizado también permitió confirmar la relevancia de aplicar metodologías mixtas al estudiar procesos de innovación tecnológica, dado que la triangulación entre observaciones, entrevistas y análisis documental posibilitó comprender tanto los aspectos técnicos como los comportamentales y culturales que influyen en la adopción del sistema.

En términos institucionales, los resultados preliminares muestran que la automatización parcial del proceso genera beneficios tangibles: reducción del tiempo de consolidación de la información, mayor ordenamiento del flujo de datos y un incremento en la transparencia del proceso de registro. No obstante, se identifican desafíos para su consolidación, entre ellos la necesidad de fortalecer los hábitos de registro, homogeneizar los procedimientos entre turnos y acompañar al personal en la adaptación a las nuevas herramientas tecnológicas. La sostenibilidad del sistema depende en gran medida de la calidad del dato de origen y de la apropiación interna del proceso automatizado.

A nivel conceptual, el proyecto reafirma el potencial de los agentes inteligentes basados en modelos para operar en entornos administrativos donde confluyen múltiples fuentes de información y donde la exactitud de los registros es determinante. Su capacidad



para organizar datos, mantener actualizaciones consistentes y responder consultas complejas los convierte en herramientas idóneas para optimizar procesos críticos en instituciones de salud.

Finalmente, la experiencia desarrollada sienta las bases para futuras líneas de trabajo orientadas a consolidar una arquitectura de gestión inteligente dentro del Centro HD. Aún sin incorporar funciones predictivas ni integraciones con otros subsistemas, la estructura creada permite continuar expandiendo el sistema de manera progresiva, siempre en función de las necesidades operativas y de la madurez organizacional. A largo plazo, avanzar hacia un modelo más integrado y sustentado en datos confiables permitirá fortalecer la eficiencia y la sostenibilidad del proceso de gestión de insumos, contribuyendo de manera directa a la calidad del servicio sanitario que la institución brinda.

Apéndice

Consultas de PostgreSQL usadas para crear y modificar la base de datos:

- **Tabla “stock”:**

```
CREATE TABLE stock (
codigo TEXT PRIMARY KEY,
rubro TEXT,
producto TEXT,
total_compras NUMERIC DEFAULT 0,
total_consumos NUMERIC DEFAULT 0,
stock_actual NUMERIC DEFAULT 0
);
```

- **Tabla “consumos”**

```
CREATE TABLE consumos (
id BIGINT PRIMARY KEY,
fecha DATE,
mes TEXT,
codigo TEXT,
cantidad NUMERIC,
rubro TEXT,
insumo TEXT,
solicitante_retira TEXT,
sector TEXT,
observaciones TEXT
);
```



- Tabla “compras”

```
CREATE TABLE compras (  
id BIGINT PRIMARY KEY,  
fecha DATE,  
codigo TEXT,  
cantidad NUMERIC,  
remito TEXT,  
proveedor TEXT,  
vencimiento DATE,  
observaciones TEXT  
);
```

- FOREIGN KEY en “consumos”

```
ALTER TABLE consumos  
ADD CONSTRAINT consumos_codigo_fkey  
FOREIGN KEY (codigo)  
REFERENCES stock(codigo)  
ON UPDATE CASCADE  
ON DELETE SET NULL;
```

- FOREIGN KEY en “compras”

```
ALTER TABLE compras  
ADD CONSTRAINT compras_codigo_fkey  
FOREIGN KEY (codigo)  
REFERENCES stock(codigo)  
ON UPDATE CASCADE  
ON DELETE SET NULL;
```

- Tabla “días_cobertura_stock”

```
CREATE TABLE dias_stock (  
codigo TEXT PRIMARY KEY,  
producto TEXT,  
consumos_por_mes JSONB,  
consumo_promedio INTEGER,  
consumo_diario INTEGER,  
stock_actual INTEGER,  
dias_stock INTEGER  
);
```

