

**Escuela Superior Politécnica del Litoral**

**Facultad de Ingeniería en Electricidad y Computación**

Desarrollo de agente de inteligencia artificial para la automatización de procesos  
en un sistema ERP

TECH-436

**Proyecto Integrador**

Previo la obtención del Título de:

**Ingeniero en Ciencias de la Computación**

Presentado por:

Jorge Daniel Mawyin Cabay

Kevin Ariel Roldán Piloza

Guayaquil - Ecuador

Año: 2025

## **Dedicatoria**

---

Dedico este proyecto a mis padres, por ser mi base, y a mis amigos, por acompañarme en este camino. Pero, sobre todo, a mi hermana: mi gran ejemplo a seguir y la razón por la que siempre busco ser mejor.

**Jorge Daniel Mawyin Cabay**

Dedico este proyecto a mi familia, por su apoyo constante, comprensión y motivación a lo largo de este proceso académico.

**Kevin Ariel Roldan Piloza**

## Agradecimientos

---

Agradecemos a las personas que participaron en la revisión del proyecto y en la realización de pruebas, cuyas observaciones y comentarios permitieron identificar mejoras y fortalecer el desarrollo del trabajo.

## Declaración Expresa

Nosotros **Jorge Daniel Mawyin Cabay** y **Kevin Ariel Roldán Pilozo** acordamos y reconocemos que:

La titularidad de los derechos patrimoniales de autor (derechos de autor) del proyecto de graduación corresponderá al autor o autores, sin perjuicio de lo cual la ESPOL recibe en este acto una licencia gratuita de plazo indefinido para el uso no comercial y comercial de la obra con facultad de sublicenciar, incluyendo la autorización para su divulgación, así como para la creación y uso de obras derivadas. En el caso de usos comerciales se respetará el porcentaje de participación en beneficios que corresponda a favor del autor o autores.

La titularidad total y exclusiva sobre los derechos patrimoniales de patente de invención, modelo de utilidad, diseño industrial, secreto industrial, software o información no divulgada que corresponda o pueda corresponder respecto de cualquier investigación, desarrollo tecnológico o invención realizada por nosotros durante el desarrollo del proyecto de graduación, pertenecerán de forma total, exclusiva e indivisible a la ESPOL, sin perjuicio del porcentaje que nos corresponda de los beneficios económicos que la ESPOL reciba por la explotación de nuestra innovación, de ser el caso.

En los casos donde la Oficina de Transferencia de Resultados de Investigación (OTRI) de la ESPOL comunique a los autores que existe una innovación potencialmente patentable sobre los resultados del proyecto de graduación, no se realizará publicación o divulgación alguna, sin la autorización expresa y previa de la ESPOL.

Guayaquil, 17 de octubre del 2025.

---

Jorge Daniel Mawyin Cabay

---

Kevin Ariel Roldán Pilozo

## **Evaluadores**

---

**Ronald Raul Criollo Bonilla**

Profesor de Materia

---

**Francisco Miguel Ramírez Méndez**

Tutor de proyecto

## RESUMEN

En los sistemas ERP actuales, las interfaces gráficas tradicionales pueden resultar complejas para los usuarios, especialmente para aquellos con poca experiencia. En el proyecto actual se desarrolla un agente conversacional basado en inteligencia artificial para realizar consultas y ejecutar operaciones del ERP mediante lenguaje natural, con el objetivo de reducir la curva de aprendizaje sin requerir conocimientos técnicos avanzados. Para cumplir esto, se integra modelos de lenguaje de gran escala (LLM), un sistema de recuperación aumentada por generación (RAG) y el estándar Model Context Protocol (MCP) para la ejecución controlada de herramientas del ERP, apoyados por un servicio de orquestación y una base de datos para el almacenamiento embeddings. Durante la evaluación con usuarios con y sin experiencia en sistemas ERPs se pudo comprobar que usuarios nuevos fueron los principales beneficiados, mejorando la comprensión y localización de los procesos, mientras que los usuarios experimentados usaron el agente se mas como un apoyo para consultas generales. No obstante, se detectó que la latencia de respuesta del agente en esta fase piloto puede afectar su uso en tareas puntuales. En conclusión, el proyecto demuestra que un agente conversacional es una solución viable como capa de apoyo para mejorar la interacción humano ERP.

**Palabras clave:** Agente conversacional, sistemas ERP, automatización, lenguaje natural, MCP.

## ABSTRACT

*In modern ERP systems, traditional graphical user interfaces (GUIs) can be complex, particularly for inexperienced users. This project develops an AI-driven conversational agent that enables natural language queries and the execution of selected ERP operations. The primary objective is to flatten the learning curve and facilitate access to key functionalities without requiring advanced technical expertise. The implemented architecture integrates Large Language Models (LLMs), Retrieval-Augmented Generation (RAG), and the Model Context Protocol (MCP) for the controlled execution of ERP tools, supported by an orchestration service and a vector database for embedding storage. Evaluations involving both experienced and inexperienced ERP users demonstrated that the agent primarily benefits new users by improving process comprehension and discovery. In contrast, experienced users utilized the agent as a support tool for general inquiries. However, findings indicated that response latency during this pilot phase could hinder its effectiveness for specific tasks. In conclusion, the project demonstrates that a conversational agent is a viable support layer for enhancing human–ERP interaction.*

**Keywords:** *Conversational agent, ERP systems, automation, natural language, MCP.*

# ÍNDICE GENERAL

Resumen .....	i
Abstract .....	ii
Índice general .....	iii
Abreviaturas .....	vii
Índice de figuras .....	viii
Índice de tablas .....	ix
Capítulo 1 .....	1
1. Introducción.....	2
1.1 Descripción del Problema .....	3
1.2 Justificación del Problema.....	4
1.3 Objetivos .....	5
1.3.1 Objetivo general .....	5
1.3.2 Objetivos específicos.....	5
1.4 Marco teórico .....	5
1.4.1 Procesamiento de lenguaje natural .....	6
1.4.2 Agentes conversacionales en contextos empresariales .....	7
1.4.3 Modelos de arquitectura en agentes conversacionales .....	8
1.4.4 Sistemas ERP en la gestión empresarial.....	9
1.4.5 Interacción humano-computadora en entornos empresariales.....	10
Capítulo 2 .....	12
2. Metodología.....	13
2.1 Análisis.....	13

2.1.1	Requerimientos.....	13
2.1.2	Alcance y limitaciones de la solución .....	14
2.1.3	Riesgos y beneficios de la solución.....	15
2.1.4	Usuarios de la solución.....	16
2.1.5	Obtención y preparación de datos .....	17
2.2	Prototipado .....	17
2.2.1	Escenario Actual y Alternativas Conceptuales .....	17
2.2.2	Matriz de Evaluación Comparativa y Decisión.....	18
2.2.3	Evaluación del prototipo .....	20
2.3	Estrategia de orquestación e implementación del agente.....	21
2.3.1	Alternativas evaluadas:.....	21
2.3.2	Criterios de decisión.....	22
2.3.3	Decisión.....	22
2.3.4	Implicaciones en el diseño .....	23
2.3.5	Riesgos específicos y mitigación .....	23
3.	Diseño de la Solución.....	24
3.1	Arquitectura.....	24
3.1.1	Modelo de “4+1” Vistas de Kruchten .....	24
3.2	Diseño de la Base de Datos .....	34
3.3	Flujo de Orquestación en n8n.....	35
Capítulo 3	.....	37
4.	Resultados y análisis .....	38
4.1	Plan de Implementación .....	38

4.1.1	Sprints.....	38
4.1.2	Manuales .....	40
4.2	Pruebas .....	41
4.3	Resultados .....	42
4.3.1	Resultados de desempeño y eficiencia operativa .....	43
4.3.2	Tasa de éxito y manejo de errores .....	44
4.3.3	Comparativa por perfil: Expertos vs. Nuevos Usuarios .....	45
4.3.4	Resultados de usabilidad y satisfacción .....	46
4.3.5	Análisis cualitativo de la experiencia de usuario .....	47
4.4	Análisis de Costos .....	47
4.4.1	Escenario de cálculo .....	48
4.4.2	Resumen de costos estimados .....	49
Capítulo 4	.....	50
5.	Conclusiones y recomendaciones.....	51
5.1	Conclusiones .....	51
5.2	Recomendaciones .....	52
6.	Referencias .....	54
7.	Apendices .....	59
7.1	Apéndice A – Historias de Usuario y Requerimientos.....	59
7.1.1	Especificación de Historias de Usuario.....	59
7.1.2	Matriz de Relación: Escenarios vs. Requerimientos .....	69
7.1.3	Definición Detallada de Requerimientos .....	70
7.2	Apéndice B – Pantallas del Chat del agente.....	74

7.3	Apéndice C - Plan de Pruebas y Diseño Experimental .....	76
7.3.1	Descripción General y Objetivos .....	76
7.3.2	Participantes y Entorno .....	76
7.3.3	Escenarios de Prueba.....	77
7.3.4	Procedimiento y Métricas.....	78
7.3.5	Trazabilidad de Requisitos .....	78
7.3.6	Cronograma y Riesgos .....	79
7.3.7	Instrumentos de Medición Utilizados.....	79
7.3.8	Encuesta de Usabilidad (SUS) y Preguntas Abiertas .....	80
7.4	Apéndice D – Análisis de precios bajo escenario de LLM APIs .....	82

## ABREVIATURAS

API	Application Programming Interface
ERP	Enterprise Resource Planning
ESPOL	Escuela Superior Politécnica del Litoral
GUI	Graphical User Interface
IA	Inteligencia Artificial
LLM	Large Language Models
MCP	Model Context Protocol
NLG	Natural Language Generation
NLP	Natural Language Processing
NLU	Natural Language Understanding
RAG	Retrieval-Augmented Generation
SUS	System Usability Scale

## ÍNDICE DE FIGURAS

Figura 1 <i>Modelo de referencia de un agente híbrido: razonamiento, recuperación y ejecución</i> .....	9
Figura 2 <i>Fases del proceso metodológico del proyecto</i> .....	13
Figura 3 <i>Diagrama de Casos de Uso del Agente Conversacional Integrado con ERP</i> ....	25
Figura 4 <i>Diagrama de Clases del Orquestador del Agente IA</i> .....	27
Figura 5 <i>Diagrama de Actividad del flujo principal del agente</i> .....	28
Figura 6 <i>Diagrama de Actividad del flujo subida de archivos para el RAG</i> .....	30
Figura 7 <i>Diagrama de secuencia del flujo de interacción entre el usuario y el agente conversacional</i> .....	31
Figura 8 <i>Diagrama de secuencia del flujo interno del agente: consulta RAG y ejecución de herramientas</i> .....	32
Figura 9 <i>Diagrama de componentes de la arquitectura del agente conversacional</i> .....	33
Figura 10 <i>Diagrama de despliegue del agente conversacional y servicios asociados</i> ....	34
Figura 11 <i>Diagrama de base de datos del agente conversacional</i> .....	35
Figura 12 <i>Flujos de orquestación del agente conversacional y del sistema RAG en n8n</i>	36
Figura 13 <i>Cronograma Sprint 0 - Fase uno</i> .....	38
Figura 14 <i>Cronograma Sprint 1 - Fase dos</i> .....	39
Figura 15 <i>Cronograma Sprint 2 - Fase 3</i> .....	40
Figura 16 <i>Tiempo promedio de ejecución por tarea (Segundos)</i> .....	43
Figura 17 <i>Promedio de interacciones requeridas (Clicks vs. Mensajes)</i> .....	44
Figura 18 <i>Curva de aprendizaje: Tiempos de ejecución según perfil de usuario</i> .....	45
Figura 19 <i>Puntaje obtenido en la escala SUS</i> .....	46

## ÍNDICE DE TABLAS

Tabla 1 <i>Análisis de riesgos generales del sistema</i> .....	15
Tabla 2 <i>Matriz de selección de prototipos</i> .....	19
Tabla 3 <i>Comparación de alternativas de orquestación para el agente IA.</i> .....	21
Tabla 4 <i>Comparación de criterios de decisión para selección de alternativa.</i> .....	22
Tabla 5 <i>Riesgos específicos y medidas de mitigación del agente IA con implementación de n8n.</i> .....	23
Tabla 6 <i>Estrategia de pruebas y validación del sistema.</i> .....	41
Tabla 7 <i>Resumen de costos mensuales estimados del agente IA.</i> .....	49
Tabla 8 <i>Historias de Usuario con sus respectivos escenarios</i> .....	60
Tabla 9 <i>Relación de historias de usuario con requerimientos</i> .....	69
Tabla 10 <i>Relación escenarios de prueba y requerimientos funcionales.</i> .....	78
Tabla 11 <i>Riesgos identificados del plan de pruebas y mitigación.</i> .....	79
Tabla 12 <i>Ficha de registro de métricas del plan de pruebas.</i> .....	80
Tabla 13 <i>Comparativa de APIs de modelos de lenguaje (precios por millón de tokens)</i> .	82
Tabla 14 <i>Comparativa de APIs de modelos de embeddings (precios por millón de tokens)</i> .....	83

# **CAPÍTULO 1**

## 1. INTRODUCCIÓN

En los últimos años, la digitalización de los procesos empresariales se ha convertido en un pilar fundamental para mantener la competitividad organizacional. En este contexto, los sistemas de planificación de recursos empresariales (ERP) se posicionan como herramientas clave, al integrar en una sola plataforma la gestión de inventarios, finanzas, recursos humanos y logística. Gracias a esta integración, la información deja de estar fragmentada y se garantiza una visión unificada y actualizada de las operaciones (Govea Souza, 2021).

A pesar de sus beneficios, los ERP tradicionales, aquellos como soluciones rígidas sin capacidades nativas de IA (Schütte, 2024, Capítulo 22), presentan serios desafíos de usabilidad dado que su interacción se basa exclusivamente en interfaces gráficas de usuario (GUI) poco intuitivas y que exigen conocimientos técnicos específicos, elevando la curva de aprendizaje y limitando la adopción del sistema. Estas dificultades en el uso de interfaces gráficas y falta de usabilidad son problemas que ocurren incluso en plataformas de alto nivel como Oracle NetSuite y SAP, resultando en un requerimiento mayor de capacitación del personal y reduciendo la eficiencia operativa, lo que convierte a esta complejidad en un tema central en el análisis de modernización de ERP (Junco, 2024). En consecuencia, una parte significativa de las funcionalidades permanece casi sin utilizarse, lo que afecta la productividad y el retorno de la inversión, creando una brecha crítica entre el potencial del software y su uso real.

Para abordar esta problemática, el presente proyecto propone el desarrollo de un agente conversacional que actúe como un traductor funcional entre el usuario y el ERP, siendo su principal aportación la capacidad del agente para interpretar instrucciones en lenguaje natural y transformarlas en acciones ejecutables dentro del software empresarial. Esto se logra mediante un enfoque combinado de un sistema de consulta en tiempo real, de procesos y documentación del ERP, con un módulo de ejecución que convierte las intenciones del usuario en operaciones automatizadas. De esta manera, se busca disminuir la dependencia de la GUI, hacer más accesible el ERP y optimizar la eficiencia operativa.

En investigaciones recientes, como la de Mhaskey (2024), Pokala (2024) y Hrishev & Shakev (2022), se analiza las oportunidades de la IA en plataformas empresariales, centrándose

ampliamente en la analítica predictiva, la optimización de inventarios y la automatización de tareas repetitivas (RPA). Mientras estos trabajos aplican la inteligencia artificial como soporte analítico o para automatizar tareas auxiliares, el presente proyecto adopta un enfoque distinto: utiliza modelos de lenguaje para ejecutar directamente operaciones del ERP, permitiendo acciones como consultar, crear o modificar registros a partir de instrucciones en lenguaje natural. Este enfoque habilita la automatización de procesos operativos complejos mediante una interfaz conversacional y representa una contribución diferenciada frente a las aplicaciones centradas únicamente en predicción o automatización repetitiva.

Esta contribución se presenta de forma organizada en cuatro capítulos del presente documento. El Capítulo 1 establece el contexto de la problemática y realiza una revisión del estado del arte. El Capítulo 2 detalla la metodología y el diseño de la solución que cubre desde el análisis de requerimientos hasta el modelado de datos. El Capítulo 3 expone los resultados obtenidos y el análisis de las pruebas de validación del prototipo. Finalmente, el trabajo cierra en el Capítulo 4 con las conclusiones, contrastando los hallazgos con los objetivos planteados y sugiriendo líneas de trabajo futuras.

## **1.1 Descripción del Problema**

Los ERP tradicionales presentan una interfaz rígida lo que le exige al usuario conocer sobre la terminología técnica y rutas de navegación específicas, lo que genera problemas de usabilidad en la operación diaria (Asif et al., 2022). Esto se traduce en pérdida de tiempo en tareas repetitivas, errores al obtener información y una curva de aprendizaje que frena la productividad. Estas dificultades se repiten en organizaciones que dependen del ERP como sistema central de gestión (Haro et al., 2023, p. 3).

Según Asif et al. (2022), la falta de personalización, la complejidad de navegación y la necesidad de capacitación constante generan frustración entre los usuarios y dificultan la adopción. Los autores señalan que la usabilidad y la experiencia del usuario son factores que

determinan el éxito de un ERP, y que la interacción debe rediseñarse hacia modelos más accesibles.

En paralelo, la complejidad operativa de estos sistemas se ha convertido en un freno para la innovación. (Pokala, 2024) confirma que los ERP tradicionales han tenido dificultades para facilitar la toma de decisiones basada en datos y la automatización a gran escala. Castillo-Salamín (2025), por su parte, plantea que incorporar herramientas como IA, big data e IoT permite reducir errores derivados de esa complejidad.

Frente a esto, los agentes conversacionales basados en IA ofrecen una alternativa: traducir instrucciones en lenguaje natural a operaciones dentro del sistema. Con ello se busca reducir la carga cognitiva del usuario, mejorar la precisión en tareas operativas y aumentar la productividad, como se detalla en la Sección 1.4.2.

## **1.2 Justificación del Problema**

Superar las limitaciones de usabilidad de los ERP tradicionales es relevante porque impacta directamente en la productividad, la calidad de los datos y la eficiencia operativa. Cuando las interfaces exigen conocimientos técnicos avanzados, el personal comete más errores, tarda más en tareas rutinarias y subutiliza las funcionalidades disponibles (Jawad & Villatoro Amaya, 2023). Este problema no es menor ya que afecta tanto los costos operativos como la capacidad de las empresas para aprovechar sus herramientas digitales.

El desarrollo de un agente conversacional que permita interactuar con el ERP mediante lenguaje natural aborda esta brecha. (Asfoura et al., 2023) destacan que este tipo de interfaces reduce la dependencia de capacitación técnica y promueve una experiencia de uso más directa. Además, integrar IA en la gestión empresarial contribuye a modernizar el flujo de trabajo y a reducir errores de operación manual.

### **1.3 Objetivos**

#### **1.3.1 *Objetivo general***

Desarrollar un agente conversacional basado en inteligencia artificial que traduzca instrucciones en lenguaje natural a operaciones dentro de un sistema ERP, con el propósito de reducir la complejidad operativa y mejorar la eficiencia en los procesos empresariales.

#### **1.3.2 *Objetivos específicos***

Identificar los procesos del sistema ERP que puedan ser automatizados, incluyendo las acciones clave y sus parámetros.

