



TaskFlow AI

Un equipo de agentes de IA coordinados para que cualquier PYME opere con **inteligencia artificial** sin escribir una sola línea de código

Año: 2026

Curso: 5K4

Grupo Nro: 8

Integrantes

Francisco Ochoa Rodríguez

Guido Mendez

Fabrizio Torres Daniele

Gonzalo Majlis

Tomas Morinigo

Legajo

96065

96910

94231

95047

95639



**UTN
FRC**

Cuerpo Docente: Sabrina A. Berreta, Victoria Ahumada,
Francisco Aquino, Silvina Arenas, Natalia Jaime, Felipe
Steffolani



TaskFlow AI

De una instrucción a una **acción real**

Plataforma SaaS de orquestación multi-agente para PYMES



"desorganización" "tareas manuales"
"demoras" "soluciones aisladas"
"conocimiento técnico" "imprevisibilidad"



"operación organizada" "tareas automatizadas"
"agentes especializados" "procesos más ágiles"
"todo en un mismo workspace" "lenguaje natural"
"acciones coordinadas" "resultados trazables"



INSTRUCCIÓN
Lenguaje natural

"envía un correo de seguimiento al cliente"



MANAGER

Interpreta las instrucciones y las **descompone** en tareas para luego **delegarlas** a los agentes.



WORKSPACE

Espacio compartido donde los agentes y el cliente **interactúan** con los **recursos** del proyecto.

AGENTES ESPECIALIZADOS

Creados como **expertos** en áreas de la empresa, **resuelven tareas específicas** con **herramientas propias**.

UNA PLATAFORMA, TODO TU EQUIPO COORDINADO

TaskFlow AI transforma instrucciones en acciones concretas mediante agentes especializados que organizan, ejecutan y documentan cada tarea en un espacio de trabajo compartido.

Menos trabajo manual, más control y resultados trazables.

STACK DEL PRODUCTO



UNIVERSIDAD
TECNOLÓGICA NACIONAL
FACULTAD REGIONAL
CÓRDOBA

Ingeniería en Sistemas de Información
Proyecto Final
5K4 - 2026 - Grupo 8

Majlis, Gonzalo
Méndez, Guido
Morinigo, Tomás
Ochoa Rodríguez, Francisco
Torres Daniele, Fabrizio

gonzalomajlis2@gmail.com
guidomendez08@gmail.com
morinigotomas1@gmail.com
francochaoarodriguez@gmail.com
fabrotorres3@gmail.com

JTP
Berreta, Sabrina
Responsable del curso
Jaime, Natalia



TaskFlow AI: Plataforma de orquestación multi-agente para la automatización de procesos operativos en PYMEs

Ochoa Rodríguez, Francisco - Mendez, Guido - Torres Daniele, Fabrizio

Majlis, Gonzalo Ezequiel - Morinigo, Tomas

Universidad Tecnológica Nacional, Facultad Regional Córdoba

Abstract

Se desarrolló TaskFlow AI, una plataforma SaaS multi-tenant orientada a PYMEs latinoamericanas, con el objetivo de automatizar procesos operativos de ventas y logística mediante agentes de inteligencia artificial coordinados. El sistema se construyó sobre una arquitectura en la que un Agente Manager interpreta instrucciones en lenguaje natural, las descompone en subtarefas y las delega a agentes especialistas, apoyados en un workspace persistente por tenant que actúa como memoria organizacional.

Se implementó la orquestación con LangGraph, la ejecución asíncrona con ARQ y Redis, la persistencia y el aislamiento multi-tenant con PostgreSQL (Supabase) y Row Level Security, y la integración real con Gmail mediante un flujo OAuth con credenciales cifradas. El desarrollo se gestionó con metodología ScrumBan en sprints de dos semanas.

Como resultado, se completó un ciclo end-to-end en el que una instrucción del usuario derivó en el envío real de un correo y en el registro de su resultado en el workspace. Se definieron además umbrales de calidad para los agentes, incluyendo una tasa máxima de alucinación del 15% medida con LangSmith.

Palabras clave

Orquestación multi-agente, agentes de IA, PYMEs, SaaS multi-tenant, memoria persistente, automatización de procesos, LangGraph. Categoría: Producto.

Introducción

La adopción de inteligencia artificial en las PYMEs es todavía limitada. Muchas empresas pequeñas y medianas gestionan ventas, pedidos, stock y reportes con herramientas dispersas como Gmail, Google Sheets y planillas internas, lo que genera procesos fragmentados, dependientes de la memoria humana y difíciles de escalar. Las soluciones existentes no resuelven el problema de forma integral: plataformas como Zapier o Make automatizan acciones puntuales pero no coordinan agentes con memoria y contexto compartido, mientras que frameworks como LangGraph o CrewAI requieren un conocimiento técnico que excede el perfil de

una PYME. A esto se suman la dificultad de prever el costo del consumo de tokens y la naturaleza sincrónica de muchas soluciones, que no acompañan procesos de negocio que requieren continuidad.

El objetivo de este proyecto es desarrollar TaskFlow AI, una plataforma SaaS que permite a una PYME operar agentes de IA especializados sin necesidad de programar ni configurar flujos técnicos. El usuario interactúa en lenguaje natural con un Agente Manager que interpreta el pedido, recupera contexto relevante, delega subtarefas a agentes especialistas y consolida una respuesta. De este modo, el sistema funciona como una capa operativa inteligente sobre las herramientas que la empresa ya utiliza. Como referencia de validación se adopta la empresa piloto Fábrica de Fideos Sbiroli, del rubro alimenticio.