- TRIGGER: completar rubro e insumo en “consumos” + mes en español



Función

```
CREATE OR REPLACE FUNCTION fill_consumos_info()
RETURNS trigger AS $$
BEGIN
    -- Rubro e insumo desde stock
    SELECT s.rubro, s.producto
    INTO NEW.rubro, NEW.insumo
    FROM stock s
    WHERE s.codigo = NEW.codigo;

    -- Mes en español
    IF NEW.fecha IS NOT NULL THEN
        NEW.mes := to_char(NEW.fecha, 'TMMonth', 'lc_time=es_ES.utf8');
    END IF;

    RETURN NEW;
END;
$$ LANGUAGE plpgsql;
```

Trigger

```
CREATE TRIGGER trg_fill_consumos_info
BEFORE INSERT OR UPDATE ON consumos
FOR EACH ROW
EXECUTE FUNCTION fill_consumos_info();
```

- **TRIGGER: completar rubro y producto en “compras”**

Función

```
CREATE OR REPLACE FUNCTION fill_compras_info()
RETURNS trigger AS $$
BEGIN
    SELECT s.rubro, s.producto
    INTO NEW.rubro, NEW.producto
    FROM stock s
    WHERE s.codigo = NEW.codigo;

    RETURN NEW;
END;
$$ LANGUAGE plpgsql;
```

Trigger

```
CREATE TRIGGER trg_fill_compras_info
BEFORE INSERT OR UPDATE ON compras
FOR EACH ROW
EXECUTE FUNCTION fill_compras_info();
```

TRIGGER: actualizar total compras, total consumos y stock_actual en "stock"

Función

```
CREATE OR REPLACE FUNCTION actualizar_stock()
RETURNS trigger AS $$
DECLARE
    v_codigo text;
BEGIN
    IF (TG_OP = 'DELETE') THEN
        v_codigo := OLD.codigo;
    ELSE
        v_codigo := NEW.codigo;
    END IF;

    -- Total compras
    UPDATE stock
    SET total_compras = (
        SELECT COALESCE(SUM(c.cantidad), 0)
        FROM compras c
        WHERE c.codigo = v_codigo
    )
    WHERE codigo = v_codigo;

    -- Total consumos
    UPDATE stock
    SET total_consumos = (
        SELECT COALESCE(SUM(cn.cantidad), 0)
        FROM consumos cn
        WHERE cn.codigo = v_codigo
    )
    WHERE codigo = v_codigo;

    -- Stock actual
    UPDATE stock
    SET stock_actual = total_compras - total_consumos
```



```
WHERE codigo = v_codigo;
```

```
RETURN NEW;
```

```
END;
```

```
$$ LANGUAGE plpgsql;
```

Trigger

```
CREATE TRIGGER trg_stock_compras
```

```
AFTER INSERT OR UPDATE OR DELETE ON compras
```

```
FOR EACH ROW
```

```
EXECUTE FUNCTION actualizar_stock();
```

```
CREATE TRIGGER trg_stock_compras
```

```
AFTER INSERT OR UPDATE OR DELETE ON compras
```

```
FOR EACH ROW
```

```
EXECUTE FUNCTION actualizar_stock();
```

Consultas de Nodos de código utilizados en N8N para transformar las hojas “Compras” y “Consumos”

