

INSTITUTO TECNOLÓGICO Y DE ESTUDIOS SUPERIORES DE MONTERREY

# Maestría en Inteligencia Artificial:

*Tarea1: Programación de una solución paralela*

**Alumno: Oscar Enrique García García - A01016093**

**Profesor Titular:** Gilberto Echeverría Furió

**Profesor asistente:** Yetnalezi Quintas Ruiz

Cómputo en la nube

Enero, 2026

# Índice

1. Introducción .....	1
2. Antecedentes .....	2
3. Entendimiento del negocio .....	3
3.1. Formulación del problema .....	3
3.2. Contexto .....	3
3.3. Objetivos .....	4
3.4. Alcance del Proyecto Integrador .....	4
3.5. Preguntas clave .....	5
3.6. Involucrados .....	5
4. Entendimiento de los datos .....	7
4.1. Descripción de los datos .....	7
4.2. Técnica de Machine Learning .....	7
4.3. Identificación de variables .....	7
5. Referencias .....	9

# 1. Introducción

Tradicionalmente, el software se ha escrito para el cómputo secuencial: un programa recibe una instrucción, la ejecuta y, solo cuando termina, pasa a la siguiente. Sin embargo, los procesadores modernos cuentan con múltiples núcleos (cores) que pueden trabajar simultáneamente.

La programación en paralelo es el proceso de dividir un problema grande en partes más pequeñas que pueden resolverse al mismo tiempo, aprovechando cada núcleo del procesador de forma concurrente y, de esta forma, optimizar el tiempo de ejecución.

El uso de paralelismo es fundamental en el desarrollo actual por varias razones:

- **Reducción de Tiempo:** Tareas que tardarían horas en un solo hilo pueden completarse en minutos al distribuir la carga.
- **Manejo de Big Data:** Permite procesar volúmenes masivos de información que, de otro modo, saturarían un solo núcleo.
- **Eficiencia Energética:** A menudo es más eficiente realizar un trabajo rápido usando varios núcleos a una frecuencia moderada que forzar un solo núcleo a su máxima capacidad por mucho tiempo.

## 2. Antecedentes

Spin es una organización que opera en el sector financiero y que mantiene una estructura de comunicación interna altamente dependiente de herramientas digitales, particularmente Slack. Dentro de este entorno, el área de Finanzas es responsable de la gestión, análisis y validación de los gastos de viaje realizados por los colaboradores, conocidos internamente como *Spinners*.

Actualmente, dicho proceso se realiza de forma mayormente manual, lo que implica la revisión individual de cada registro con el fin de verificar el cumplimiento de las políticas internas. Paralelamente, las consultas relacionadas con políticas internas de viajes, viáticos, equipos de cómputo y telefonía, etc. se atienden a través de múltiples canales, como mensajes directos, correos electrónicos y llamadas, sin contar con un punto centralizado de referencia.

La información oficial de la organización se encuentra distribuida en distintas plataformas y formatos, tales como Google Drive, Slack, Canvas y La Órbita (plataforma web interna), lo que provoca confusión, versiones contradictorias y consumo de información desactualizada. Este contexto genera fricción operativa, reprocesos y una limitada capacidad de escalamiento conforme la organización crece.

Ante estas condiciones, surge la iniciativa de desarrollar una solución integral que centralice la información institucional, automatice la gestión de gastos de viaje y provea un asistente conversacional capaz de resolver dudas frecuentes de manera consistente, confiable y alineada con las políticas oficiales de Spin.

## 3. Entendimiento del negocio

### 3.1. Formulación del problema

El problema central que se busca resolver es la ineficiencia operativa derivada de la gestión manual de los gastos de viaje y de la atención descentralizada de consultas sobre políticas internas. Esta situación genera altos costos de tiempo, errores humanos, respuestas inconsistentes y una dependencia excesiva del criterio individual del personal del área de Finanzas.

### 3.2. Contexto

El crecimiento de la organización ha incrementado tanto el volumen de gastos de viaje como la cantidad de consultas internas. Sin embargo, los procesos y herramientas actuales no escalan al mismo ritmo, lo que provoca saturación operativa y afecta negativamente la experiencia de los colaboradores. La ausencia de un sistema centralizado limita la trazabilidad y dificulta la estandarización de la información.

Resolver este problema es fundamental para mejorar la eficiencia del área de Finanzas, garantizar coherencia institucional y ofrecer una experiencia más ágil y clara a los *Spinners* dentro de su flujo de trabajo habitual.

La Figura 1 presenta la arquitectura general de la solución propuesta, en la cual se integran los canales de interacción de los usuarios, el asistente conversacional, los procesos de validación de gastos de viaje y la infraestructura que soporta el uso de modelos de aprendizaje automático.