Diseñar la arquitectura del agente conversacional de IA, estableciendo los componentes principales del flujo de procesamiento del lenguaje natural y el mecanismo de invocación de operaciones.

Prototipar un agente conversacional de IA capaz de interpretar instrucciones en lenguaje natural y ejecutar operaciones automáticas dentro del sistema ERP.

Evaluar el desempeño del agente mediante pruebas que midan la precisión de la interpretación y la eficacia de las operaciones ejecutadas.

### **1.4 Marco teórico**

El marco teórico fue elaborado mediante una revisión del tipo exploratoria y estructurada de literatura de Scopus, IEEE Xplore y Google Scholar, fuentes donde se localizaron la mayoría de los estudios que se utilizaron en este documento. La selección incluyó artículos académicos, libros técnicos y revisiones recientes relacionados con inteligencia artificial, procesamiento de lenguaje natural, agentes conversacionales y modernización de sistemas ERP. Se priorizaron trabajos publicados entre 2020 y 2025 y se eligieron aquellos que aportan conceptos directamente vinculados con el desarrollo del agente propuesto.

El desarrollo de agentes conversacionales inteligentes representa en la actualidad un avance significativo en la aplicación moderna de la inteligencia artificial (IA). Según Muro et al. (2024, p. 7) la automatización mediante IA está revolucionando los entornos productivos al incrementar la eficiencia y reducir costos operativos, lo que refleja su impacto transversal en diversos sectores. En este sentido, los agentes conversacionales surgen como una extensión natural de dicha automatización, ya que ofrecen interfaces en lenguaje natural que reducen la complejidad operativa al consultar información y activar funciones transaccionales.

#### ***1.4.1 Procesamiento de lenguaje natural***

Al combinar la lingüística y la informática, el procesamiento de lenguaje natural (NLP) se ha consolidado como una de las ramas más importantes de la inteligencia artificial, permitiendo que los sistemas sean capaces de interactuar usando lenguaje humano. En la práctica, esto se divide en dos grandes tareas: comprensión del lenguaje natural (NLU), que busca entender la intención y el significado; y generación del lenguaje natural (NLG), que asume la tarea de producir respuestas en lenguaje humano (Jamuie et al., 2025).

En los últimos años, el NLP ha avanzado rápidamente con modelos de gran escala y arquitecturas Transformers que han mejorado la coherencia, la contextualización y la capacidad de razonamiento de los sistemas (Brown et al., 2020; Supriyono et al., 2024), avances que permiten aplicaciones más precisas en tareas como los agentes conversacionales, la traducción automática, el análisis de sentimientos y la generación de texto.

El enfoque moderno del NLP comprende tres etapas claves que son el preprocesamiento del texto, la representación semántica mediante embeddings contextuales y el modelado que interpreta relaciones complejas del lenguaje (Treviso et al., 2023). Al aplicar dichas etapas en sistemas empresariales, la construcción de agentes conversacionales se complementan con varias técnicas como la recuperación aumentada por generación (RAG), para recupera evidencias de fuentes externas antes de generar, y llamadas a herramientas (tool calling) que ofrecen

autonomía para conectarse mediante APIs y ejecutar acciones directas en la aplicación (Putri et al., 2024). Esta sinergia sienta las bases para que el agente no solo entienda la petición, sino que también actúe sobre el sistema digital de la organización.

#### ***1.4.2 Agentes conversacionales en contextos empresariales***

Syvänen & Valentini (2020) definen a los agentes conversacionales como sistemas de inteligencia artificial que interpretan intenciones y generan respuestas lógicas, facilitando la interacción y comunicación entre humanos y computadoras mediante texto o voz. El poder emplear aprendizaje profundo y modelos preentrenados hace que estas herramientas logren mantener conversaciones más naturales conservando un contexto, a diferencia de los chatbots antiguos basados en simples reglas (NSAIF et al., 2024).

Justamente por su facilidad para acoplarse a plataformas web y corporativas, estos agentes desempeñan un rol vital en la automatización y asistencia digital moderna (NSAIF et al., 2024). En uso empresarial, estos agentes actúan como capa de interfaz frente a sistemas complejos (p. ej., un ERP): toman solicitudes en lenguaje natural y las convierten en acciones concretas como consultar, crear, actualizar o validar información. Con ello reducen la curva de aprendizaje y errores de captura así como ayudar a agilizar tareas repetitivas (Hernández & Ávila, 2024). La conexión con los sistemas suele hacerse mediante integraciones y APIs que pueden invocar herramientas (tool calling) para ejecutar operaciones o recuperar evidencia antes de responder, lo que aumenta la confiabilidad y veracidad de la salida brindada.

En escenarios más amplios, comúnmente se piensa en sistemas multi-agente (MAS), donde varios agentes colaboran y se reparten tareas para lograr cumplir con un objetivo. Pérez et al. (2019) describen estos sistemas con elementos como roles, protocolos de interacción y reglas de coordinación para decidir “quién hace qué” y “cuándo”. En organizaciones con múltiples sistemas (ERP, WMS, CRM, IoT), esta coordinación permite decisiones distribuidas y flujos más robustos, lo que se refleja en mejoras reportadas en planificación y control, pronóstico de

demanda, inventarios, reabastecimiento, coordinación con proveedores y logística (Hernández & Ávila, 2024).

### ***1.4.3 Modelos de arquitectura en agentes conversacionales***

Los agentes conversacionales modernos se organizan como arquitecturas compuestas que integran: NLU (para captar intención y extraer datos), orquestación del diálogo (para decidir el siguiente paso), NLG (para redactar la respuesta) y mecanismos de acceso a conocimiento y herramientas. De acuerdo con el marco reciente (Krishnan, 2025), los LLM funcionan como núcleo de razonamiento, pero se amplifican mediante módulos de memoria (episódica/semántica), planificación (descomponer tareas), uso de herramientas (tool calling) y percepción de contexto, lo que permite pasar de respuestas puramente textuales a acciones verificables sobre sistemas externos. En este esquema, el orquestador mantiene el estado conversacional, decide la siguiente acción (preguntar, recuperar, ejecutar, responder, etc) y aplica políticas de seguridad/permisos antes de invocar servicios del sistema destino.

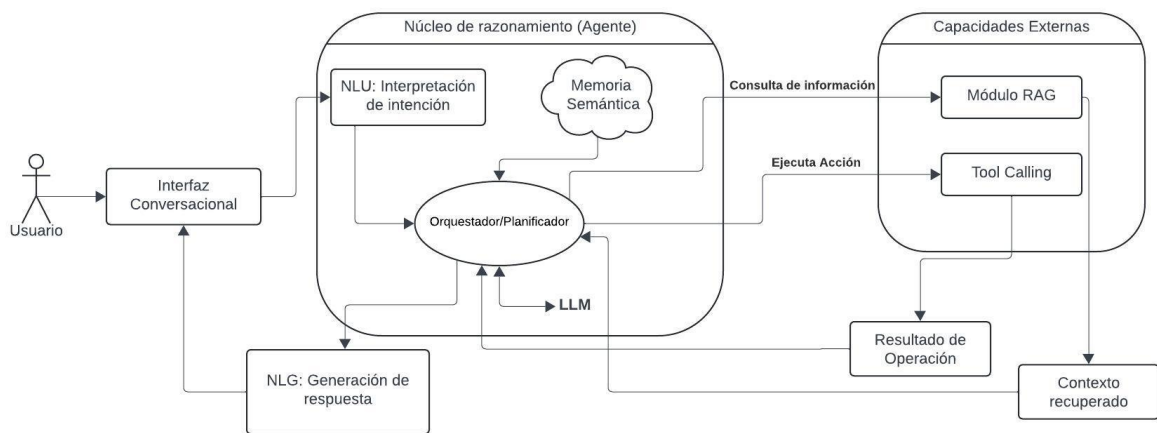
En la práctica, las arquitecturas LLM-centradas se enriquecen con dos técnicas. La primera es la recuperación aumentada por generación (RAG). La segunda es el uso de conocimiento estructurado, concretamente bases de conocimiento (KB). Estas bases a menudo se modelan mediante grafos de conocimiento (KG). RAG recupera evidencias actuales y confiables antes de generar. Por su parte, los grafos de conocimiento aportan relaciones semánticas y trazabilidad en dominios empresariales. El tool calling conecta el agente con APIs del ERP. Esto permite consultar o modificar entidades de negocio bajo control de roles y auditoría (Salvatore Raieli & Gabriele Iuculano, 2025).

En conjunto, estas piezas configuran un patrón de referencia para arquitecturas agénticas modernas (Abou Ali et al., 2025), tal como se ilustra en la Figura 1. Este esquema inicia con la interfaz conversacional y el procesamiento NLU, seguido por un núcleo de orquestación que, apoyado por el LLM, aplica estrategias de planificación y políticas de seguridad. Dependiendo

de la intención detectada, el sistema deriva el flujo hacia la recuperación de información (RAG/KB) para consultas documentales, o hacia el uso de herramientas (Tool Calling) para interactuar con las APIs del ERP.

### Figura 1

*Modelo de referencia de un agente híbrido: razonamiento, recuperación y ejecución*



Basado en esta jerarquía, el presente proyecto adopta un Modelo de Arquitectura Híbrida denominado Agentic RAG (Singh et al., 2025). A diferencia de los chatbots puramente informativos o los scripts de automatización rígidos, este modelo integra la capacidad de razonamiento del LLM con módulos de ejecución transaccional (NSAIF et al., 2024). Esta elección permite que el agente no solo responda preguntas basándose en manuales (componente RAG), sino que también actúe como un operador dentro del sistema de gestión (componente de Herramientas), cumpliendo con los requisitos de trazabilidad y mantenibilidad necesarios en entornos empresariales.

#### 1.4.4 *Sistemas ERP en la gestión empresarial*

Para Haro et al. (2023) centralizar procesos clave mediante un ERP eleva la productividad y garantiza acceso inmediato a datos de distintas áreas como finanzas o producción, agilizando directamente la toma de decisiones gerenciales. La implementación de un ERP funcional exige planificar el alcance, elegir al proveedor, configurar y personalizar el

software y ejecutar migraciones y pruebas, todo esto seguido de una capacitación y gestión del cambio hacia el personal.

Según (Jurado, 2024), los estudios académicos realizados a partir de Scopus y Web of Science muestran un crecimiento constante en la producción de investigaciones sobre ERP y gestión entre 2013 y 2023. En estas líneas de investigación se refleja un interés consolidado en temas de Tecnologías de la Información y la Comunicación (TIC), rendimiento corporativo y reingeniería de procesos aplicados a la cadena de suministro.

En la práctica, implementar estas herramientas genera impactos mixtos. Zabala et al. (2021) evaluaron el ERP del gobierno autónomo descentralizado (GAD) Municipal de Riobamba, encontrando directivos satisfechos por la unificación de datos internos, pero advirtiendo que la ciudadanía no percibió mejoras en el servicio. Esto evidencia que la calidad técnica del sistema debe ir obligatoriamente acompañada de estrategias continuas de adaptación.

#### ***1.4.5 Interacción humano-computadora en entornos empresariales***

Diseñar, evaluar y construir sistemas interactivos pensados en el usuario es el núcleo de la Interacción Humano-Computadora (IHC). Gordon Graell (2023) la describe como un esfuerzo multidisciplinario donde convergen la ingeniería de software, la ergonomía y el diseño visual para abarcar tanto el uso clásico de escritorio como interfaces táctiles, auditivas o inmersivas.

En el ámbito empresarial, la IHC se traduce en interfaces que son fáciles de usar, lo que ayuda a disminuir la carga cognitiva y a realizar tareas de manera efectiva. En (Asif et al., 2022), se menciona que la falta de etiquetas claras, los sistemas de búsqueda deficientes y el mal manejo de errores son obstáculos frecuentes que frenan la eficiencia operativa. Para mejorar la adopción y evitar perjudicar la productividad, las organizaciones necesitan integrar evaluaciones con los usuarios, como pruebas de usabilidad, evaluaciones heurísticas y recorridos cognitivos, para mejorar la adopción real de la herramienta.



## **CAPÍTULO 2**

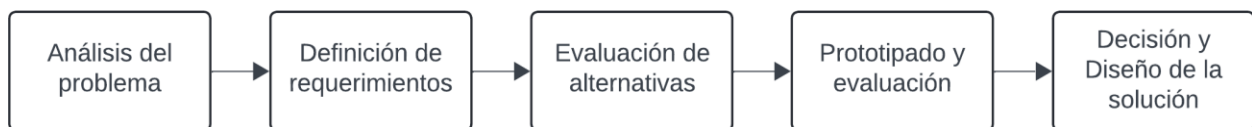
## 2. METODOLOGÍA.

El proyecto adoptó un enfoque iterativo basado en prototipado con el fin de validar de forma progresiva las decisiones del sistema antes de su diseño final. Se inició con el análisis del problema y la definición de los requerimientos, seguido por la evaluación de distintas alternativas de solución y la construcción de prototipos conceptuales.

La viabilidad de estos modelos iniciales se puso a prueba directamente con los actores involucrados, lo que permitió realizar ajustes antes de seleccionar el enfoque definitivo y definir la arquitectura del agente conversacional. El esquema detallado de estas etapas metodológicas se ilustra en la Figura 2.

### Figura 2

*Fases del proceso metodológico del proyecto*



### 2.1 Análisis

#### 2.1.1 Requerimientos

Los requerimientos del sistema fueron definidos a partir de la problemática de usabilidad en el ERP actual, dividiéndolos en funcionales y no funcionales y detallándolos en el *Apéndice A* para su revisión. Los requerimientos funcionales describen las capacidades del agente para interpretar lenguaje natural además de orquestar y ejecutar herramientas transaccionales en el ERP de manera autónoma. Por otro lado, los requerimientos no funcionales establecen restricciones de calidad enfocadas en la latencia de respuesta, seguridad y coherencia en la interacción, garantizando la viabilidad del agente en un entorno empresarial sin comprometer la integridad de los datos ni la experiencia del usuario final.

## 2.1.2 Alcance y limitaciones de la solución

### 2.1.2.1 Alcance

Como principales actividades/entregables se tiene:

- **Configuración y documentación de las herramientas del MCP:** selección, definición y registro de las herramientas del ERP que el agente podrá invocar.
- **Diseño de la arquitectura del agente:** definición de estructura interna, componentes y comunicación con el ERP web.
- **Desarrollo del componente MCP:** encargado de gestionar las herramientas disponibles y enlace entre el agente y las funciones del ERP.
- **Construcción del flujo principal del agente:** incluye la interpretación de instrucciones en lenguaje natural, la selección de herramientas y ejecución de acciones en el ERP.
- **Desarrollo del chat del agente en la interfaz web:** interfaz visual para la interacción del usuario con el agente y conexión con el backend.
- **Pruebas en un entorno controlado:** validación del funcionamiento del agente, incluyendo RAG y herramientas MCP, utilizando datos de prueba conforme al proceso de obtención descrito en la *Sección 2.1.5*.

### 2.1.2.2 Limitaciones

En esta sección se establecen los aspectos que quedan fuera del alcance del proyecto o se reservan para fases futuras.

- **Modalidad de Interacción:** El sistema se limita a entradas y salidas de texto; no incluye reconocimiento de voz ni procesamiento de archivos adjuntos (imágenes, documentos).
- **Cobertura Funcional:** La implementación no abarca la totalidad del sistema ERP, se restringe a los módulos piloto seleccionados para la validación, excluyendo el resto de los módulos operativos.

- **Seguridad y Accesos:** El agente opera bajo los permisos existentes y no tiene la capacidad de modificar roles, usuarios ni políticas de seguridad del ERP.
- **Fuente de Conocimiento:** El sistema RAG se alimenta exclusivamente de la documentación interna aprobada, cuya obtención y criterios de selección se describen en la *Sección 2.1.5*.
- **Escalabilidad e Implementación:** El enfoque se centra en un prototipo funcional para validación técnica; no incluye la adaptación a otros sistemas ERP comerciales.

### 2.1.3 Riesgos y beneficios de la solución

#### 2.1.3.1 Riesgos

Los riesgos detectados abarcan aspectos técnicos, operativos y de adopción del sistema.

La Tabla 1 resume sus causas, impactos y acciones de mitigación

**Tabla 1**

*Análisis de riesgos generales del sistema*

Riesgo	Causa	Impacto	Medida de mitigación
Fallas en la integración con el ERP	Cambios o incompatibilidades en la API	Retrasos o pérdida de comunicación entre módulos	Pruebas de integración continuas y documentación actualizada
Baja precisión del modelo de IA	Selección inadecuada del LLM o embeddings	Respuestas incorrectas o pérdida de confianza de los usuarios	Evaluar modelos y configuraciones antes de la integración
Resistencia de los usuarios a la adopción	Falta de capacitación o desconfianza en la IA	Baja adopción del agente	Sesiones de demostración, capacitación y soporte inicial
Riesgos de seguridad o	Configuraciones erróneas o	Acceso no autorizado o filtración de información	Políticas de seguridad, cifrado y revisiones periódicas

Riesgo	Causa	Impacto	Medida de mitigación
exposición de datos	vulnerabilidades en el acceso al ERP		
RAG desactualizado	Falta de mantenimiento de documentación	Respuestas obsoletas o inexactas	Establecer un calendario de actualización y control de versiones del RAG
Dependencia de infraestructura externa	Dependencia de infraestructura externa	Dependencia de infraestructura externa	Dependencia de infraestructura externa

### 2.1.3.2 Beneficios

- Reducción de la curva de aprendizaje y mejor aprovechamiento de las funcionalidades del ERP.
- Incremento en la velocidad y precisión de las tareas repetitivas o de consulta.
- Respuestas contextualizadas y fundamentadas en documentación verificada (RAG).
- Modernización del sistema ERP mediante una interacción más natural y accesible para el usuario final.

### 2.1.4 Usuarios de la solución

- **Usuarios operativos del ERP:** ejecutan tareas interactuando con el agente para realizar consultas o ejecutar procesos comunes sin navegar por múltiples módulos.
- **Equipo de tecnología de información (TI) o desarrollo:** gestiona las herramientas del MCP, configura el agente y mantiene actualizada la base de conocimiento (RAG).
- **Gerentes:** evalúan el impacto del agente en la eficiencia operativa, la reducción de tareas repetitivas y la mejora en la toma de decisiones que se traducen a reducción de costos.

### **2.1.5 Obtención y preparación de datos**

La solución propuesta utiliza datos provenientes de dos fuentes principales: la documentación interna del ERP y los datos empleados durante la ejecución de pruebas funcionales en un entorno controlado.

Para el componente de recuperación aumentada por generación (RAG), se utilizaron documentos internos del ERP previamente aprobados, como manuales de usuario y guías operativas. Se priorizaron documentos actualizados y relevantes para los módulos piloto.

Los datos utilizados para las pruebas y validaciones funcionales son escenarios controlados del ERP, utilizando información de prueba y evitando el uso de datos sensibles, permitiendo validar el comportamiento del agente y su integración con el ERP sin comprometer la información real.

## **2.2 Prototipado**

El diseño y validación del prototipo de la solución se dividió en dos etapas. Primero se contrastaron diversas alternativas de solución para encontrar la más viable y posteriormente se realizó una evaluación del prototipo de la interfaz de usuario (UI) con los interesados clave, centrada en validar el diseño de interacción y la usabilidad de la propuesta.