- **Hoja Consumos**

```
function normalizarClave(clave) {
```

```
  return clave
```

```
    .toLowerCase()
```

```
    .normalize('NFD').replace(/[\u0300-\u036f]/g, '') // elimina tildes
```

```
    .replace(/\s+/g, '_')
```

```
    .replace(/^[^w]/g, '');
```

```
}
```

```
function excelSerialToDate(serial) {
```

```
  if (isNaN(serial)) return null;
```

```
  const excelEpoch = new Date(1899, 11, 30);
```

```
  const fecha = new Date(excelEpoch.getTime() + serial * 86400000);
```

```
  return fecha.toISOString().slice(0, 10);
```

```
}
```

```
function parseFecha(valor) {
```

```
  if (valor == null || valor === '') return null;
```

```
  if (valor instanceof Date && !isNaN(valor)) {
```

```
    return valor.toISOString().slice(0, 10);
```

```
}
```



```

if (!isNaN(valor)) {
  return excelSerialToDate(Number(valor));
}

const partes = String(valor).split(/[\\-]/);
if (partes.length === 3) {
  if (partes[0].length === 4) {
    return `${partes[0]}-${partes[1].padStart(2, '0')}-${partes[2].padStart(2, '0')}`;
  } else {
    const anio = partes[2].length === 2 ? '20' + partes[2] : partes[2];
    return `${anio}-${partes[1].padStart(2, '0')}-${partes[0].padStart(2, '0')}`;
  }
}

return null;
}

return items.map(item => {
  const entrada = {};
  for (const [k, v] of Object.entries(item.json)) {
    entrada[normalizarClave(k)] = v;
  }

  // --- Fecha ---
  const fecha = parseFecha(entrada.fecha);

  // --- Cantidad ---
  let cantidad = 0;
  if (entrada.cantidad !== undefined && entrada.cantidad !== null && entrada.cantidad !== '')
  {
    if (typeof entrada.cantidad === 'number') {
      cantidad = entrada.cantidad;
    } else {
      const valor = String(entrada.cantidad).replace(',', '.');
      cantidad = parseFloat(valor) || 0;
    }
  }

  // --- Devolver solo las columnas necesarias ---
  return {

```



```
json: {
  id: entrada.id ? String(entrada.id).trim() : null,
  fecha,
  codigo: String(entrada.codigo || '').trim(),
  cantidad,
  solicitante_retira: entrada.solicitante_retira || entrada['solicitante/retira'] || null,
  sector: entrada.sector || null,
  observaciones: entrada.observaciones || null
}
};
});
```

- **Hoja Compras**

```
function excelSerialToDate(serial) {
  if (isNaN(serial)) return null;
  const excelEpoch = new Date(1899, 11, 30);
  const fecha = new Date(excelEpoch.getTime() + serial * 86400000);
  return fecha.toISOString().slice(0, 10);
}

const output = [];

// --- 1) Tomar solo la fila 7 como encabezados ---
const headerRow = items[2].json;
const headers = Object.values(headerRow);

// --- 2) Procesar desde fila 8 en adelante ---
for (let i = 3; i < items.length; i++) {
  const row = items[i].json;

  // si la fila está vacía, saltar
  if (Object.values(row).every(v => v === null || v === undefined || v === "")) continue;

  // mapear según las posiciones correctas
  const data = {
    id: row["__EMPTY"] || null,
    fecha: row["RENAL S.R.L."] ? excelSerialToDate(row["RENAL S.R.L."]) : null,
    codigo: row["__EMPTY_1"] ? String(row["__EMPTY_1"]).trim() : null,
    cantidad: row["__EMPTY_5"] || 0,
    remito: row["__EMPTY_6"] || null,
    proveedor: row["__EMPTY_7"] || null,
```




```
vencimiento: row["__EMPTY_8"] ? excelSerialToDate(row["__EMPTY_8"]) : null,
observaciones: row["__EMPTY_9"] || null,
};

output.push({ json: data });
}
return output;
```

Consultas de Nodos PostgreSQL utilizadas N8N para actualizar la base de datos en Supabase

- **Hoja Consumos**

```
INSERT INTO consumos (
  id,
  fecha,
  codigo,
  cantidad,
  solicitante_retira,
  sector,
  observaciones
)
VALUES (
  {{$json.id}},
  {{ $json.fecha ? '${'$json.fecha}' : 'NULL' }},
  {{ $json.codigo ? '${'$json.codigo}' : 'NULL' }},
  {{ $json.cantidad ? $json.cantidad : 'NULL' }},
  {{ $json.solicitante_retira ? '${'$json.solicitante_retira}' : 'NULL' }},
  {{ $json.sector ? '${'$json.sector}' : 'NULL' }},
  {{ $json.observaciones ? '${'$json.observaciones}' : 'NULL' }}
)
ON CONFLICT (id) DO NOTHING;
```