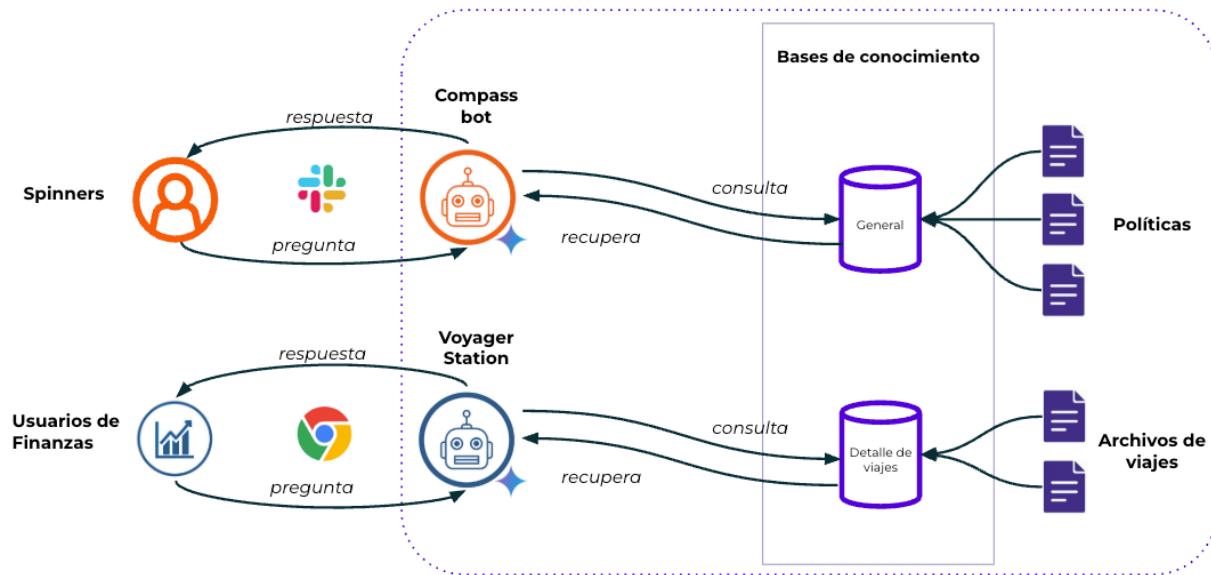


Figura 1: Arquitectura general de la solución Spin Voyager.

### 3.3. Objetivos

El objetivo del proyecto es diseñar e implementar una plataforma inteligente que permita automatizar la gestión y validación de los gastos de viaje, así como un asistente conversacional integrado en Slack que centralice las consultas sobre políticas internas. De manera específica, se busca reducir la carga operativa, mejorar la precisión de las respuestas y optimizar la experiencia del colaborador.

Como se muestra en la Figura 2, el proceso de gestión de gastos de viaje se automatiza mediante validaciones basadas en políticas internas, reduciendo el trabajo manual del área de Finanzas y mejorando la trazabilidad del proceso.

### 3.4. Alcance del Proyecto Integrador

Este proyecto se centrará en el desarrollo del agente conversacional embebido en la herramienta *core* de comunicación dentro de Spin: Slack, así como de una plataforma web, ejecutada por un equipo de operaciones, que será capaz de gestionar la seguridad y acceso, monitorear el desempeño del bot, analizar las conversaciones de los *Spinners* y

obtener métricas de analítica. Todo lo anterior involucra el diseño, desarrollo e implementación de los siguientes puntos:

1. Diseño de un modelo de datos con metodología Data Vault.
2. Desarrollo de código del agente conversacional con Gemini.
3. Implementación del código core en soluciones escalables y en la nube: Cloud Run.
4. Integración del agente conversacional con Slack, a través de su API para desarrolladores.
5. Creación de una base de datos PostgreSQL (idealmente en AlloyDB)

que fungirá como repositorio central del modelo del punto 1.

6. Desarrollo e implementación de la plataforma web para gestión, monitoreo y analítica en React.

### 3.5. Preguntas clave

Entre las preguntas que guían el desarrollo del proyecto se encuentran: ¿cómo automatizar la validación de gastos respetando las políticas internas?, ¿cómo garantizar respuestas consistentes y confiables?, ¿qué impacto tendrá la solución en la eficiencia operativa del área de Finanzas?, ¿cómo ajustar el agente conversacional con el fin de mejorar su precisión y velocidad?, y ¿cómo medir la adopción y precisión del asistente conversacional?

### 3.6. Involucrados

El proyecto involucra a distintos actores dentro de la organización. El sponsor del proyecto es el área de Finanzas dentro de Spin. El desarrollo e implementación están a cargo del Chapter de AI Products, liderado por Homero Merino que fungirá como nuestro patrocinador del proyecto , mientras que todos los *Spinners* fungirán como usuarios finales de la solución.

---

Adicionalmente, como se mencionó en la sección 3.4, también se considera un área de Operaciones que estará a cargo de la plataforma web de gestión de seguridad y accesos, monitoreo y mejora del agente, analítica avanzada, etc.