### **2.2.1 Escenario Actual y Alternativas Conceptuales**

Tomando la situación actual como punto de partida, el análisis exploró tres rutas conceptuales diferentes para abordar el problema.

#### **Escenario Actual (Línea Base): Interfaz Gráfica (GUI) Tradicional**

Operar el ERP vigente exige interactuar estrictamente con su interfaz gráfica tradicional. Este enfoque presenta una alta curva de aprendizaje y una fuerte dependencia de la operación manual, disminuyendo la eficiencia día a día. No se considera una solución, sino un punto de referencia para medir mejoras.

### **Prototipo 1: Enfoque Determinista (Basado en Reglas)**

La primera alternativa plantea construir un chatbot simple basado en un árbol de decisión y reglas fijas, limitado a interpretar comandos exactos y ejecutar scripts preconfigurados. Este enfoque carece de capacidad para comprender lenguaje natural, sinónimos o intenciones ambiguas (Asfoura et al., 2023; NSAIF et al., 2024).

### **Prototipo 2: Enfoque de Conocimiento (Basado en RAG)**

Este prototipo introduce un agente de IA con un Modelo de Lenguaje (LLM) y un sistema de Recuperación Aumentada por Generación (RAG). Así, el agente logra nutrirse de la base documental del ERP para contestar dudas cotidianas. Sin embargo, este prototipo se limita a capacidades informativas y no permite ejecutar acciones transaccionales, lo que restringe su impacto en la automatización de procesos (Putri et al., 2024; Singh et al., 2025).

### **Prototipo 3: Enfoque Híbrido (RAG + Herramientas)**

Este modelo toma la arquitectura del Prototipo 2 y le agrega la capacidad de ejecución de herramientas a través de APIs (tool calling). De este modo, se permite no solo consultar información, sino también interpretar intenciones del usuario y transformarlas en operaciones directas dentro del ERP.

#### ***2.2.2 Matriz de Evaluación Comparativa y Decisión***

Para definir qué camino tomar, se cruzaron las tres alternativas contra el escenario base utilizando la matriz de evaluación detallada en la Tabla 2. El análisis se centró en criterios clave como curva de aprendizaje, capacidad transaccional, acceso al conocimiento, flexibilidad y complejidad de implementación. Esta comparación permite identificar el enfoque más alineado con los objetivos del proyecto (Abou Ali et al., 2025; Krishnan, 2025; Salvatore Raieli & Gabriele Iuculano, 2025).

**Tabla 2***Matriz de selección de prototipos*

Criterio de Evaluación	Interfaz Gráfica (GUI) Actual	Prototipo 1: Chatbot Básico (Basado en Reglas)	Prototipo 2: Agente RAG (Solo Informativo)	Prototipo 3: Agente Conversacional (RAG + Ejecución de Herramientas)
Curva de Aprendizaje	Alta	Media	Baja	Muy Baja
Capacidad Transaccional (Ejecutar acciones)	Completa	Muy Limitada	Nula	Completa
Acceso a Base de Conocimiento (RAG)	Nulo (Requiere consulta manual de guías)	Nulo (Respuestas preprogramadas)	Alta (Función principal)	Alta (Integrada)
Precisión en Interpretación de Intención	N/A	Baja	Media	Alta
Nivel de Automatización	Bajo	Bajo	Bajo (Solo en consultas)	Alto
Complejidad de Implementación	N/A (Línea base)	Baja	Media	Alta
Selección	Descartado (Problema)	Descartado (Insuficiente)	Descartado (Insuficiente)	Seleccionado

Como se observa en la tabla, el Prototipo 3 presenta la mayor complejidad de implementación, sin embargo, esta desventaja resulta aceptable en el contexto del proyecto, ya que es el único enfoque que permite ejecutar acciones reales sobre el ERP a partir de instrucciones en lenguaje natural, objetivo central de la propuesta. Si bien la automatización

transaccional con modelos de lenguaje introduce riesgos, como interpretaciones incorrectas o ejecuciones no deseadas, estos se controlan conceptualmente mediante confirmaciones explícitas, validaciones previas y control de acceso por roles.

En contraste, los prototipos basados únicamente en reglas o en consulta informativa se descartan por no cumplir el criterio de automatización operativa completa. Por estas razones, se selecciona el Prototipo 3 como el enfoque más adecuado para el desarrollo del proyecto.

### **2.2.3 Evaluación del prototipo**

La evaluación del prototipo se realizó de manera cualitativa mediante reuniones iterativas con desarrolladores del sistema ERP y el jefe de la empresa. Dado que el prototipado se limitó a la interfaz visual del chat como punto de interacción entre el usuario y el agente, no se aplicaron encuestas formales, priorizando la retroalimentación directa de los participantes, enfoque alineado con prácticas de evaluación de experiencia de usuario en sistemas ERP durante fases tempranas de diseño (Jawad & Villatoro Amaya, 2023).

Para medir la viabilidad del diseño, el análisis abarcó aspectos como la usabilidad general, el espacio visual de la ventana de chat, la legibilidad de los textos y la respuesta del sistema ante posibles fallos, cuidando siempre que la estética respetara la interfaz original del ERP. Asimismo, se evaluaron las interacciones dentro del chat, como el uso de mensajes predefinidos y mecanismos de confirmación explícita antes de ejecutar acciones sobre el sistema, en concordancia con principios de diseño y evaluación de la interacción humano-computadora (Gordon Graell, 2023).

Finalmente, tras debatir las opciones, el equipo de desarrollo y el cliente llegaron a un acuerdo unánime. La alternativa definitiva resultó ser aquella que lograba acoplarse de forma natural y fluida al entorno de trabajo diario, sin resultar invasiva. En el *Apéndice B*, se incluyen las vistas finales del chat.

## 2.3 Estrategia de orquestación e implementación del agente

### 2.3.1 Alternativas evaluadas:

Con el fin de definir el mecanismo de orquestación del agente de IA, se evaluaron distintas alternativas que permiten coordinar la comunicación entre el modelo de lenguaje, el MCP, el RAG y el ERP. La Tabla 3 presenta las opciones consideradas junto con sus principales ventajas y desventajas.

**Tabla 3**

*Comparación de alternativas de orquestación para el agente IA.*

Alternativa	Descripción	Ventajas	Desventajas
A1: Orquestador propio desde cero	Desarrollo completo de un servicio dedicado que gestionaría todas las conexiones entre las distintas partes del agente (ERP, LLM, etc).	Control total, trazabilidad completa, personalización total.	Mayor tiempo y costo de desarrollo, mantenimiento complejo.
A2: Orquestación con n8n	Uso de n8n como motor visual de orquestación con integración al MCP y RAG, mediante flujos configurables (n8n, 2025).	Desarrollo rápido, menor esfuerzo técnico, escalabilidad modular.	Dependencia de una herramienta externa, menor flexibilidad para personalizaciones profundas.

La alternativa A1 ofrece mayor control y personalización al desarrollarse completamente desde cero; sin embargo, implica un mayor tiempo de implementación y un costo de mantenimiento elevado. Por el contrario, la alternativa A2 prioriza la rapidez de desarrollo y el modularidad mediante una herramienta de orquestación existente, lo que la hace más adecuada para una fase inicial funcional del agente.

### 2.3.2 Criterios de decisión

Para seleccionar la alternativa más adecuada, se definieron criterios orientados tanto a la viabilidad técnica como al contexto del proyecto. La Tabla 4 compara cada alternativa en función de aspectos clave como tiempo de implementación, mantenibilidad, complejidad técnica y costo operativo.

**Tabla 4**

*Comparación de criterios de decisión para selección de alternativa.*

Criterio	A1	A2
Tiempo de implementación	Alto	Bajo
Mantenibilidad	Media	Alta
Escalabilidad	Alta	Media
Facilidad de auditoría	Alta	Media
Complejidad técnica	Alta	Baja
Costo operativo	Alto	Bajo

A partir de esta evaluación, se observa que la alternativa A2 presenta ventajas significativas en tiempo de implementación, complejidad técnica y costo operativo, mientras que A1 destaca principalmente en control y auditoría, lo que evidencia un contraste entre rapidez de adopción y nivel de control técnico.

### 2.3.3 Decisión

Se eligió la opción A2 (n8n como motor de orquestación) por ofrecer un balance adecuado entre flexibilidad y compatibilidad con MCP y RAG, agilizando significativamente el desarrollo sin necesidad de crear un orquestador propio y, a futuro, permite ampliar el agente hacia nuevas herramientas o canales, como servicios de mensajería o integraciones externas.

### 2.3.4 Implicaciones en el diseño

El uso de n8n llevó a adoptar una arquitectura híbrida donde esta herramienta define el flujo del agente e integra los servicios necesarios, gestionando interacciones con el LLM, el módulo de conocimiento (RAG), las herramientas del ERP vía MCP y la conexión con el frontend. Esta configuración simplificó la validación y facilitó la reutilización de componentes para futuras ampliaciones sin alterar la lógica central del agente (n8n, 2025).

### 2.3.5 Riesgos específicos y mitigación

La Tabla 5 resume los principales riesgos asociados a la implementación del agente de IA utilizando n8n, junto con su causa, impacto y medidas de mitigación propuestas.

**Tabla 5**

*Riesgos específicos y medidas de mitigación del agente IA con implementación de n8n.*

Riesgo	Causa	Impacto	Medida de mitigación
Escalabilidad restringida	Limitaciones de rendimiento de n8n en flujos concurrentes.	Degradación del rendimiento bajo alta demanda	Uso de contenedores con balanceo de carga y separación de flujos críticos
Dependencia de la herramienta de orquestación	n8n como punto central de integraciones	Fallos o incompatibilidades por cambios futuros de la herramienta.	Actualizaciones controladas, pruebas en entornos previos y monitoreo de la comunidad y documentación oficial

### 3. DISEÑO DE LA SOLUCIÓN

#### 3.1 Arquitectura

La arquitectura de la solución define la estructura del sistema, sus componentes principales y la forma en que interactúan para cumplir los objetivos del proyecto. La propuesta se basa en un agente conversacional integrado al ERP que utiliza un modelo LLM y un enfoque RAG para acceder a información corporativa, junto con MCP para la ejecución controlada de herramientas del ERP.

La orquestación se realiza mediante n8n, que coordina los flujos del agente y permite una integración modular, adaptable a futuros servicios o canales.

##### 3.1.1 Modelo de “4+1” Vistas de Kruchten

Para documentar la arquitectura se utiliza el modelo “4+1” de Kruchten que describe el sistema en cinco vistas complementarias, orientadas a las necesidades de distintos actores, como usuarios finales, desarrolladores y administradores de infraestructura. Las cuatro vistas principales (Lógica, Procesos, Desarrollo y Física) describen la estructura y el comportamiento del sistema, mientras que la vista de Casos de Uso (+1) funciona como eje central.

###### 3.1.1.1 Vista de Casos de Uso (+1)

Esta vista es el pilar central del modelo, describe la funcionalidad del sistema desde la perspectiva del usuario y valida que la arquitectura cumpla con los requerimientos funcionales definidos (RF-01 a RF-27). Los actores principales identificados se detallan a continuación y su interacción con el sistema se representa en la Figura 3:

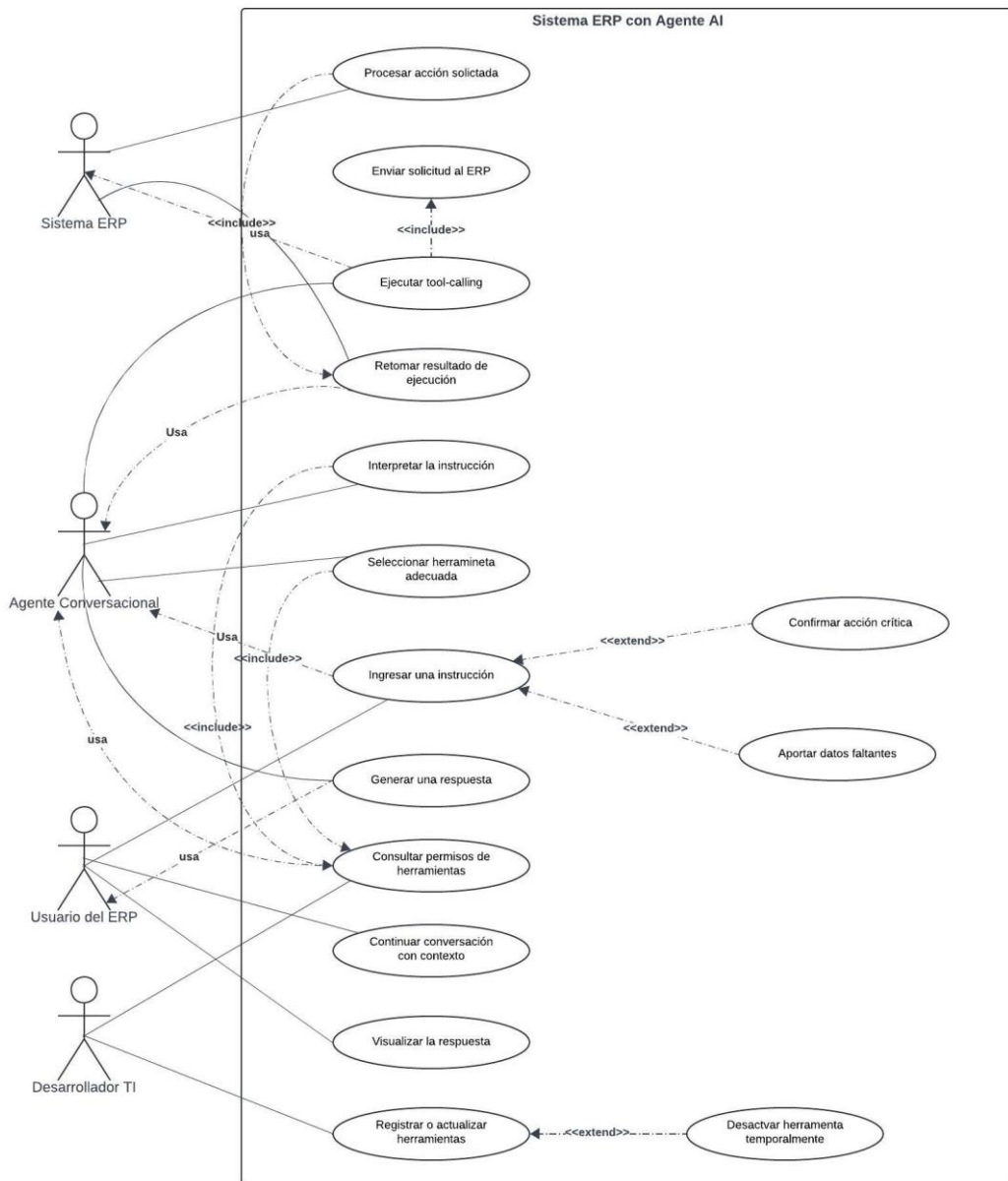
- **Usuario ERP:** El usuario final que interactúa con el agente.
- **Desarrollador (Equipo TI):** El personal técnico que mantiene y configura el agente.
- **Sistema ERP:** El sistema externo con el que el agente interactúa.

- **Base de Conocimiento:** El repositorio de documentos (p. ej., Google Drive) que el agente consulta.

**Diagrama de Casos de uso**

**Figura 3**

*Diagrama de Casos de Uso del Agente Conversacional Integrado con ERP*



Como se representa en el diagrama, el Usuario del ERP puede ingresar instrucciones, continuar la conversación con contexto y visualizar las respuestas generadas por el agente. El Agente Conversacional interpreta la instrucción, selecciona la herramienta adecuada, consulta

permisos y genera la respuesta. El Sistema ERP recibe solicitudes mediante tool-calls, ejecuta la acción requerida y retorna los resultados. Finalmente, el Desarrollador TI puede registrar, actualizar o desactivar herramientas según sea necesario. Las relaciones include y extend muestran los flujos opcionales y complementarios dentro del proceso.

### **3.1.1.2 Vista Lógica**

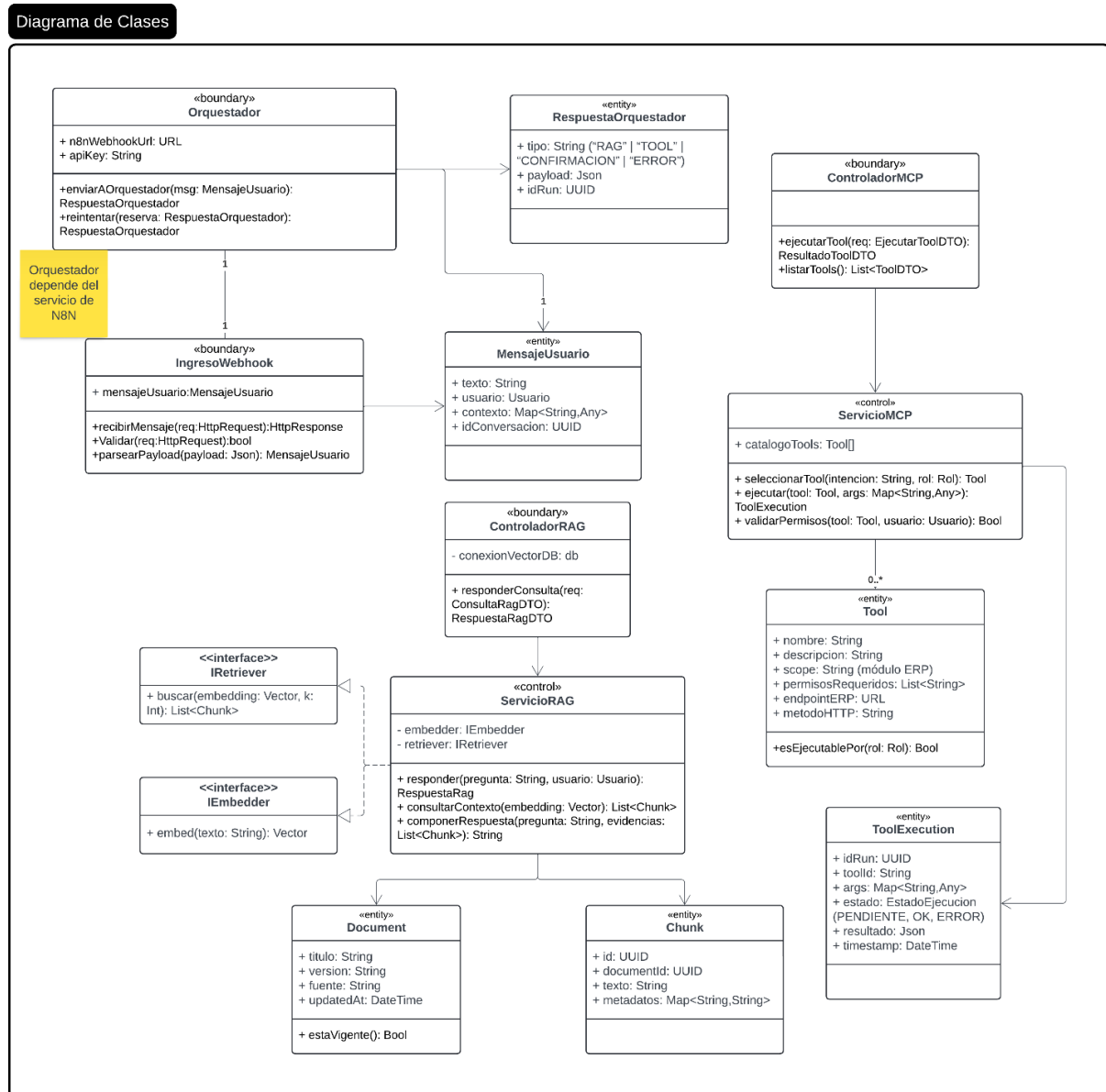
La vista lógica expone la estructura estática del sistema mediante clases, interfaces y sus relaciones, enfocándose en la funcionalidad que brinda a los usuarios.