- **Hoja Compras**

```
INSERT INTO compras (
  id,
  fecha,
  codigo,
  cantidad,
  remito,
  proveedor,
  vencimiento,
  observaciones
```



```
)
VALUES (
  {{ $json.id }},
  {{ $json.fecha ? '${$json.fecha}' : 'NULL' }},
  {{ $json.codigo ?? 'NULL' }},
  {{ $json.cantidad ?? 'NULL' }},
  {{ $json.remito ? '${$json.remito}' : 'NULL' }},
  {{ $json.proveedor ? '${$json.proveedor}' : 'NULL' }},
  {{ $json.vencimiento ? '${$json.vencimiento}' : 'NULL' }},
  {{ $json.observaciones ? '${$json.observaciones}' : 'NULL' }}
)
ON CONFLICT (id) DO NOTHING;
```

Prompt utilizado para la configuración del agente de IA:

Sos un agente experto en análisis de stock y consumos.

Tenés acceso a las siguientes herramientas MCP:

- *select_consumos* → ejecuta consultas SQL sobre la tabla consumos
- *select_compras* → ejecuta consultas SQL sobre la tabla compras
- *select_stock* → ejecuta consultas SQL sobre la tabla stock
- *select_dias_stock* → ejecuta consultas SQL sobre la tabla dias_stock

Reglas obligatorias:

1. *Cada vez que necesites datos de la base, DEBÉS usar una de estas herramientas. Nunca respondés datos inventados.*
2. *Cuando uses una herramienta, siempre enviás un JSON con esta estructura:*

```
{
  "query": "<consulta SQL>"
}
```
3. *Nunca escribís la consulta SQL fuera del JSON. Nunca agregás texto fuera del mensaje de herramienta.*
4. *Elegís SIEMPRE la herramienta que corresponde a la tabla.*

Ejemplos:

- *Si el usuario pregunta por consumos* → *select_consumos*
 - *Si pregunta por compras* → *select_compras*
 - *Si pregunta por el stock* → *select_stock*
 - *Si pide días de stock* → *select_dias_stock*
5. *Podés hacer cualquier consulta SQL válida: SELECT, ORDER BY, GROUP BY, LIMIT, filtros, etc.*
 6. *Si el usuario pide comparaciones, cálculos o análisis complejos:*
 - *Primero pedís los datos usando la herramienta correcta*
 - *Luego procesás la información en tu respuesta final*



7. Nunca respondas "undefined", "query undefined" ni texto fuera del JSON.

Si no sabés qué tabla usar, preguntá primero.

8. SOLO usás estas herramientas. No tenés acceso directo a Postgres.

Tu objetivo es ayudarme a consultar, analizar y comparar datos del stock y consumos usando EXCLUSIVAMENTE las herramientas MCP.

Después de usar una herramienta MCP y recibir los resultados, SIEMPRE generás una respuesta final para el usuario.

Tu flujo es:

1. Elegís la herramienta adecuada

2. Ejecutás la consulta SQL enviando:

```
{
  "query": "<consulta>"
}
```

3. Recibís el resultado de la herramienta

4. Respondés al usuario en lenguaje natural explicando los datos obtenidos

Nunca termines la conversación únicamente invocando la herramienta.

Siempre devolvé una respuesta final.