---

## 4. Entendimiento de los datos

### 4.1. Descripción de los datos

El proyecto se centra principalmente en datos no estructurados que incluyen documentos de políticas internas, lineamientos, FAQs y otros archivos oficiales almacenados en Google Drive.

Adicionalmente, se consideran las consultas realizadas por los usuarios en Slack, las cuales permiten analizar patrones de uso, temas recurrentes y áreas de oportunidad en la cobertura del conocimiento institucional.

### 4.2. Técnica de Machine Learning

La solución emplea una combinación de técnicas de aprendizaje automático. Se utiliza aprendizaje supervisado para la clasificación de intención de las consultas, aprendizaje no supervisado mediante embeddings para la recuperación de información relevante y modelos de lenguaje profundo para la generación de respuestas naturales. Todo ello se integra bajo un enfoque de Retrieval-Augmented Generation (RAG), asegurando que las respuestas se basen exclusivamente en fuentes oficiales.

### 4.3. Identificación de variables

Las variables de entrada incluyen el texto de las consultas de los usuarios, los documentos de políticas internas así como los metadatos del usuario, tales como rol y área. Las variables de salida consisten en la clasificación de intención, respuestas conversacionales alineadas con las políticas, alertas de posibles incumplimientos y métricas de uso y adopción del sistema.

Asimismo, se pretende que el agente sea capaz de detectar información personal de los usuarios (PII) y les dé un tratamiento diferente para *enmascararlos* a nivel de base de datos, mejorando la seguridad del mismo.

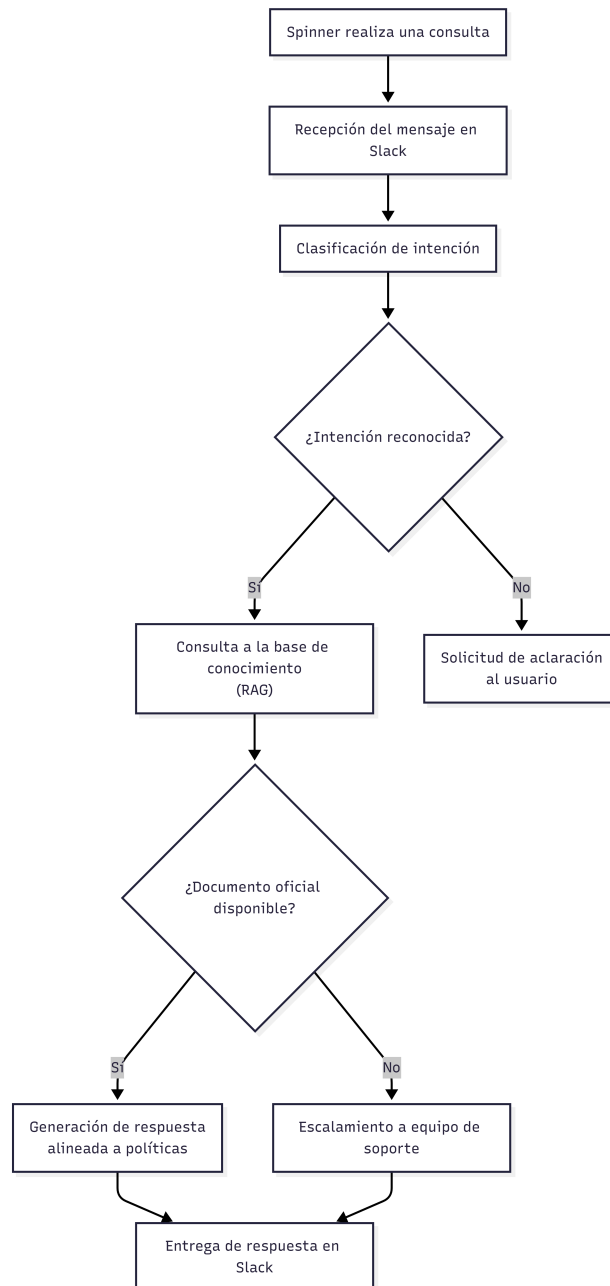


Figura 2: Flujo de operación del asistente conversacional Spin Compass.

La Figura anterior ilustra el flujo de operación del asistente conversacional, desde la recepción de la consulta en Slack hasta la generación de una respuesta alineada con las políticas internas o el escalamiento al equipo de soporte cuando no existe información documentada.

---

## 5. Referencias

- [1] L. Visengeriyeva, A. Kammer, I. Bär, A. Kniesz, y M. Plöd, «CRISP-ML(Q): The ML lifecycle process». Accedido: 22 de enero de 2026. [En línea]. Disponible en: <https://ml-ops.org/content/crisp-ml>
- [2] R. Hernández-Sampieri y C. Mendoza, *Metodología de la investigación: Las rutas de la investigación cuantitativa, cualitativa y mixta*, 3.<sup>a</sup> ed. McGraw-Hill, 2023.