#### **Diagrama de Clases**

El diagrama de clases, presentado en la Figura 4, muestra las entidades principales del dominio del agente junto con los servicios que gestionan la ejecución de acciones y las consultas RAG. También se incluyen los componentes de borde que permiten la interacción con la interfaz web y el orquestador n8n, con el objetivo de ofrecer una visión clara y modular del sistema.

Figura 4

Diagrama de Clases del Orquestador del Agente IA



### 3.1.1.3 Vista de Procesos

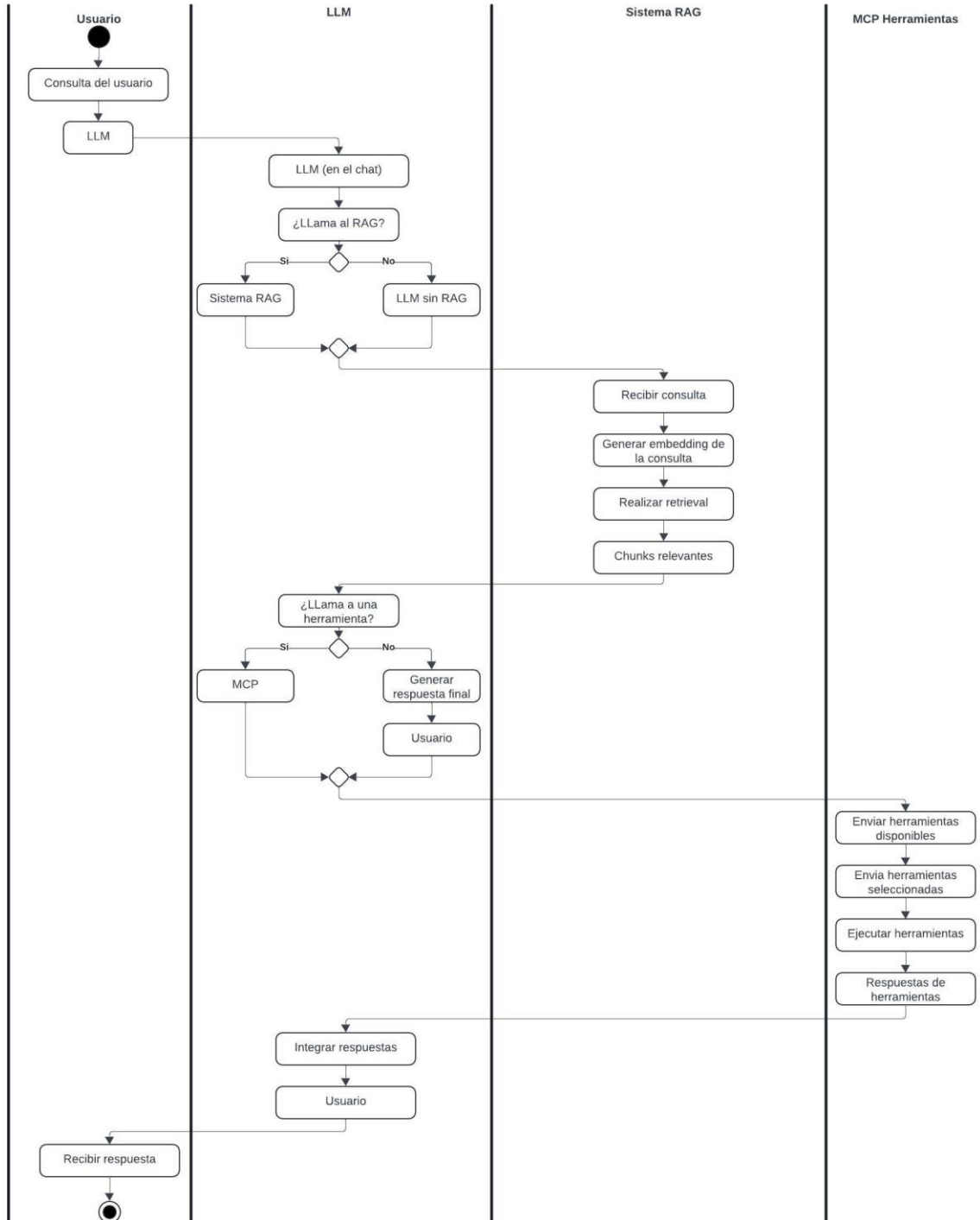
En esta vista se representa la dinámica del sistema, mostrando flujos de trabajo, concurrencia y comunicación entre procesos para explicar cómo interactúan los componentes en tiempo de ejecución.

### Diagrama de Actividad

La Figura 5 presenta el diagrama de actividad correspondiente al flujo principal del agente, el cual se activa cada vez que el usuario envía una consulta al sistema.

Figura 5

Diagrama de Actividad del flujo principal del agente



En función del análisis de la consulta, el sistema puede seguir uno de estos tres caminos:

- **Consulta de conocimiento:** se activa la consulta sobre un flujo de RAG donde la consulta se vectoriza y se realiza la búsqueda semántica sobre la base de conocimiento.

Como resultado, se devuelven los chunks (fragmentos) relevantes que aportan contexto al LLM para generar una respuesta basada en esos documentos.

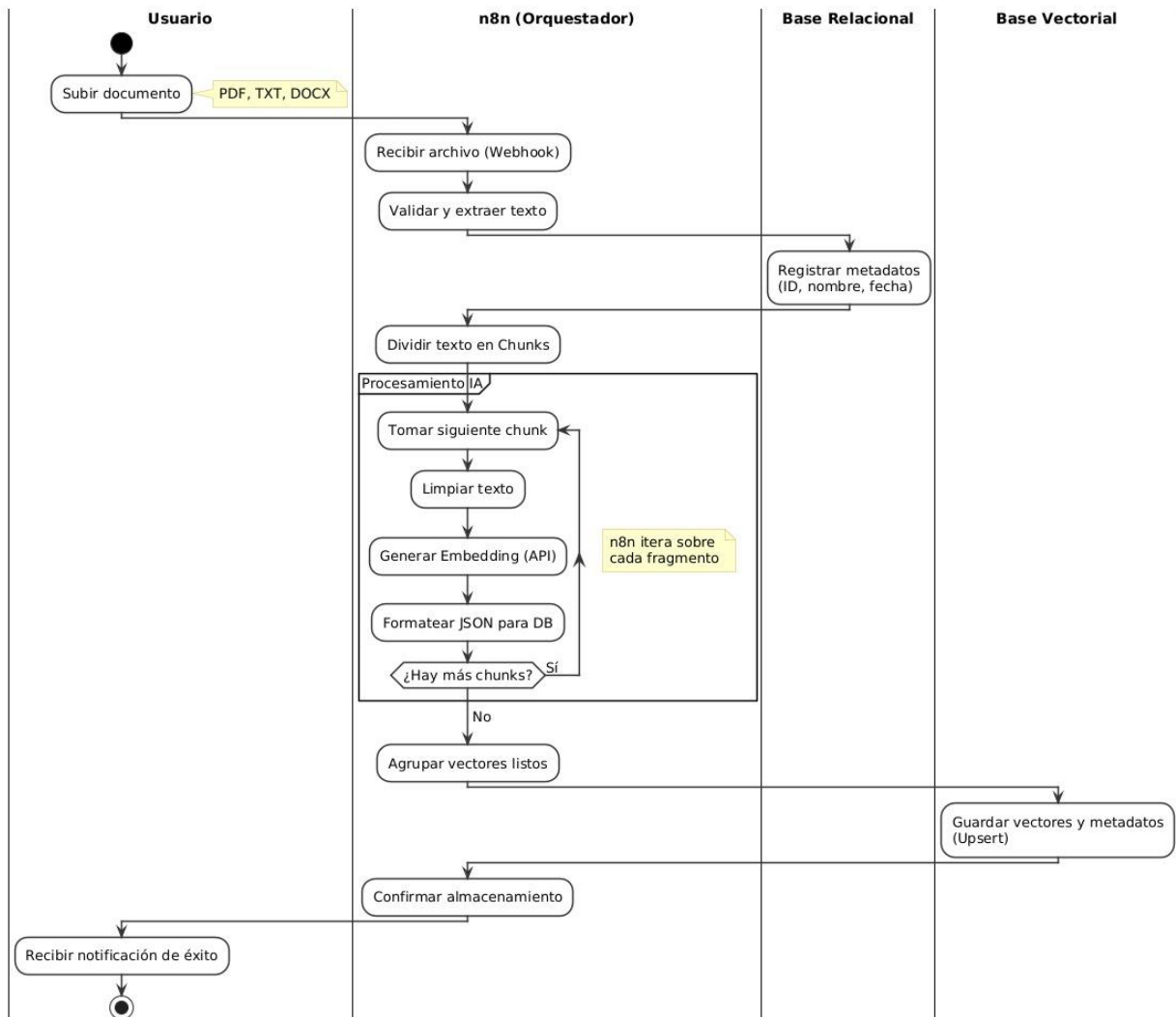
- **Orden de acción:** se ejecuta el flujo de herramientas, donde el modelo selecciona la acción adecuada y el MCP la ejecuta de forma controlada sobre el ERP.
- **Interacción conversacional simple:** El LLM genera la respuesta directamente.

Finalmente, en todos los casos, el LLM formula la respuesta final (generates answer) y la entrega al User.

Asimismo, la carga de información al sistema RAG se realiza mediante un proceso offline, independiente de la interacción del usuario, para preparar la base de conocimiento. La Figura 6 presenta el diagrama de actividad del flujo de subida de archivos para el RAG.

**Figura 6**

*Diagrama de Actividad del flujo subida de archivos para el RAG*



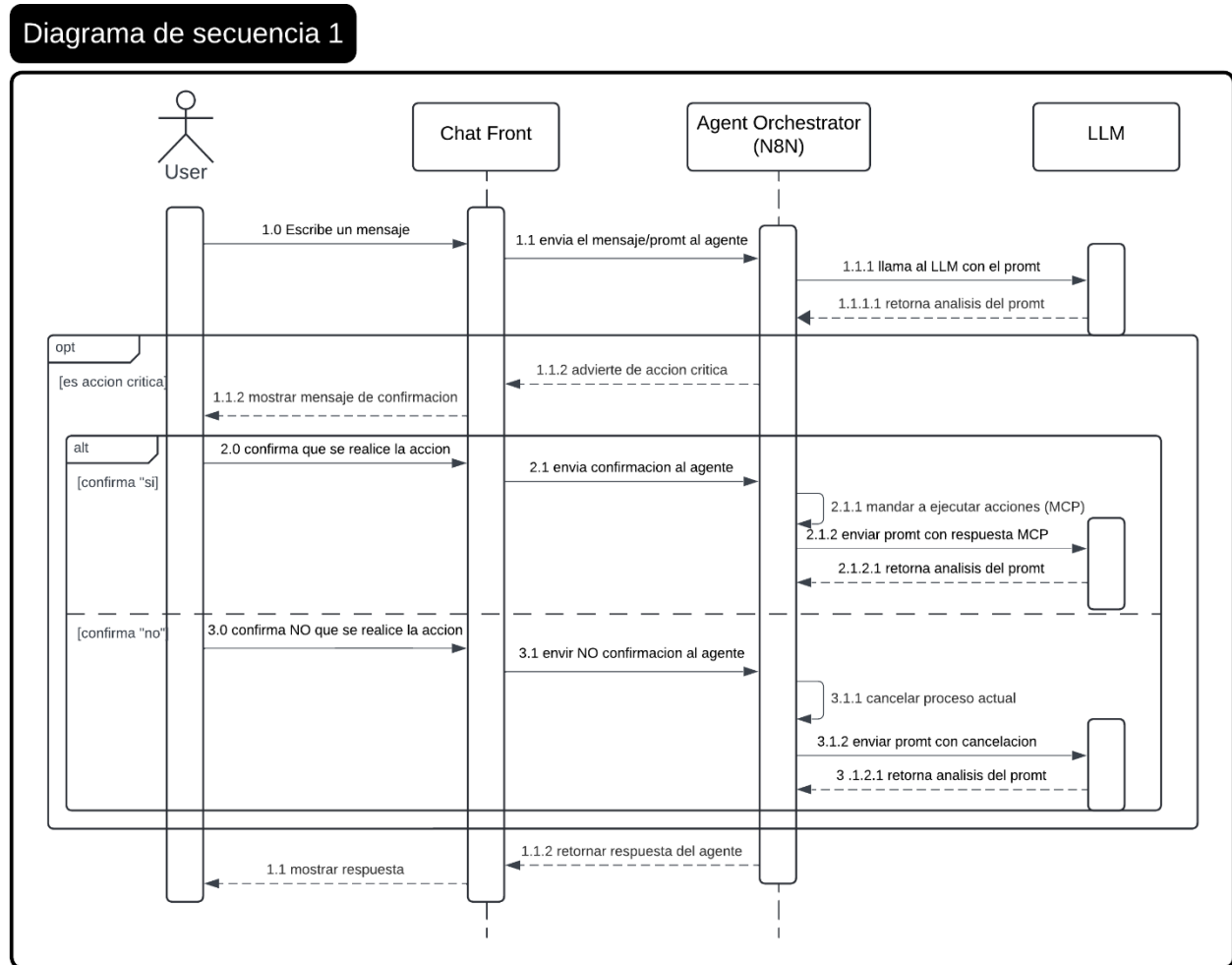
En este flujo, los documentos se cargan, se fragmentan, se convierten en vectores y se almacenan en la base de datos vectorial, junto con sus metadatos cuando corresponde

### Diagrama de Secuencia 1

El diagrama de la Figura 7 muestra el flujo de interacción cuando el agente detecta que una instrucción del usuario implica una acción crítica dentro del sistema. El orquestador solicita confirmación explícita a través del frontend, y dependiendo de la respuesta del usuario (“sí” o “no”), continúa con la ejecución de la herramienta correspondiente o se cancela el proceso.

**Figura 7**

*Diagrama de secuencia del flujo de interacción entre el usuario y el agente conversacional*

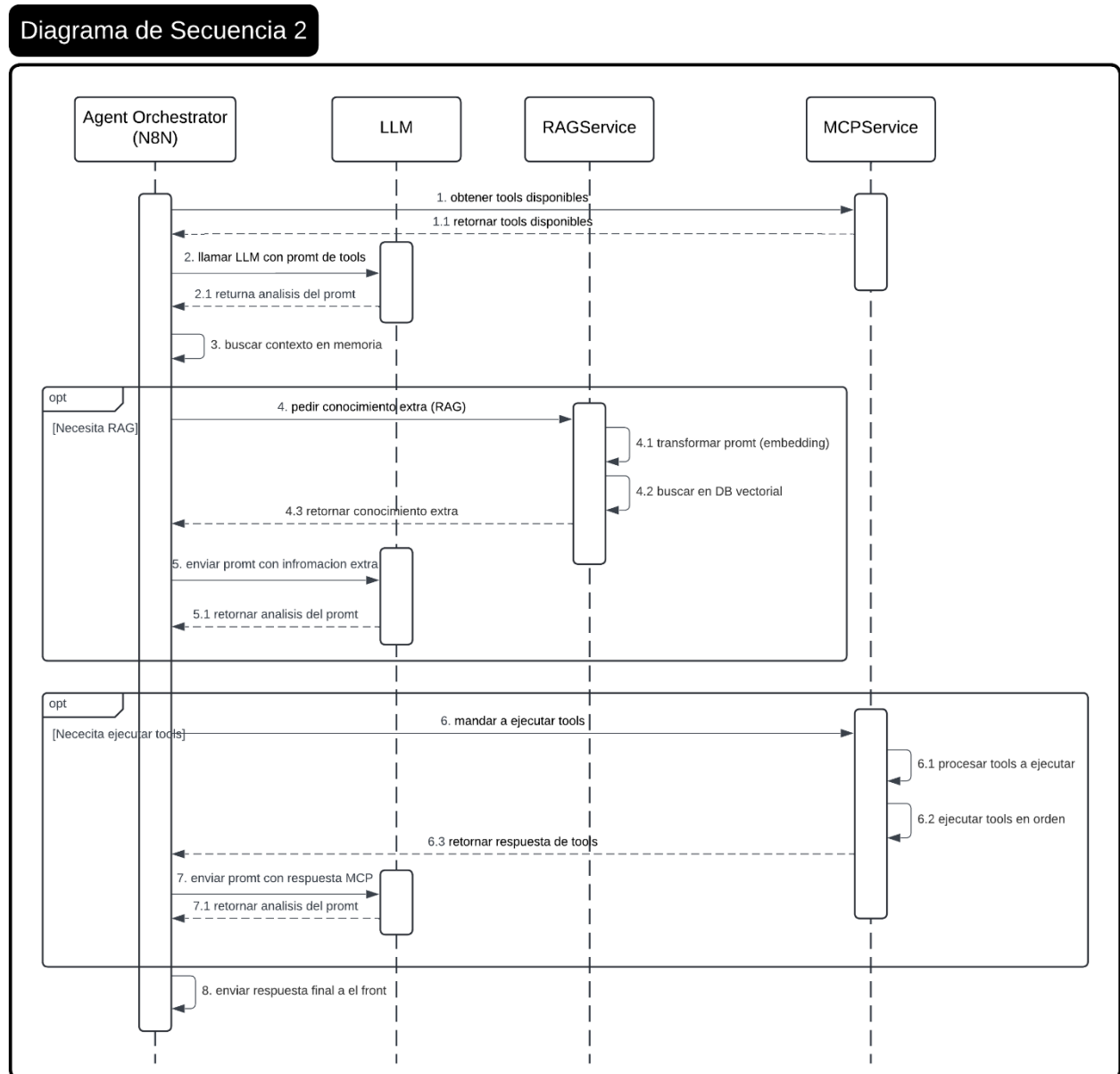


## Diagrama de Secuencia 2

La Figura 8 representa el flujo interno del agente al procesar una instrucción mostrando como se coordina la llamada al LLM para interpretar la intención, identifica si se requiere información adicional mediante RAG o la ejecución de una herramienta vía MCP, y envía la solicitud al componente correspondiente.

**Figura 8**

Diagrama de secuencia del flujo interno del agente: consulta RAG y ejecución de herramientas.



### 3.1.1.4 Vista de Desarrollo (Implementacion)

La vista de Desarrollo aborda la organización interna del software, incluyendo módulos, empaquetado y dependencias dentro del entorno de desarrollo.