Referencias

- Betancourt, J. (2016). *Gestión de procesos y mejora continua*. Editorial Alfaomega.
- Centro Único Coordinador de Ablación e Implante de Tucumán (CUCAITUC). (2025). *Datos provinciales – Sintra*.
https://sintra.incuca.gov.ar/mod_orgprovinciales/reportes_provinciales.php?jur=25
- Chiavenato, I. (2011). *Administración: Proceso administrativo* (4.ª ed.). McGraw-Hill.
- Chen, H. (2006). *Practical guide to mixed methods research*. SAGE Publications.
- Codd, E. F. (1970). A relational model of data for large shared data banks. *Communications of the ACM*, 13(6), 377–387.
- Creswell, J. W. (2013a). *Research design: Qualitative, quantitative, and mixed methods approaches* (4th ed.). SAGE Publications.
- DeCuir-Gunby, J. T., & Schutz, P. A. (2017). *Developing mixed methods research in the social and behavioral sciences*. Routledge.
- Dhar, V. (2013). Data science and prediction. *Communications of the ACM*, 56(12), 64–73. <https://doi.org/10.1145/2500499>
- Dumas, M., La Rosa, M., Mendling, J., & Reijers, H. A. (2018). *Fundamentals of business process management* (2nd ed.). Springer.
- Elmasri, R., & Navathe, S. B. (2017). *Fundamentals of database systems* (7th ed.). Pearson.
- Fayyad, U. M., Piatetsky-Shapiro, G., & Smyth, P. (1996). From data mining to knowledge discovery in databases. *AI Magazine*, 17(3), 37–54.



- Harrington, H. J. (1991). *Business process improvement: The breakthrough strategy for total quality, productivity, and competitiveness*. McGraw-Hill.
- Hernández-Sampieri, R., & Mendoza, C. (2008). *Metodología de la investigación* (5.ª ed.). McGraw-Hill.
- ISO. (2015). *ISO 9001:2015. Sistemas de gestión de la calidad – Requisitos*. Organización Internacional de Normalización.
- Jennings, N. R., & Wooldridge, M. (1998). Applications of intelligent agents. En *Intelligent agents* (pp. 3–28). Springer.
- Jiang, F., Jiang, Y., Zhi, H., Dong, Y., Li, H., Ma, S., & Wang, Y. (2017). Artificial intelligence in healthcare: Past, present and future. *Seminars in Cancer Biology*, 54, 1–11. <https://doi.org/10.1016/j.semcan.2017.04.001>
- Klein, S. (2016). *Business process management: Concepts, languages, architectures*. Springer.
- Kotter, J. P. (1996). *Leading change*. Harvard Business School Press.
- Lieber, J., & Weisner, T. (2010). *Integrating mixed methods in social research*. Oxford University Press.
- Management Sciences for Health. (2002). *La gestión del suministro de medicamentos*. OMS/OPS.
- Ministerio de Salud de la Nación. (2019). *Segunda Encuesta Nacional de Nutrición y Salud (ENNyS 2)*. https://www.argentina.gob.ar/sites/default/files/resolucion_ministerial_de_aprobacion_prog_erca_-_anexo.pdf
- Open Source Initiative. (2007). *The Open Source Definition (Version 1.9)*. <https://opensource.org/osd>
- PostgreSQL Global Development Group. (2024). *PostgreSQL documentation*. <https://www.postgresql.org/docs/>
- Provost, F., & Fawcett, T. (2013). *Data science for business: What you need to know about data mining and data-analytic thinking*. O'Reilly Media.
- Robbins, S. P., & Judge, T. A. (2017). *Comportamiento organizacional* (17.ª ed.). Pearson.
- Russell, S. J., & Norvig, P. (2020). *Artificial intelligence: A modern approach* (4th ed.). Pearson.
- Stallman, R. M. (2002). *Free software, free society: Selected essays of Richard M. Stallman*. GNU Press.
- Topol, E. (2019). *Deep medicine: How artificial intelligence can make healthcare human again*. Basic Books.
- Willcocks, L., & Lacity, M. (2016). *Service automation: Robots and the future of work*. SB Publishin