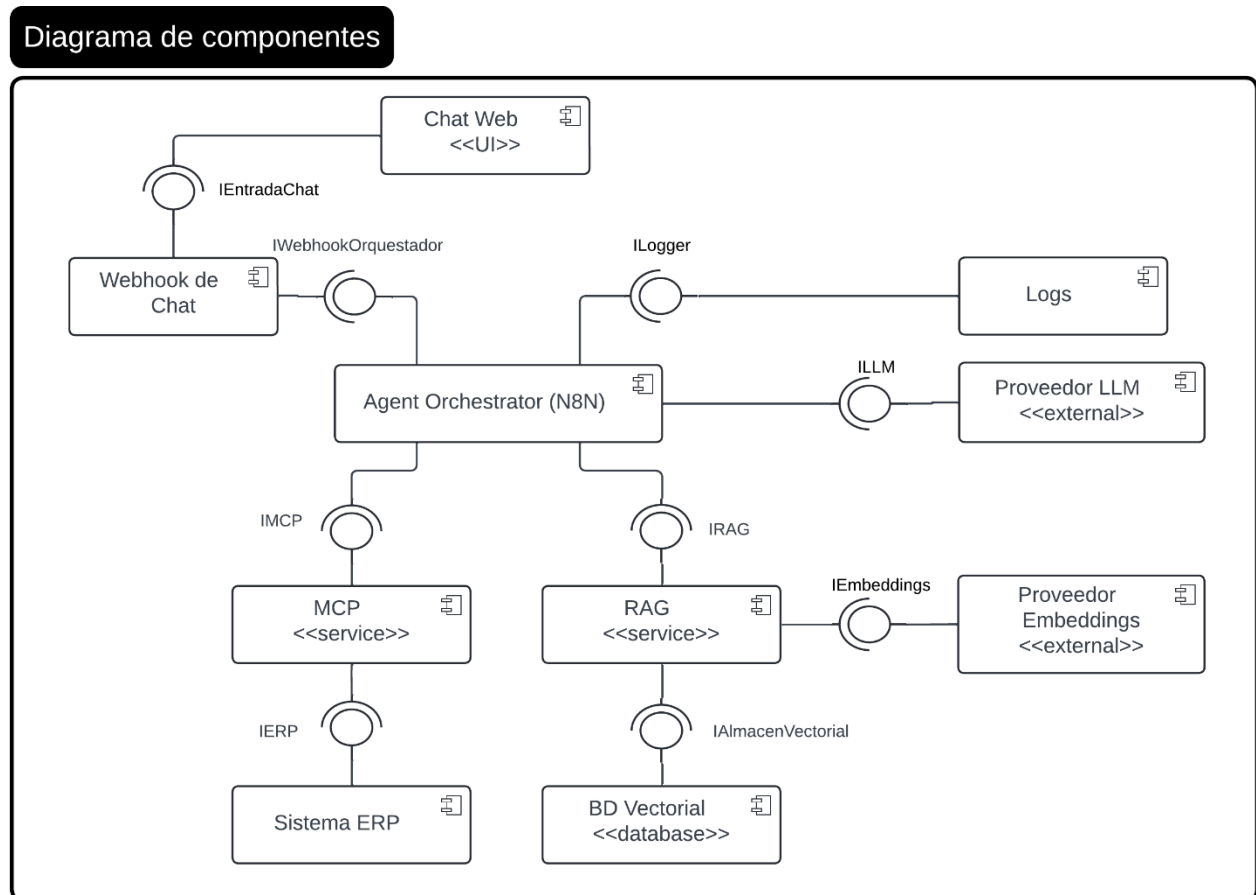
### Diagrama de Componentes

El diagrama de la Figura 9 muestra la estructura modular de la solución, destacando la interacción entre la interfaz de chat, el backend del agente, el orquestador implementado en n8n,

los servicios especializados de RAG y MCP y las integraciones externas, como el proveedor de LLM, la base de datos vectorial y el sistema ERP.

### Figura 9

Diagrama de componentes de la arquitectura del agente conversacional.



#### 3.1.1.5 Vista Física (Despliegue)

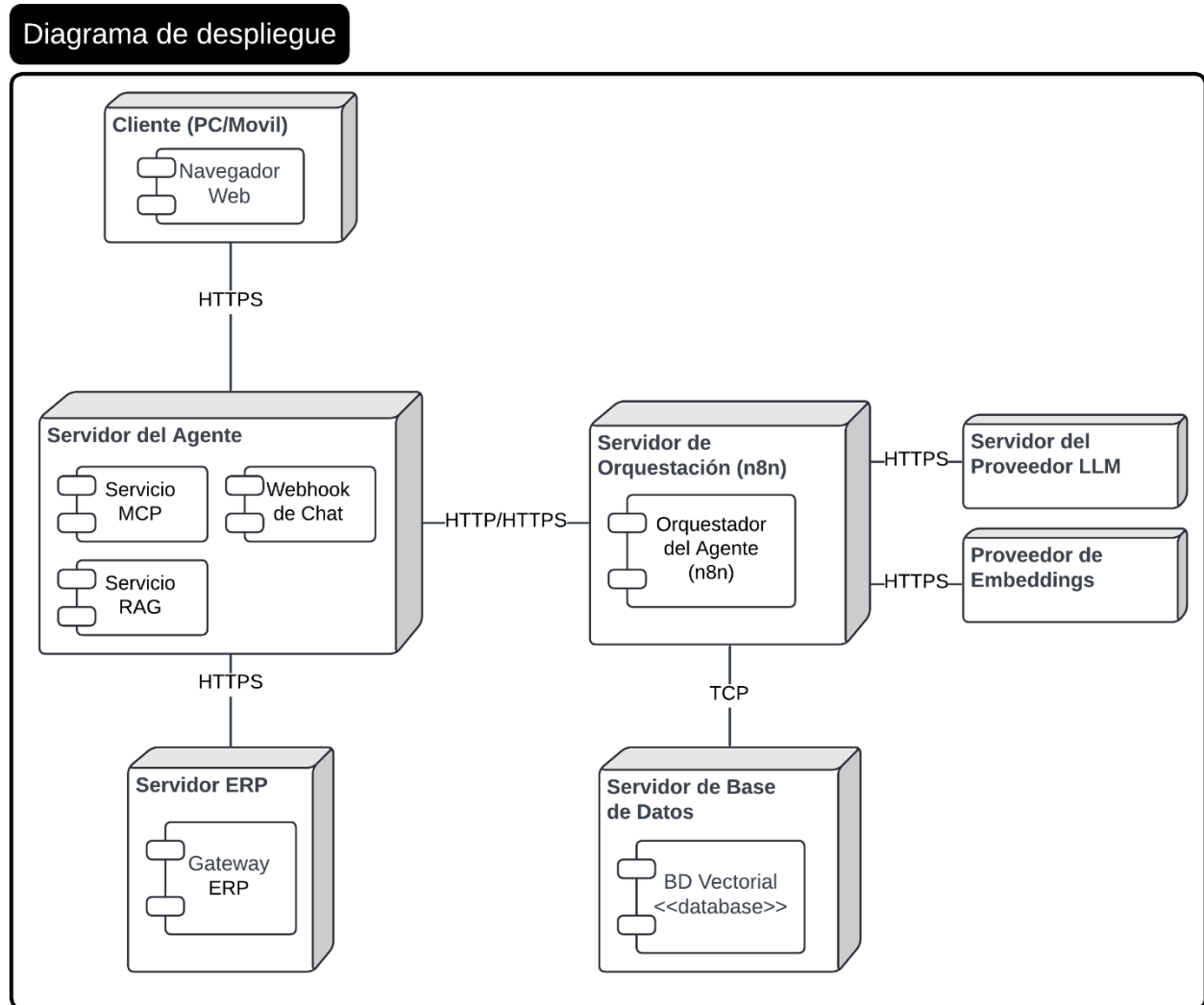
Esta vista presenta cómo se despliegan los componentes del sistema sobre la infraestructura física o virtual, detallando nodos, servidores y conexiones de red necesarias para su funcionamiento.

#### Diagrama de Despliegue

El diagrama de la Figura 10 presenta la distribución física del agente conversacional, mostrando los nodos donde se despliegan sus componentes principales y las conexiones con los servicios externos necesarios para su funcionamiento.

**Figura 10**

*Diagrama de despliegue del agente conversacional y servicios asociados.*



### 3.2 Diseño de la Base de Datos

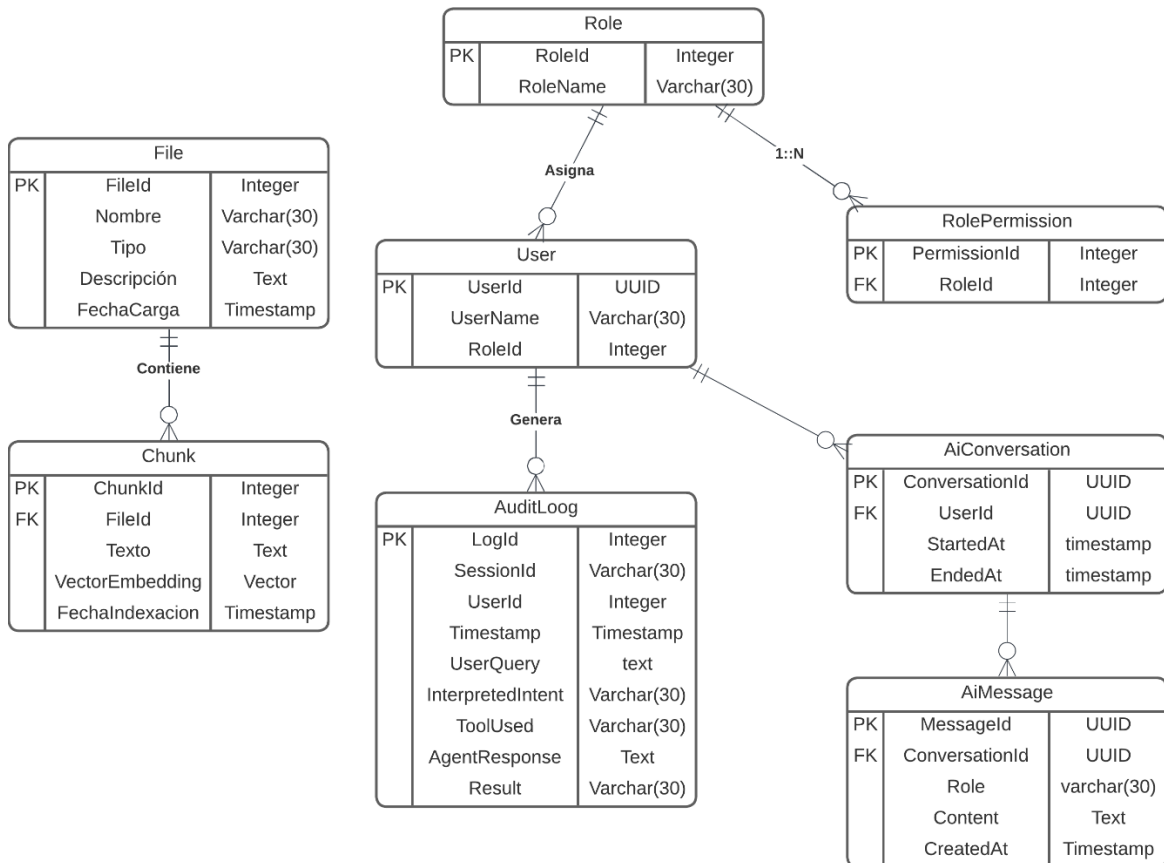
La Figura 11 muestra el modelo de datos utilizado por el agente conversacional, el cual organiza la gestión de documentos, fragmentos indexados, conversaciones y registros de auditoría generados durante las interacciones con el sistema. Este diseño permite organizar la información necesaria para el proceso de recuperación (RAG) y permite la trazabilidad de las acciones ejecutadas por el agente.

La base de datos de la solución incorpora las tablas File y Chunk, para el almacenamiento de documentos y embeddings, AiConversation y AiMessage para el registro de conversaciones y

AuditLog para el seguimiento de consultas, decisiones y resultados del agente. Las tablas User, Role y RolePermission se incluyen únicamente con fines de modelado, para ilustrar una posible integración con un sistema externo de gestión permisos sin ser parte de la solución desarrollada.

**Figura 11**

*Diagrama de base de datos del agente conversacional.*

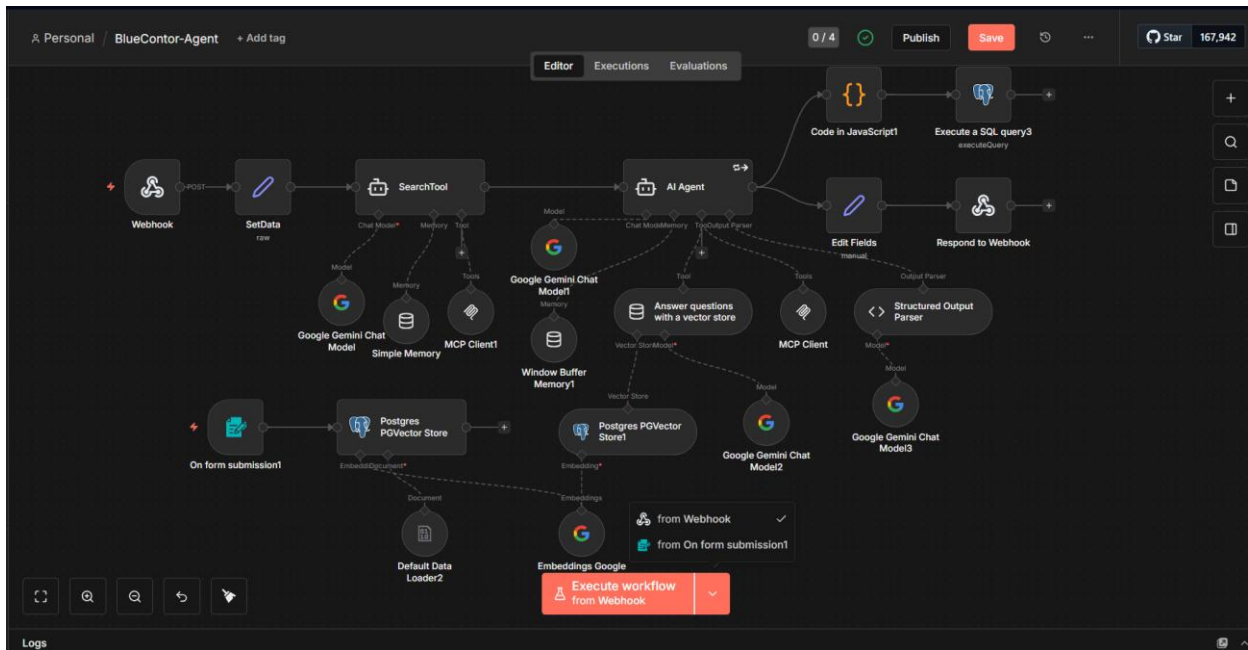


### 3.3 Flujo de Orquestación en n8n

La Figura 12 muestra el flujo de orquestación implementado en n8n, el cual coordina la ejecución del agente conversacional y el proceso de carga de documentos para el sistema RAG.

**Figura 12**

*Flujos de orquestación del agente conversacional y del sistema RAG en n8n*



El flujo del agente, activado mediante un webhook, procesa la consulta del usuario y permite al modelo decidir entre responder directamente, consultar el repositorio vectorial o ejecutar acciones controladas sobre el ERP, mientras que el flujo de carga de archivos se encarga de procesar documentos, fragmentarlos, vectorizarlos y almacenarlos en la base de datos para su uso en el sistema RAG, sin interferir con la operación del agente.

## **CAPÍTULO 3**



### 4.1.1.2 Sprint 1 – Diseño de la solución

En esta fase se abordó el diseño de la arquitectura del sistema, la definición de requerimientos, la elaboración de prototipos (Figma) y la validación conceptual de la solución, permitiendo establecer la arquitectura final del agente y los flujos principales de interacción. En la Figura 14 se puede ver las actividades de este Sprint.

**Figura 14**

*Cronograma Sprint 1 - Fase dos*

NÚMERO EDT	TÍTULO DE LA TAREA	RESPONSABLE DE LA TAREA	FECHA DE INICIO	FECHA DE ENTREGA	DURACIÓN (DIAS)	% COMPLETADO DE LA TAREA	FASE DOS																
							SEMANA 4 (29 Oct)				SEMANA 5 (27 Oct)				SEMANA 6 (03 Nov)				SEMANA 7 (10 Nov)				SEM 8
							20	21	22	23	24	27	28	29	30	31	3	4	5	6	7	10	11
M	T	W	T	F	M	T	W	T	F	M	T	W	T	F	M	T	W	T	F	M			
<b>2</b>	<b>Sprint 1</b>																						
2.1	Diseño de la arquitectura del agente-primera version	Jorge Mawyin, Kevin Roldán	20/10/25	27/10/25	8	100%																	
2.2	Reunión con cliente.	Jorge Mawyin, Kevin Roldán	24/10/25	24/10/25	1	100%																	
2.3	Confirmación de requerimientos y redacción de historias de usuario	Jorge Mawyin, Kevin Roldán	21/10/25	24/10/25	4	100%																	
2.4	Open API de endpoint seleccionados	Kevin Roldán	20/10/25	29/10/25	10	100%																	
2.5	Reunión con tutor para los requerimientos	Jorge Mawyin, Kevin Roldán	23/10/25	23/10/25	1	100%																	
2.6	Redacción de una planificación inicial de el desasrrollo del agente (pedida por el cliente)	Jorge Mawyin, Kevin Roldán	24/10/25	30/10/25	7	100%																	
2.7	Diseño de RAG (varias opciones de arquitectura)	Jorge Mawyin	27/10/25	31/10/25	5	100%																	
2.8	Registro/Diseño de herramientas/MCP Server	Kevin Roldán	27/10/25	07/11/25	12	80%																	
2.9	Reunion cliente (presentacion de RAG y MCP)	Kevin Roldán	30/10/25	30/10/25	1	100%																	
2.10	Elaboración del primer borrador del Capítulo 2	Jorge Mawyin, Kevin Roldán	28/10/25	05/11/25	9	100%																	
2.11	Análisis y refinamiento del Modelo 4+1 Kruchten	Jorge Mawyin, Kevin Roldán	06/11/25	10/11/25	5	100%																	
2.12	Desarrollo del segundo borrador de Capítulo 2	Jorge Mawyin, Kevin Roldán	07/11/25	10/11/25	4	100%																	
2.13	Pruebas de usuario de prototipo	Kevin Roldán	10/11/25	10/11/25	1	100%																	
2.14	Análisis de alternativas para el modelo AI del agente	Jorge Mawyin, Kevin Roldán	10/11/25	12/11/25	3	100%																	
2.15	Reunión con el cliente: revisión de avances y definiciones técnicas	Jorge Mawyin, Kevin Roldán	12/11/25	12/11/25	1	100%																	
2.15.1	Revisión del prototipo y pruebas de usuario	Jorge Mawyin	12/11/25	12/11/25	1	100%																	
2.15.2	Discusión de alternativas de modelos AI	Kevin Roldán	12/11/25	12/11/25	1	100%																	
2.16	Actualización del prototipo según retroalimentación del cliente	Jorge Mawyin	12/11/25	14/11/25	3	100%																	
2.17	Finalización del capítulo 2 y preparación de la presentación	Jorge Mawyin, Kevin Roldán	10/11/25	17/11/25	8	100%																	
2.18	Aprobación final del prototipo por el cliente	Jorge Mawyin, Kevin Roldán	17/11/25	17/11/25	1	100%																	

### 4.1.1.3 Sprint 2 – Desarrollo e integración

Por último, en la Figura 15 muestra como el Sprint 2 se enfocó en el desarrollo completo del sistema, tomando como base la arquitectura y los requerimientos previamente definidos. En esta etapa se implementaron el sistema RAG, el servidor MCP, la integración con n8n y la interfaz conversacional del agente en el ERP, además de pruebas y validaciones internas con ajustes finales.



## 4.2 Pruebas

Para la validación de la solución propuesta, se ejecutó una estrategia experimental en un entorno web de escritorio. Se realizaron pruebas con la participación de 7 usuarios, de los cuales 3 eran personal de soporte técnico, familiarizados con el ERP, y 4 eran usuarios que recién iniciaban en el uso del ERP.

El propósito principal fue verificar la capacidad del sistema para interpretar instrucciones en lenguaje natural y ejecutar operaciones en el ERP, recopilando métricas tanto de eficiencia operativa como de satisfacción del usuario. A continuación, la Tabla 6 detalla las cuatro dimensiones principales en las que se estructuró la evaluación, describiendo los objetivos y el alcance específico de cada tipo de prueba.

**Tabla 6**

*Estrategia de pruebas y validación del sistema.*

Tipo de Prueba	Objetivo Principal	Alcance y Detalles de la Validación
Pruebas Funcionales	Verificar la capacidad de respuesta del agente ante instrucciones complejas en escenarios de negocio reales.	<ul style="list-style-type: none"> <li>• <b>Consulta RAG:</b> Recuperación de políticas internas y documentación.</li> <li>• <b>Ejecución Transaccional:</b> Creación de registros (clientes) con validación de datos.</li> <li>• <b>Manejo de Ambigüedad:</b> Solicitud inteligente de información faltante al usuario.</li> <li>• <b>Seguridad:</b> Confirmación explícita antes de acciones críticas.</li> </ul>

Tipo de Prueba	Objetivo Principal	Alcance y Detalles de la Validación
Pruebas de Integración	Validar el flujo de datos y la comunicación correcta entre los componentes distribuidos de la arquitectura.	<ul style="list-style-type: none"> <li>• <b>Orquestación:</b> Procesamiento de solicitudes mediante flujos en n8n.</li> <li>• <b>Conexión MCP:</b> Ejecución correcta de herramientas y llamadas a las APIs del ERP.</li> <li>• <b>Auditoría:</b> Registro y trazabilidad de interacciones en el AuditLog para control.</li> </ul>
Pruebas de Usabilidad	Medir el desempeño del agente comparado con la interfaz gráfica y evaluar la satisfacción del usuario.	<ul style="list-style-type: none"> <li>• <b>Métricas Cuantitativas:</b> Medición de la reducción de tiempos de ejecución y tasa de éxito en tareas.</li> <li>• <b>Evaluación Cualitativa:</b> Aplicación de la escala SUS para medir aceptación y claridad.</li> </ul>

El diseño experimental detallado, incluyendo los guiones de los escenarios, los criterios de aceptación y los instrumentos de medición utilizados (formularios y encuestas), se encuentran documentados en el *Apéndice C*.

### 4.3 Resultados

En esta sección se presentan los hallazgos obtenidos tras la ejecución de los escenarios de prueba descritos anteriormente. El análisis se centra en contrastar el desempeño del Método A

(Interfaz Gráfica Tradicional) frente al Método B (Agente Conversacional), evaluando las métricas de eficiencia, eficacia y satisfacción recolectadas.

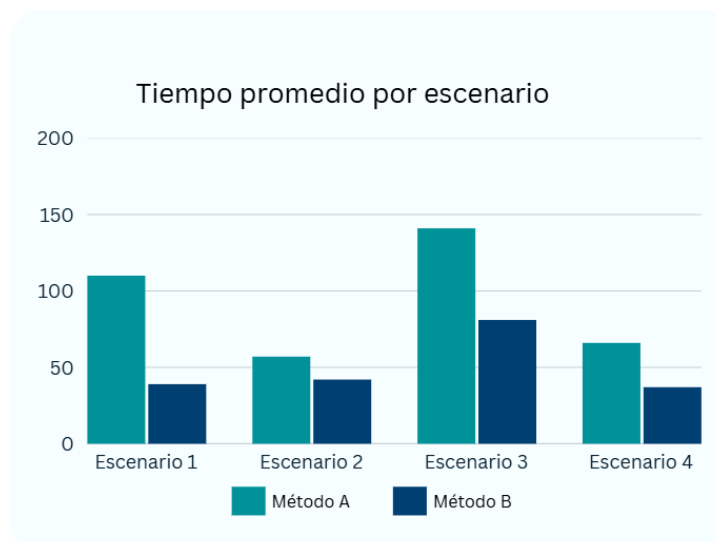
Asimismo, se examinan los datos desagregados por perfil de usuario para determinar el impacto específico de la herramienta en la curva de aprendizaje, comparando la respuesta del grupo experto frente a los usuarios iniciales.

#### 4.3.1 Resultados de desempeño y eficiencia operativa

Se compararon los tiempos promedio de ejecución para las cuatro tareas críticas. Como se observa en la Figura 16, el uso del agente conversacional generó una reducción de tiempos en todos los escenarios, siendo esta diferencia crítica en las tareas que involucran navegación compleja.

**Figura 16**

*Tiempo promedio de ejecución por tarea (Segundos)*



Los datos revelan reducciones significativas de tiempo gracias a la intervención del agente:

- Tarea 1 (Consulta RAG): Se obtuvo una reducción del 53% (de 82s a 39s).

Mientras que en el Método A los usuarios novatos navegaron incorrectamente por

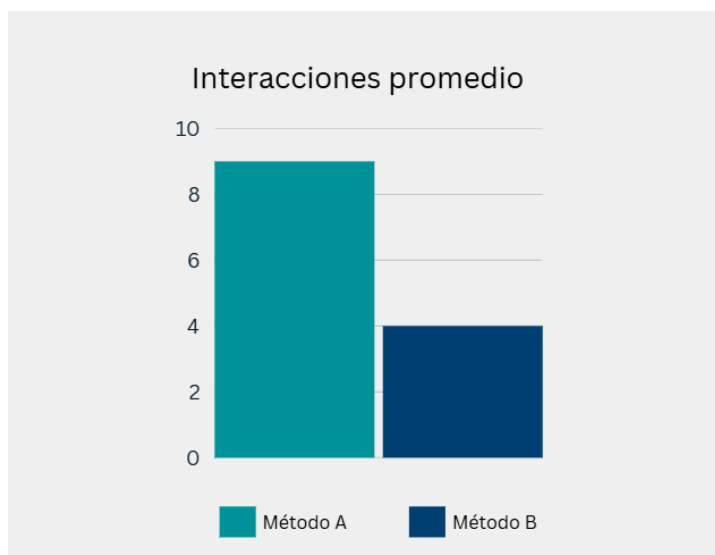
diferentes secciones de la plataforma antes de encontrar la sección correspondiente, en el Método B obtuvieron la respuesta directa.

- Tarea 3 (Manejo de Ambigüedad): Esta tarea presentó la mayor complejidad en la interfaz tradicional (112s promedio), provocando que usuarios novatos no lograran interpretar la sección en donde se realizaba dicho escenario. El agente redujo este tiempo a 81s, guiando al usuario mediante preguntas conversacionales para completar los datos faltantes.

Adicionalmente, se analizó el esfuerzo físico requerido. La Figura 17 ilustra la disminución en la cantidad de interacciones (clics y cambios de campo vs. mensajes enviados).

**Figura 17**

*Promedio de interacciones requeridas (Clics vs. Mensajes)*



El agente conversacional redujo la interacción física exactamente en un 52,5%, bajando de un promedio de 8.6 clics/campos en la interfaz tradicional a solo 4 mensajes/interacciones con el agente. Esto valida la hipótesis de que la interfaz conversacional disminuye la fatiga de navegación.

#### 4.3.2 Tasa de éxito y manejo de errores

Se observó un patrón distinto en los errores de cada método:

- Método A (Interfaz tradicional): Los fallos fueron de navegación. En las Tareas 1 y 3, usuarios novatos abandonaron porque no encontraban la ruta correcta en el menú. Un participante mostró frustración visible.
- Método B (Agente Conversacional): Los fallos fueron de contexto. En las Tareas 2 y 3, algunos usuarios tuvieron que reiniciar el chat porque el agente mezclaba instrucciones de turnos anteriores.

Con el Método A se identificó que el usuario nuevo queda bloqueado si no conoce la ruta. El Método B tiene sus propios problemas (la gestión de memoria del agente requiere mejoras), pero permitió que usuarios sin experiencia completaran procesos que en la GUI habían abandonado.

#### 4.3.3 Comparativa por perfil: Expertos vs. Nuevos Usuarios

El hallazgo más claro del estudio es el efecto del agente como ecualizador entre perfiles. La Figura 18 muestra la brecha de desempeño entre expertos y novatos.

#### Figura 18

*Curva de aprendizaje: Tiempos de ejecución según perfil de usuario*



- **Línea Naranja (Novatos):** Los usuarios tuvieron tiempo promedio >140s en tareas complejas. Con el agente (Método B) bajaron a ~50s, una reducción superior al 60%.
- **Línea Azul (Expertos):** Tiempos bajos y estables en ambos métodos. Aun así, varios usaron el agente como atajo para crear registros rápido.

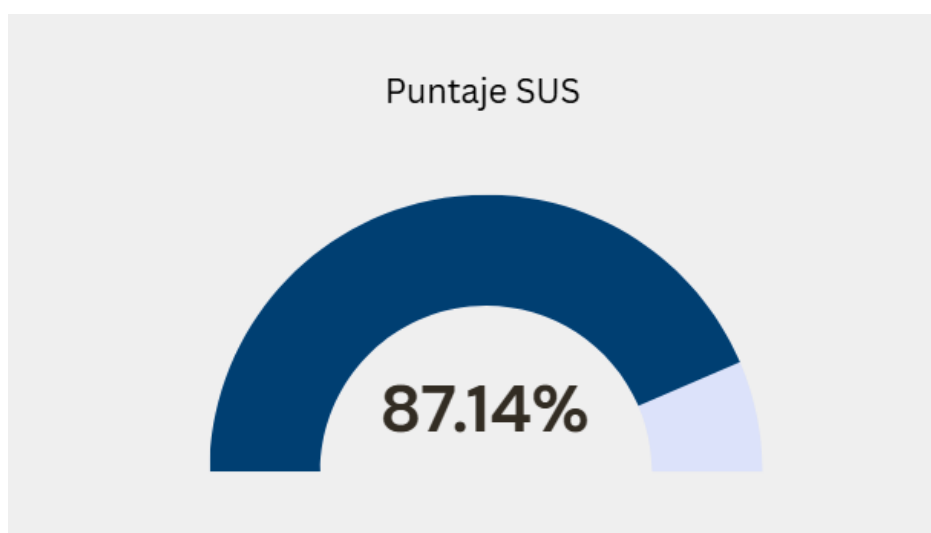
En la práctica, el agente elimina la barrera de "no saber dónde buscar". Un pasante con el agente opera a una velocidad cercana a la de un experto.

#### 4.3.4 Resultados de usabilidad y satisfacción

Para medir la percepción subjetiva se aplicó la escala SUS. El puntaje global fue de 87.14/100, como se observa en la Figura 19.

#### Figura 19

*Puntaje obtenido en la escala SUS*



Con 87.14, el sistema queda en categoría "Excelente" según SUS, por encima del promedio de la industria (68/100). Esto ocurre a pesar de los errores de contexto observados durante las pruebas: la experiencia general fue bien valorada.

#### 4.3.5 *Análisis cualitativo de la experiencia de usuario*

Se revisaron las respuestas abiertas del cuestionario final para complementar el puntaje numérico. Surgieron tres temas:

**Utilidad para Novatos:** Los participantes sin experiencia previa valoraron la capacidad de guía del agente. Uno comentó: "En todo fue útil. No conozco el sistema, así que no tenía idea de dónde buscar". Esto confirma el aporte del agente al onboarding de usuarios nuevos.

**Claridad y Confianza:** Si bien se destacó la claridad en las respuestas generadas, se identificó una preocupación recurrente respecto a la visibilidad de los resultados en acciones críticas. Un usuario sugirió la necesidad de "verificar que se había guardado correctamente en la Base... no es muy intuitivo solo con el mensaje de texto". Esto indica que, aunque el usuario confía en la instrucción dada, requiere una confirmación visual más robusta (feedback del sistema) para sentir seguridad total sobre la transacción realizada.

**Oportunidades de Mejora:** Los usuarios identificaron puntos técnicos específicos que, de ser implementados, mejorarían la fluidez de la interacción en futuras iteraciones:

1. Mejorar la redirección visual durante la ejecución de una acción, llevando al usuario paso a paso a través de las pantallas exactas en donde va ocurriendo el cambio para que el usuario aprenda como realizarlas.
2. Habilitar un historial de chat persistente que permita revisar instrucciones anteriores sin perder el hilo conversacional.
3. Solucionar la pérdida de contexto en conversaciones largas, lo cual obligaba a reiniciar el chat en ciertos escenarios complejos.

#### 4.4 **Análisis de Costos**

El análisis de costos se realizó con el fin de estimar los recursos económicos necesarios para la operación inicial del agente conversacional. Los valores presentados corresponden a un

escenario base de uso moderado y permiten evaluar la viabilidad económica del prototipo funcional.

#### 4.4.1 Escenario de cálculo

Para la estimación se consideró el siguiente escenario mensual:

- Sesiones de chat: 500
- Mensajes por sesión: 10

$$\text{Requests mensuales} = 500 \times 10 = 5\,000$$

- Tokens de entrada promedio por request: ~10 000
- Tokens de salida promedio por request: ~500

$$\text{Tokens de entrada} = 5\,000 \times 10\,000 = 50\,000\,000$$

$$\text{Tokens de salida} = 5\,000 \times 500 = 2\,500\,000$$

Para el sistema de Recuperación Aumentada (RAG) se asumió que solo una parte de las solicitudes requiere consulta al índice vectorial:

- Porcentaje de requests que activan RAG: 40 %

$$\text{Requests RAG} = 5\,000 \times 0.40 = 2\,000$$

- Tokens promedio por consulta de embedding: 500

$$\text{Tokens de embeddings} = 2\,000 \times 500 = 1\,000\,000$$

En esta fase se optó por el uso de APIs comerciales de modelos de lenguaje y embeddings, ya que este enfoque resulta significativamente más económico que desplegar modelos propios en infraestructura con GPU dedicada. La arquitectura del sistema permite ambos enfoques; sin embargo, un despliegue con GPU implicaría costos mayores sin afectar la lógica ni el funcionamiento del sistema.

#### 4.4.2 Resumen de costos estimados

La Tabla 7 presenta un resumen consolidado de los costos mensuales aproximados asociados a infraestructura, consumo de modelos y operación inicial del sistema. Los precios de infraestructura corresponden a planes públicos de DigitalOcean, seleccionados por requerimiento del cliente del presente proyecto.

**Tabla 7**

*Resumen de costos mensuales estimados del agente IA*

Concepto	Escenario considerado	Costo mensual estimado (USD)
Servidor n8n-MCP (DigitalOcean Droplet)	1 vCPU, 1 GB RAM	4.00
Base de datos PostgreSQL + pgvector	Tier básico (1 GB RAM)	15.15
Modelo de lenguaje (Google Gemini 2.5 Pro)	50M tokens entrada + 2.5M salida	87.50
Modelo de embeddings (gemini-embedding-001 - RAG)	1M tokens embeddings	0.15
Costo mensual total estimado	—	106.8 USD

El cálculo se basa en el modelo utilizado en el prototipo, con el fin de reflejar un escenario de costo real y reproducible. La comparación de precios con otros proveedores y modelos se presenta en el *Apéndice D*. Estos valores pueden variar dependiendo del proveedor, del modelo específico utilizado o del crecimiento en el volumen de uso.

## Capítulo 4

## 5. CONCLUSIONES Y RECOMENDACIONES

### 5.1 Conclusiones

A partir del desarrollo y evaluación del agente conversacional de inteligencia artificial integrado a un sistema ERP, se obtuvieron las siguientes conclusiones:

1. Se desarrolló un agente conversacional de IA capaz de interpretar instrucciones en lenguaje natural y convertirlas en operaciones dentro del ERP, cumpliendo con el objetivo de reducir la complejidad operativa y mejorar la interacción usuario sistema.
2. Se identificaron procesos del ERP susceptibles de automatización, definiendo acciones clave y parámetros necesarios; además, se evidenció que la falta de estandarización en endpoints (formatos de campos y ambigüedad entre funciones similares) influye directamente en la correcta interpretación y ejecución de acciones por parte del agente.
3. Se diseñó una arquitectura del agente que integra procesamiento de lenguaje natural con un mecanismo de invocación de operaciones del ERP; este diseño permitió combinar recuperación de información mediante RAG con la ejecución transaccional, habilitando al agente como asistente informativo y como operador de acciones.
4. Se prototipó e integró un agente conversacional capaz de interpretar instrucciones en lenguaje natural y ejecutar operaciones dentro del ERP, mostrando mayor aporte en usuarios con poca experiencia en ERPs al facilitar la localización y ejecución de procesos; en usuarios con experiencia, su valor se concentró principalmente en consultas generales y apoyo informativo.
5. La evaluación mediante pruebas evidenció una alta capacidad del agente para completar tareas y apoyar la operación; sin embargo, los tiempos de respuesta observados muestran que el rendimiento debe optimizarse para un uso productivo sostenido. Adicionalmente, bajo un escenario de uso moderado, el costo mensual estimado resulta manejable para

una empresa pequeña/mediana usando APIs comerciales de modelos de lenguaje y embeddings.

6. El proyecto demuestra la viabilidad de integrar agentes conversacionales en ERPs como una capa de apoyo, aportando un caso aplicado con potencial de mejora y evolución.

## 5.2 Recomendaciones

Con base a las limitaciones encontradas durante las validaciones del sistema con usuarios con experiencia y no experiencia en ERPs, además de las complicaciones encontradas durante el desarrollo del proyecto, se proponen las siguientes recomendaciones para futuros trabajos:

1. Se recomienda definir criterios uniformes para endpoints (fechas/horas/zonas), ya que durante la integración se observó que variaciones mínimas en formatos generan interpretaciones inconsistentes y errores difíciles de rastrear.
2. Es recomendable revisar y normalizar la nomenclatura y semántica de las herramientas disponibles para el agente, evitando funciones con nombres similares, pero comportamientos distintos que puedan inducir a selecciones incorrectas.
3. Dado que el rendimiento y la ejecución concurrente pueden convertirse en un cuello de botella, se recomienda migrar a “queue mode” con workers y límites de concurrencia y, si la carga crece, considerar un orquestador a medida; en este proyecto se priorizó simplicidad y rapidez de integración, pero bajo cargas pesadas el control fino de latencia/colas/estado requiere una configuración y operación más complejas.
4. Se recomienda fortalecer la capa MCP incorporando validación estricta de parámetros (tipos/formatos), manejo de errores más trazable y versionado de herramientas, ya que durante el desarrollo se observó que entradas ambiguas o fallos poco explícitos dificultan el diagnóstico y aumentan el riesgo de ejecuciones incorrectas.

5. Considerando que este es un campo en constante cambio, se recomienda mantener una arquitectura modular y desacoplada que facilite la incorporación de nuevas técnicas y mejoras sin reestructurar completamente el sistema.
6. Se recomienda ampliar y mantener formalmente la base documental del RAG, porque con actualización manual y pocos documentos se identificaron límites de cobertura y consistencia en respuestas informativas.
7. Como línea futura, se recomienda evaluar la integración del agente con aplicaciones de mensajería como WhatsApp, Telegram o WeChat, manteniendo los mismos mecanismos de seguridad, confirmación y auditoría.
8. Se recomienda realizar pruebas con modelos de lenguaje en infraestructura propia (GPU) y compararlos con APIs comerciales; durante el proyecto se eligieron APIs por velocidad de implementación, pero esa decisión limita el control directo sobre latencia, el costo a escala y la gobernanza de datos, aspectos que se vuelven críticos cuando el uso crece o existen requisitos de privacidad.

## 6. REFERENCIAS

- Abou Ali, M., Dornaika, F., & Charafeddine, J. (2025). Agentic AI: A comprehensive survey of architectures, applications, and future directions. *Artificial Intelligence Review*, 59(1), 11. <https://doi.org/10.1007/s10462-025-11422-4>
- Asfoura, E., Kassem, G., Alhuthaifi, B., & Belhaj, F. (2023). Developing Chatbot Conversational Systems & the Future Generation Enterprise Systems. *International Journal of Interactive Mobile Technologies (iJIM)*, 17(10), 155-175. <https://doi.org/10.3991/ijim.v17i10.37851>
- Asif, A., AlFrraj, D., & Alshamari, M. A. (2022). A Comprehensive Approach of Exploring Usability Problems in Enterprise Resource Planning Systems. *Applied Sciences*, 12(5), 2293. <https://doi.org/10.3390/app12052293>
- Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., ... Amodei, D. (2020). Language Models are Few-Shot Learners. *Advances in Neural Information Processing Systems*, 33, 1877-1901. [https://proceedings.neurips.cc/paper\\_files/paper/2020/hash/1457c0d6bfc4967418bfb8ac142f64a-Abstract.html](https://proceedings.neurips.cc/paper_files/paper/2020/hash/1457c0d6bfc4967418bfb8ac142f64a-Abstract.html)
- Castillo-Salamín, J. A. (2025). Modelo para evaluar competencias, preparación tecnológica y percepción de tendencias en logística y operaciones empresariales. *Revista Científica Orbis Cognitiona*, 9(2), 253-277. <https://doi.org/10.48204/j.orbis.v9n2.a7656>
- Gordon Graell, R. D. G. (2023). Interacción humano-computador y sus aportes en el desarrollo de la Informática Aplicada a la Educación. *Universidad y Sociedad*, 15(2), 110-119.
- Govea Souza, J. A. (2021). Sistema de planificación de recursos empresariales (ERP) y su influencia en los procesos de negocio de empresas distribuidoras de productos de

- consumo masivo en Lima Metropolitana en el 2019. *Industrial Data*, 24(1), 201-217.  
<https://doi.org/10.15381/idata.v24i1.19831>
- Haro, A. F., Martínez, E. J., Chango, T. S., Zambrano, T. P., & Zambrano, M. F. (2023). Enterprise resource planning (ERP) procesos para una implementación óptima y eficiente. *Prometeo Conocimiento Científico*, 3(1), e21-e21. (2021-).  
<https://doi.org/10.55204/pcc.v3i1.e21>
- Hernández, J. M. S., & Ávila, J. L. S. (2024). El uso de la inteligencia artificial en la gestión empresarial para la toma de decisiones en la planeación de desplazamiento del material de lento movimiento: The use of artificial intelligence in business management for decision-making in the planning of the movement of slow-moving material. *LATAM Revista Latinoamericana de Ciencias Sociales y Humanidades*, 5(4), 3338-3350.  
<https://doi.org/10.56712/latam.v5i4.2499>
- Hrishev, R., & Shakev, N. (2022). *ARTIFICIAL INTELLIGENCE IN ERP SYSTEMS*. [https://e-university.tu-sofia.bg/e-publ/files/11141\\_Artificial\\_intelligence\\_in\\_ERP\\_systems.pdf](https://e-university.tu-sofia.bg/e-publ/files/11141_Artificial_intelligence_in_ERP_systems.pdf)
- Jamui, A., Blessing, F., & Ogunrinde, V. (2025). *Natural Language Processing: A Comprehensive Survey*.
- Jawad, A., & Villatoro Amaya, M. (2023). *User experience (UX) design in ERP systems: Optimizing user interface*. <https://www.theseus.fi/handle/10024/816039>
- Junco, J. S. M. (2024). *DESARROLLO FRONTEND PARA VERSIÓN MÓVIL DE UN SISTEMA ERP*.
- Jurado, D. M. B. (2024). Las tecnologías de la información y la comunicación en los ERP para la gestión empresarial: Un análisis bibliométrico. *Ciencias Administrativas*, (25), 147.  
<https://doi.org/10.24215/23143738e147>
- Krishnan, N. (2025). *AI Agents: Evolution, Architecture, and Real-World Applications* (arXiv:2503.12687; Versión 1). arXiv. <https://doi.org/10.48550/arXiv.2503.12687>

- Mhaskey, S. V. (2024). Integration of Artificial Intelligence (AI) in Enterprise Resource Planning (ERP) Systems: Opportunities, Challenges, and Implications. *International Journal of Computer Engineering in Research Trends*, 11(12), 1-9.  
<https://doi.org/10.22362/ijcert/2024/v11/i12/v11i1201>
- Muro, J. A. R., Vera, R. J. J., González, W. W. B., & Jimbo, W. W. B. (2024). El impacto de la inteligencia artificial en tiempos modernos. *Sapiens in Artificial Intelligence*, 1(2), 1-14.  
<https://doi.org/10.71068/0xr55n04>
- n8n. (2025, diciembre 10). *Explore n8n Docs: Your Resource for Workflow Automation and Integrations* | n8n Docs. n8n Docs. <https://docs.n8n.io/>
- NSAIF, W., SALIH, H., Saleh, H., & AL-NUAIM, B. (2024). Conversational Agents: An Exploration into Chatbot Evolution, Architecture, and Important Techniques. *The Eurasia Proceedings of Science Technology Engineering and Mathematics*, 27, 246-262.  
<https://doi.org/10.55549/epstem.1518795>
- Pérez, A., Pertuz, V., Leiva, M., Vega, A., Pérez, A., Pertuz, V., Leiva, M., & Vega, A. (2019). Structural and Functional Elements of Multi-Agent Systems for the Organizational Cooperation in Medium-sized Companies. *Información tecnológica*, 30(4), 155-164.  
<https://doi.org/10.4067/S0718-07642019000400155>
- Pokala, P. (2024). Artificial Intelligence in Enterprise Resource Planning: A Systematic Review of Innovations, Applications, and Future Directions. *International Journal of Computer Applications & Information Technology*, 7, 1276-1289.  
<https://doi.org/10.5281/zenodo.14170247>
- Putri, M. R., Husodo, A. Y., & Irmawati, B. (2024). Simplification of Embedding Process in Retrieval Augmented Generation for Optimizing Question Answering Chatbot Model. *2024 IEEE International Conference on Communication, Networks and Satellite (COMNETSAT)*, 665-670. <https://doi.org/10.1109/COMNETSAT63286.2024.10862926>

- Salvatore Raieli & Gabriele Iuculano. (2025). *Building AI Agents with LLMs, RAG, and Knowledge Graphs: A practical guide to autonomous and modern AI agents*. Packt Publishing. <http://ieeexplore.ieee.org.espol.lookproxy.com/document/11099035>
- Schütte, R. (2024). The next generation of ERP systems: Problems of traditional ERP-Systems and the next wave of really standardized ERP-Systems. *Informing possible future worlds—essays in honour of Ulrich Frank. Logos, Berlin*, 427-452.
- Singh, A., Ehtesham, A., Kumar, S., & Khoei, T. T. (2025). *Agentic Retrieval-Augmented Generation: A Survey on Agentic RAG* (arXiv:2501.09136). arXiv. <https://doi.org/10.48550/arXiv.2501.09136>
- Supriyono, Wibawa, A. P., Suyono, & Kurniawan, F. (2024). Advancements in natural language processing: Implications, challenges, and future directions. *Telematics and Informatics Reports, 16*, 100173. <https://doi.org/10.1016/j.teler.2024.100173>
- Syvänen, S., & Valentini, C. (2020). Conversational agents in online organization–stakeholder interactions: A state-of-the-art analysis and implications for further research. *Journal of Communication Management, 24*(4), 339-362. <https://doi.org/10.1108/JCOM-11-2019-0145>
- Treviso, M., Lee, J.-U., Ji, T., Aken, B. van, Cao, Q., Ciosici, M. R., Hassid, M., Heafield, K., Hooker, S., Raffel, C., Martins, P. H., Martins, A. F. T., Forde, J. Z., Milder, P., Simpson, E., Slonim, N., Dodge, J., Strubell, E., Balasubramanian, N., ... Schwartz, R. (2023). Efficient Methods for Natural Language Processing: A Survey. *Transactions of the Association for Computational Linguistics, 11*, 826-860. [https://doi.org/10.1162/tacl\\_a\\_00577](https://doi.org/10.1162/tacl_a_00577)
- Zabala, R. M., Granja, L. G., Calderón, H. A., Velasteguí, L. E., Zabala, R. M., Granja, L. G., Calderón, H. A., & Velasteguí, L. E. (2021). Efecto en la gestión organizacional y la satisfacción de los usuarios de un sistema informático de planificación de recursos

empresariales (ERP) en Riobamba, Ecuador. *Información tecnológica*, 32(5), 101-110.

<https://doi.org/10.4067/S0718-07642021000500101>

## **7. APENDICES**

### **7.1 Apéndice A – Historias de Usuario y Requerimientos**

En esta sección se presenta la siguiente relación entre las Historias de Usuario (basadas en escenarios) y los Requerimientos Funcionales (RF) y No Funcionales (RNF) que rigen el comportamiento técnico del agente.

#### ***7.1.1 Especificación de Historias de Usuario***

Para el levantamiento de requerimientos, se levantaron historias de usuario para capturar las necesidades desde la perspectiva de los diferentes roles del sistema, estas se detallan en la siguiente Tabla 8.

**Tabla 8***Historias de Usuario con sus respectivos escenarios*

ID	Rol	Característica / Funcionalidad	Razón / Resultado	Escenario	Criterio de Aceptación (Título)	Contexto	Evento	Comportamiento esperado
U-001	Como un Usuario del ERP	Necesito enviar instrucciones en lenguaje natural.	Con la finalidad de ejecutar tareas sin navegar manualmen te por el sistema.	1	Interpretación correcta de comandos simples.	En caso que el usuario redacte una instrucción breve válida (ej. “Muéstrame las facturas pendientes”).	Cuando se envié el mensaje	El sistema interpreta correctamente la instrucción y muestra resultados.
				2	Validación de entrada	En caso que el usuario escriba en el campo del chat	Cuando se valide el mensaje	El sistema solo permite texto o expresiones reconocibles
				3	Sinónimos y reformulacion es	En caso que use frases diferentes con igual intención	Cuando se procese el mensaje	El sistema detecta la misma intención y responde adecuadamente

ID	Rol	Característica / Funcionalidad	Razón / Resultado	Escenario	Criterio de Aceptación (Título)	Contexto	Evento	Comportamiento esperado
				4	Errores ortográficos menores	En caso que existan errores de escritura	Cuando se detecte el texto	El sistema corrige o interpreta el mensaje sin fallar
U- 002	Como un Usuario del ERP	Necesito que sistema me aclare dudas o pida confirmación cuando la instrucción no sea clara o sea crítica	Con la finalidad de evitar errores y completar datos faltantes.	1	Solicitud de datos faltantes	En caso que la instrucción sea ambigua o incompleta / faltan parámetros	Cuando se analice el mensaje	El sistema solicita parámetros faltantes (p. ej., rango de fechas) antes de continuar
				2	Confirmación de acciones críticas	En caso que la instrucción afecte zonas críticas o datos sensibles	Cuando se detecte intención crítica	El sistema solicita confirmación explícita mediante botones ("Si"/"No")
U- 003	Como un Usuario del ERP	Necesito que el sistema recuerde el	Para poder mantener conversaciones	1	Referencias contextuales	En caso que el usuario haga referencia a algo reciente (“ese documento”)	Cuando se analice el texto	El sistema usa el contexto anterior para entender la instrucción

ID	Rol	Característica / Funcionalidad	Razón / Resultado	Escenario	Criterio de Aceptación (Título)	Contexto	Evento	Comportamiento esperado
		contexto reciente	continuas sin repetir información	2	Pérdida de contexto	En caso que la sesión expire o se cierre	Cuando se detecte sesión nueva	El sistema notifica que el contexto anterior se reinició
U-004	Como un Usuario del ERP	Necesito que, una vez interpretada mi instrucción, el sistema ejecute y muestre resultados de forma clara	Con la finalidad de visualizar rápidamente información o efectos de acciones	1	Respuesta informativa	En caso que la acción sea de consulta o de aclaración de dudas	Cuando se estructura la respuesta a enviar	El sistema muestra resumen claro y enlaces a secciones del sistema de ser necesario
				2	Confirmación de cambios realizados	En caso que la acción realice algún cambio en los datos de el sistema (crear/editar/eliminar)	Cuando se termine de procesar la acción	El sistema informa qué hizo; redirige o recarga datos según corresponda
				3	Cancelación segura	El usuario no confirma la acción	Al recibir “no” en las opciones de	El sistema cancela ejecución y registra el intento en logs

ID	Rol	Característica / Funcionalidad	Razón / Resultado	Escenario	Criterio de Aceptación (Título)	Contexto	Evento	Comportamiento esperado
							respuesta o timeout	
U-005	Como un Usuario del ERP	Necesito recibir mensajes claros y útiles del sistema	Con la finalidad de entender lo que ocurre y saber que hacer	1	Error de conexión	En caso que el sistema no responda o tarde mas de lo esperado	Cuando falle la llamada o supere el timeout configurad o	El sistema informa de la demora/desconexión, muestra “No se pudo conectar” y un botón Reintentar
				2	Error desconocido	En caso que ocurra una excepción no controlada	Cuando se capture el error	El sistema muestra mensaje genérico con código de referencia para soporte
				3	Mensajes de éxito	Se completa una acción correctamente	Al finalizar la ejecución	El sistema confirma con texto y datos resumidos del resultado

ID	Rol	Característica / Funcionalidad	Razón / Resultado	Escenario	Criterio de Aceptación (Título)	Contexto	Evento	Comportamiento esperado
				4	Indicador de procesamiento	En caso que el sistema esté trabajando/procesando una solicitud y la acción tarda más de 3 segundos	Cuando inicie el procesami ento	El sistema muestra “Procesando...” hasta respuesta o error
U- 006	Como un usuario del ERP	Necesito que el sistema identifique correctamente mi intención y contexto	Con la finalidad de convertir mi texto en acciones o respuestas confiables	1	Detección de intención	En caso que se reciba una instrucción en lenguaje natural	Cuando se analice el mensaje	El sistema identifica la intención principal y el contexto del usuario (rol, contexto previo, etc.)
				2	Elección de acción o herramienta adecuada	En caso de que la intención identificada puede resolverse de distintas formas dado varias herramientas candidatas o	Cuando se decida como actuar y comparen metadatos (entradas,	El sistema selecciona la opción más apropiada (respuesta directa o ejecución de herramienta) según intención, entradas y permisos

ID	Rol	Característica / Funcionalidad	Razón / Resultado	Escenario	Criterio de Aceptación (Título)	Contexto	Evento	Comportamiento esperado
						restricciones de permisos	permisos, intención)	
U- 007	Como un usuario del ERP	Necesito que el sistema me responda con información de la base de conocimiento cuando sea necesario	Con la finalidad de obtener conocimien to actualizado y relevante	1	Respuesta basada en documentos relevantes (RAG)	En caso que no haya herramienta adecuada / se requiera contexto documental	Cuando se consulte la base de conocimie nto	El sistema responde con información relevantes recuperadas para enriquecer la respuesta
				2	Referencia de origen	En caso que la respuesta provenga de documentos guías consultados del ERP	Cuando se muestre la respuesta	El sistema incluye enlaces de referencia a la sección de las guías.
T- 101	Como desarroll ador del equipo de TI	Puedo ajustar qué herramientas puede usar el sistema mediante	Con la finalidad de controlar qué puede ejecutarse	1	Actualizar definición de herramienta	En caso que se incorpore, cambie o retire una capacidad / proceso sobre el ERP	Cuando se modifique la configurac ión/código	El sistema considera la definición actualizada y la pone disponible para su uso tras la actualización

ID	Rol	Característica / Funcionalidad	Razón / Resultado	Escenario	Criterio de Aceptación (Título)	Contexto	Evento	Comportamiento esperado
		cambios de configuración/c ódigo	de forma segura				correspon diente	
				2	Control por roles	En caso que un rol no deba ejecutar cierta herramienta	Cuando se resuelva autorizaci ón	El sistema restringe u otorga acceso de herramientas según rol del usuario
				3	Publicación segura	En caso que haya reemplazo o ajuste de una herramienta	Cuando se ejecute el proceso de actualizaci ón	El sistema aplica el cambio sin afectar ejecuciones en curso y registra la modificación
				4	Desactivación temporal	En caso que una herramienta del ERP necesite mantenimiento o no deba ejecutarse temporalmente.	Cuando el desarrolla dor modifique su estado en la configurac	El sistema debería permitir marcar la tool como “inactiva” sin borrarla del registro

ID	Rol	Característica / Funcionalidad	Razón / Resultado	Escenario	Criterio de Aceptación (Título)	Contexto	Evento	Comportamiento esperado
							ión del agente o marque la tool como “inactiva”.	
T- 102	Como desarrollador del equipo de TI	Necesito contar con un registro consultable de acciones/procesos realizados por el sistema	Con la finalidad de auditar y diagnostica r sin exponer datos sensibles	1	Almacenamiento estructurado de eventos	En caso que ocurra una acción o proceso	Cuando se ejecute dicha acción o proceso	El sistema registra eventos con metadatos (usuario/rol, intención, herramienta, parámetros no sensibles, resultado, timestamp)
T- 103	Como desarrollador del	Necesito mantener estable el funcionamiento	Con la finalidad de garantizar	1	Optimización de la recuperación de	En caso de que se agreguen, eliminen o actualicen documentos	Cuando se realice el proceso de	El sistema actualiza el RAG con los nuevos vectores o índices y mejora la

ID	Rol	Característica / Funcionalidad	Razón / Resultado	Escenario	Criterio de Aceptación (Título)	Contexto	Evento	Comportamiento esperado
Historia	equipo de TI	del agente y sus módulos de memoria	el funcionamiento continuo del agente y prevenir degradaciones en el rendimiento	2	conocimiento (RAG)  Supervisión del flujo y rendimiento del agente	en la base de conocimiento.	actualización.	precisión de las respuestas, manteniendo disponible el flujo conversacional.
			o				Cuando se revise el monitoreo de los procesos internos del agente.	El sistema permanece activo mientras se permite recopilar y consultar métricas de desempeño desde las herramientas de observación, sin desarrollo de nuevas pantallas o módulos.

### 7.1.2 Matriz de Relación: Escenarios vs. Requerimientos

En la Tabla 9 se detalla que requerimientos se conectan a que escenarios planteados en las historias de usuario

**Tabla 9**

*Relación de historias de usuario con requerimientos*

ID Historia	Escenarios de Aceptación	Requerimientos Funcionales Relacionados	Requerimientos No Funcionales
U-001	Interpretación, Validación, Sinónimos y Errores	RF-01, RF-02, RF-03	RNF-5
U-002	Datos faltantes y Acciones críticas	RF-04, RF-05	RNF-3
U-003	Referencias contextuales y Pérdida de contexto	RF-06, RF-07	RNF-4
U-004	Respuesta informativa, Confirmación de cambios y Cancelación	RF-08, RF-09, RF-10, RF-11	RNF-4
U-005	Errores de conexión, Desconocidos, Éxito e Indicadores	RF-12, RF-13, RF-14	RNF-1, RNF-2, RNF-3
U-006	Detección de intención y Elección de herramienta	RF-15, RF-16, RF-20	RNF-6

ID Historia	Escenarios de Aceptación	Requerimientos Funcionales Relacionados	Requerimientos No Funcionales
U-007	Respuestas RAG y Referencia de origen	RF-17, RF-18, RF-19	RNF-5
T-101	Gestión, Roles, Publicación segura e Inactivación	RF-20, RF-21, RF-22	RNF-9, RNF-10, RNF-11
T-102	Almacenamiento estructurado de eventos	RF-23, RF-24	RNF-7, RNF-8
T-103	Optimización RAG y Supervisión de rendimiento	RF-25	RNF-1

### ***7.1.3 Definición Detallada de Requerimientos***

Una vez establecida la relación con los escenarios de usuario, se listan los requerimientos técnicos definitivos que rigen el sistema:

#### **7.1.3.1 Requerimientos Funcionales**

##### **Interacción y Procesamiento de Lenguaje Natural**

- **RF-01:** El sistema debe permitir al usuario ingresar instrucciones mediante lenguaje natural.
- **RF-02:** El sistema debe interpretar la intención del usuario a partir de comandos simples, incluso si se usan sinónimos o existen errores ortográficos menores.
- **RF-03:** El sistema debe validar que la entrada del usuario sea texto o expresiones reconocibles.

- **RF-04:** El sistema debe solicitar al usuario los parámetros faltantes (p. ej., rango de fechas) si una instrucción es ambigua o incompleta.
- **RF-05:** El sistema debe solicitar confirmación explícita al usuario (p. ej., mediante botones "Sí"/"No") antes de ejecutar acciones críticas o que afecten datos sensibles.
- **RF-06:** El sistema debe mantener el contexto de la conversación reciente para comprender referencias (p. ej., "ese documento").
- **RF-07:** El sistema debe notificar al usuario si la sesión expira y el contexto de la conversación se ha reiniciado.

### **Ejecución de Acciones y Retroalimentación al Usuario**

- **RF-08:** El sistema debe ejecutar la acción correspondiente una vez interpretada la intención del usuario.
- **RF-09:** El sistema debe mostrar un resumen claro de los resultados en acciones de consulta, incluyendo enlaces a secciones relevantes del ERP si es necesario.
- **RF-10:** El sistema debe informar al usuario sobre la correcta finalización de acciones que modifiquen datos (crear, editar, eliminar).
- **RF-11:** El sistema debe cancelar una ejecución si el usuario no la confirma explícitamente o si la solicitud expira (timeout).
- **RF-12:** El sistema debe mostrar un indicador visual (p. ej., "Procesando...")mientras espera la respuesta a una solicitud.
- **RF-13:** El sistema debe informar al usuario sobre errores de conexión o demoras (timeout) y proveer una opción para reintentar la acción.
- **RF-14:** El sistema debe mostrar un mensaje de error genérico junto con un código de referencia para soporte técnico en caso de excepciones no controladas.

### **Selección de Herramientas y Consulta de Conocimiento (RAG)**

- **RF-15:** El sistema debe identificar la intención principal del usuario y su contexto (rol, contexto previo).
- **RF-16:** El sistema debe seleccionar la acción o herramienta más apropiada basándose en la intención, los datos de entrada y los permisos del rol del usuario.
- **RF-17:** El sistema debe consultar la base de conocimiento (RAG) para responder preguntas cuando no exista una herramienta de acción adecuada o se requiera contexto documental.
- **RF-18:** El sistema debe enriquecer las respuestas con información recuperada de documentos relevantes de la base de conocimiento.
- **RF-19:** El sistema debe incluir enlaces de referencia a los documentos de origen (guías del ERP) cuando una respuesta provenga de la base de conocimiento.

#### **Administración, Seguridad y Mantenimiento**

- **RF-20:** El sistema debe restringir u otorgar el acceso a la ejecución de herramientas según el rol del usuario que realiza la solicitud.
- **RF-21:** El sistema debe permitir marcar una herramienta como "inactiva" temporalmente (p. ej., por mantenimiento) sin necesidad de eliminar su definición.
- **RF-22:** El sistema debe aplicar las actualizaciones de herramientas sin afectar las ejecuciones que estén en curso en ese momento.
- **RF-23:** El sistema debe registrar eventos de auditoría en un log estructurado, incluyendo metadatos como usuario, rol, intención, herramienta usada, resultado y timestamp.
- **RF-24:** El sistema debe asegurar que los registros de auditoría no expongan datos sensibles del usuario.
- **RF-25:** El sistema debe permitir la actualización de la base de conocimiento (RAG) (vectores e índices) cuando los documentos de origen sean modificados.


### 7.1.3.2 Requerimientos No-Funcionales

- **RNF-1: Rendimiento de Respuesta:** El sistema debe responder a consultas simples en un tiempo promedio menor o igual a 10 segundos.
- **RNF-2: Indicador de Procesamiento:** El sistema debe mostrar un indicador de actividad en menos de 2 segundos después de recibir una solicitud.
- **RNF-3: Manejo de Errores:** El sistema debe mostrar mensajes claros y no técnicos cuando ocurran fallos internos o del ERP.
- **RNF-4: Consistencia de Respuestas:** El sistema debe generar respuestas coherentes cuando se hagan preguntas iguales en condiciones similares.
- **RNF-5: Usabilidad y Claridad del Lenguaje:** Las respuestas del agente deben estar redactadas en lenguaje natural, sin tecnicismos innecesarios.
- **RNF-6: Respeto de Roles y Permisos:** El sistema debe ejecutar únicamente las acciones permitidas según el rol del usuario que realiza la solicitud.
- **RNF-7: Registro Seguro de Auditoría:** El sistema debe registrar eventos esenciales (usuario, rol, intención, herramienta utilizada, resultado, timestamp) en un log estructurado.
- **RNF-8: Protección de Datos en Auditoría:** El sistema debe asegurar que los registros de auditoría no expongan datos sensibles del usuario.
- **RNF-9: Mantenibilidad y Modularidad:** El sistema debe estructurarse en componentes independientes (interpretación, herramientas, ejecución) para facilitar modificaciones futuras.
- **RNF-10: Actualización Segura de Herramientas:** El sistema debe permitir actualizar herramientas sin afectar ejecuciones que estén en curso.
- **RNF-11: Inactivación Temporal de Herramientas:** El sistema debe permitir marcar una herramienta como inactiva temporalmente sin eliminar su definición.

## 7.2 Apéndice B – Pantallas del Chat del agente




The image displays two sequential screenshots of an agent chat interface. The left screenshot shows the initial chat screen with a welcome message and suggested actions. The right screenshot shows the chat flow for creating a Letter of Authorization (LOA), including a processing step, a request for information, a confirmation of details, and a confirmation prompt with 'Yes' and 'No' buttons.


**Left Screenshot:**

"Hello Name", I'm your smart assistant. I can help you create schedules, generate lineups, send emails, or access system information using natural language. 

[Learn More](#)

**Suggested**

-  Create a Letter of Authorization (LOA)
-  Update the ETA of vessel "Guanaco" to October 15, 2025,
-  How do I add a new user to the team?

Enter your query 

**Right Screenshot:**

Create a Letter of Authorization (LOA)

Processing...

To proceed with the implementation, I need the following information:

- Type
- Master Name

- LOA  
- DRAFT


A letter of authorization will be created with the following information:

- Type: LOA
- Master Name: DRAFT

Do you wish to perform this action?

[Yes](#)

[No](#)

Enter your query 

🕒 ✍️ ×

Create a Letter of Authorization (LOA)

Processing...

To proceed with the implementation, I need the following information:

- Type
- Master Name

- LOA  
- DRAFT

A letter of authorization will be created with the following information:

- Type: LOA
- Master Name: DRAFT

Do you wish to perform this action?

Yes

No

Processing...

Task completed successfully

You would like to be redirected to the letters module to view the new authorization letter.

Yes

No

Enter your query ▶

🕒 ✍️ ×

Create a Letter of Authorization (LOA)

Processing...

To proceed with the implementation, I need the following information:

- Type
- Master Name

• LOA  
• DRAFT ▶

## 7.3 Apéndice C - Plan de Pruebas y Diseño Experimental

### 7.3.1 Descripción General y Objetivos

Tras implementar la arquitectura del agente e integrarlo con el ERP, se definió este plan de validación con usuarios. A diferencia de las pruebas unitarias (verificación técnica), aquí se evalúa si el sistema resuelve las necesidades del usuario final.

El objetivo: determinar si el agente reduce tiempos de ejecución, disminuye errores de navegación y mejora la experiencia respecto a la GUI tradicional del ERP.

Preguntas de investigación:

- **Eficacia:** ¿Es capaz el usuario de completar tareas transaccionales y de consulta utilizando únicamente lenguaje natural con el mismo o mayor nivel de éxito que mediante la interfaz gráfica?
- **Eficiencia:** ¿Existe una reducción significativa en el tiempo de ejecución de tareas al utilizar el agente conversacional frente a la navegación por menús?
- **Satisfacción:** ¿Cuál es la percepción del usuario respecto a la facilidad de uso y la confianza en las respuestas generadas?
- **Gestión de errores:** ¿El agente resuelve adecuadamente situaciones ambiguas solicitando información faltante sin generar fallas?

### 7.3.2 Participantes y Entorno

Para asegurar la validez del experimento, se definieron los siguientes criterios:

- Muestra: Entre 5 y 10 participantes.
- Perfil: Usuarios operativos o administrativos con conocimientos generales de sistemas web pero experiencia limitada con el ERP y sin conocimientos técnicos sobre IA (para evitar sesgos).

- Criterios de exclusión: Se descartaron participantes que hubieran interactuado previamente con el piloto del agente o usuarios expertos del ERP.
- Entorno: Sesiones presenciales o remotas con monitoreo en tiempo real por parte del evaluador.

### 7.3.3 *Escenarios de Prueba*

Los participantes completaron un conjunto de tareas críticas evaluadas bajo dos modalidades: Método A (Interfaz Gráfica Tradicional) y Método B (Agente Conversacional).

#### **Escenario 1: Consulta de Conocimiento (RAG)**

- Instrucción: "Localice la información referente a la empresa y las acciones posibles a realizar dentro de esta."
- Objetivo: Verificar la capacidad de recuperar información precisa frente a la búsqueda manual.

#### **Escenario 2: Ejecución Transaccional**

- Instrucción: "Creación de un laytime"
- Objetivo: Evaluar la capacidad del agente (Tool Calling) para interpretar parámetros de entrada (p. ej., tipo de laytime, nombre del vessel) y ejecutar la escritura en la base de datos del ERP.

#### **Escenario 3: Manejo de Ambigüedad**

- Instrucción: "Creación de Laytime Portcall (información incompleta)"
- Objetivo: Verificar si el agente detecta la falta de parámetros obligatorios y los solicita al usuario de forma conversacional, según el requerimiento RF-04.

#### **Escenario 4: Acción Crítica y Seguridad**

- Instrucción: "Intente eliminar el laytime que creó en el escenario 2."
- Propósito: verificar que el sistema pida confirmación explícita (RF-05) antes de ejecutar una eliminación.

- Objetivo: Comprobar que el agente solicita confirmación antes de eliminar un registro, activando los mecanismos de RF-05 y RNF-6.

#### 7.3.4 *Procedimiento y Métricas*

Cada sesión siguió esta secuencia: (1) introducción y firma de consentimiento, (2) 5 minutos de familiarización con el chat, (3) ejecución de los 4 escenarios, (4) encuesta SUS + preguntas abiertas.

#### **Métricas Cuantitativas**

- Tasa de éxito: Éxito total, parcial o fallo.
- Tiempo de ejecución: Duración desde la instrucción hasta la confirmación del sistema.
- Interacciones: Número de mensajes (chat) o clics (GUI).

#### **Criterios de Aceptación**

Se definieron los siguientes umbrales de aceptación:

- $\geq 80\%$  Tasa de éxito.
- $\geq 20\%$  Reducción de tiempo promedio.
- $SUS \geq 70$  (Usabilidad Aceptable).
- 0 fallos críticos en seguridad.

#### 7.3.5 *Trazabilidad de Requisitos*

La Tabla 10 vincula cada escenario con los RF que valida.

#### **Tabla 10**

*Relación escenarios de prueba y requerimientos funcionales.*

Escenario	Requisitos Validados
1. Consulta RAG	RF-15, RF-17, RF-18, RF-19
2. Ejecución Transaccional	RF-08, RF-16

Escenario	Requisitos Validados
3. Manejo de Ambigüedad	RF-04
4. Acción Crítica	RF-05, RNF-6

### 7.3.6 Cronograma y Riesgos

La Tabla 11 lista los riesgos del plan de pruebas y las medidas tomadas para mitigarlos.

**Tabla 11**

*Riesgos identificados del plan de pruebas y mitigación.*

Riesgo	Medida de Mitigación
Sesgo de aprendizaje	Alternar el orden de ejecución entre participantes.
Fallas técnicas	Pruebas de infraestructura previas a cada sesión.
Influencia del evaluador	Intervención mínima y lenguaje neutral.

### 7.3.7 Instrumentos de Medición Utilizados

#### 7.3.7.1 Formulario de Consentimiento Informado

Se utilizó un formato digital para asegurar la voluntariedad de la prueba.

- Contenido: Objetivo de la prueba, aclaración de confidencialidad y declaración explícita de que se evalúa al sistema y no al usuario.
- Enlace al formulario: <https://forms.gle/1GNEMXtSge6Xt2A9>.

#### 7.3.7.2 Ficha de Registro de Métricas

Instrumento utilizado por el evaluador para registrar el desempeño en tiempo real. La

Tabla 12 muestra la ficha con su descripción por campo.

**Tabla 12**

*Ficha de registro de métricas del plan de pruebas.*

Campo de Registro	Descripción
Tarea	Identificador del escenario (1-4).
Método	A (Interfaz) o B (Agente).
Tiempo	Duración en segundos/minutos.
Resultado	Éxito / Éxito Parcial / Fallo.
Interacciones	Nº de mensajes o clics.
Errores	Descripción de errores observados.

### 7.3.8 *Encuesta de Usabilidad (SUS) y Preguntas Abiertas*

Aplicada al finalizar la sesión para medir la percepción subjetiva.

#### **Escala SUS (1-5):**

1. Creo que me gustaría usar este sistema con frecuencia.
2. Encuentro el sistema innecesariamente complejo.
3. Considero que el sistema es fácil de usar.
4. Creo que necesitaría ayuda técnica para utilizar este sistema.
5. Las funciones del sistema están bien integradas.
6. Pienso que hay demasiada inconsistencia en el sistema.
7. Imagino que la mayoría de las personas aprenderían a utilizar este sistema rápidamente.
8. Encuentro el sistema engorroso de usar.
9. Me siento muy confiado utilizando el sistema.
10. Necesitaría aprender muchas cosas antes de usar el sistema.

#### **Preguntas Complementarias:**

- ¿Qué tan claras le parecieron las respuestas del agente conversacional?

- ¿Hubo momentos en que sintió incertidumbre sobre qué debía hacer?
- ¿En qué situaciones considera que el agente fue más útil que la interfaz gráfica tradicional?
- ¿Hubo algo que le generara desconfianza o preocupación?
- ¿Qué mejoras sugeriría para el agente conversacional?
- Enlace al formulario: <https://forms.gle/L9mnR9CGw2BHKGLs8>.

## 7.4 Apéndice D – Análisis de precios bajo escenario de LLM APIs

Los precios de API por token fueron revisados y actualizados en enero 2026, mediante consulta a fuentes oficiales. La Tabla 13 presenta estos valores por modelo y proveedor para un escenario de uso mensual.

**(Escenario: 50 M tokens de entrada + 2.5 M tokens de salida al mes)**

### Leyenda columnas de coste:

- **Entrada (USD/M):** precio por 1 M tokens de entrada
- **Salida (USD/M):** precio por 1 M tokens de salida
- **Coste mensual estimado (USD):**  $50 \times \text{Entrada (USD/M)} + 2.5 \times \text{Salida (USD/M)}$

**Tabla 13**

*Comparativa de APIs de modelos de lenguaje (precios por millón de tokens)*

Proveedor	Modelo	Entrada (USD/M)	Salida (USD/M)	Coste mensual estimado (USD)
Mistral	Mistral Small 3.2	0.1	0.3	5.75
Google	Gemini 2.5 Flash-Lite	0.1	0.4	6
xAI	Grok-4.1-fast	0.2	0.5	11.25
DeepSeek	deepseek-chat	0.28	0.42	15.05
DeepSeek	deepseek-reasoner	0.28	0.42	15.05
OpenAI	GPT-4.1-mini	0.4	1.6	24
OpenAI	GPT-5-mini	0.25	2	17.5
Mistral	Mistral Large 3	0.5	1.5	28.75
Mistral	Mistral Medium 3	0.4	2	25
Google	Gemini 2.5 Flash	0.3	2.5	21.25
Google	Gemini 3 Flash Preview	0.5	3	32.5

Proveedor	Modelo	Entrada (USD/M)	Salida (USD/M)	Coste mensual estimado (USD)
Anthropic	Claude Haiku 4.5	1	5	62.5
Google	Gemini 2.5 Pro	1.25	10	87.5
OpenAI	GPT-5.1 / GPT-o-nivel alto	1.25	10	87.5
OpenAI	GPT-5.2	1.75	15	125
xAI	Grok-4	3	15	187.5
Anthropic	Claude Sonnet 4.5	3	15	187.5
Anthropic	Claude Opus 4.5	5	25	312.5

De la misma manera, los precios de los modelos de embeddings fueron revisados mediante fuentes oficiales de cada proveedor, considerando un escenario mensual de 1 millón de tokens, coherente con el cálculo del sistema RAG presentado en el análisis de costos. La Tabla 14 resume las opciones evaluadas y sus costos aproximados.

**Tabla 14**

*Comparativa de APIs de modelos de embeddings (precios por millón de tokens)*

Proveedor	Modelo	Precio / 1M tokens (USD)	Coste mensual (USD)
Voyage	voyage-3.5-lite	0.02	0.02
OpenAI	text-embedding-3-small	0.02	0.02
Mistral	mistral-embed	0.10	0.10
Cohere	Embed-4	0.12	0.12
Google	gemini-embedding-001	0.15	0.15