El artículo se organiza de la siguiente manera: la sección Herramientas y Metodología describe el stack tecnológico, las decisiones de arquitectura y el proceso de desarrollo; la sección Resultados presenta los incrementos funcionales alcanzados; la Discusión interpreta esos resultados frente al objetivo planteado y sus limitaciones; y la Conclusión sintetiza los hallazgos y el trabajo futuro.

Elementos del Trabajo y Metodología

El sistema se estructuró en torno a una arquitectura multi-agente (Figura 1). Un Agente Manager central recibió las instrucciones, interpretó la intención, dividió el trabajo en subtarefas y las asignó a agentes especialistas de ventas y logística. Cada agente operó sobre un contexto específico y registró sus resultados en un Workspace Compartido (Tenant Drive), que funcionó como memoria persistente de la empresa y permitió que los agentes consultaran información previa en

lugar de partir de cero en cada interacción. Al estar separado por tenant, garantizó el aislamiento de los datos de cada cliente.

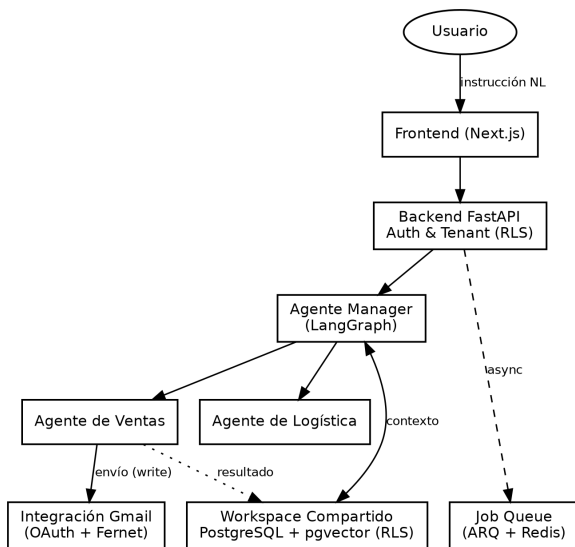


Figura 1. Arquitectura general de TaskFlow AI.

Para el frontend se utilizó Next.js con TypeScript, y para el backend FastAPI sobre Python, elegido por su integración nativa con el ecosistema de IA. La orquestación de agentes se implementó con LangGraph y la ejecución asíncrona de tareas de larga duración con ARQ y Redis, lo que permitió procesar trabajos en segundo plano sin bloquear al usuario. La persistencia y el aislamiento entre clientes se resolvieron con PostgreSQL sobre Supabase, aplicando Row Level Security por tenant; la búsqueda semántica del workspace se apoyó en la extensión pgvector. Los modelos de lenguaje se integraron de forma intercambiable (Claude y Gemini) mediante configuración por variable de entorno, y la observabilidad de los agentes se realizó con LangSmith.

La integración con Gmail se construyó sobre un flujo de autorización OAuth delegado por tenant, almacenando los tokens cifrados con Fernet. De este modo, cada cliente conectó sus propias cuentas y los agentes operaron en su nombre sin exponer credenciales.

El desarrollo se gestionó con metodología ScrumBan, en sprints de dos semanas, estimando el esfuerzo mediante T-Shirt Sizing y administrando el flujo de trabajo en un tablero Kanban sobre GitHub Projects.

La estrategia de pruebas se definió en tres niveles: pruebas unitarias, pruebas de integración, y pruebas end-to-end automatizadas con Playwright. Para la elaboración de la documentación del proyecto se utilizaron herramientas de IA generativa (Claude) como apoyo a la redacción y revisión de estilo; la verificación de los datos y el análisis final fueron responsabilidad del equipo.

Resultados

El desarrollo del MVP se completó a lo largo de cuatro sprints. En el primer sprint se entregó el flujo de autenticación y la configuración inicial del tenant con aislamiento mediante Row Level Security. En el segundo y tercer sprint se construyó el núcleo de orquestación: el Manager recibió instrucciones en lenguaje natural, las descompuso en subtarear, delegó automáticamente a los agentes especialistas y consolidó una respuesta. Además se incorporó el Workspace Compartido, que habilitó la persistencia de los resultados y la recuperación automática de contexto para nuevas tareas.

En el cuarto sprint se cerró el primer ciclo funcional end-to-end con ejecución de acciones reales. Se implementaron el catálogo de integraciones, el flujo de conexión OAuth con Gmail y el uso de la integración por parte del Agente de Ventas. Como resultado, una instrucción del usuario en lenguaje natural fue descompuesta por el Manager, delegada al Agente de Ventas, ejecutada mediante el envío real de un correo a través de Gmail y registrada en el workspace con referencia al destinatario, asunto y cuerpo. El alcance de la integración se acotó al envío (write) y lectura (read) de correos.

En términos de velocidad, el equipo completó los cuatro sprints con cumplimientos del backlog del 100%, 60%, 100% y 100% respectivamente, y entregó dieciséis historias de usuario. Cada historia entregada fue verificada mediante casos de prueba documentados. Se establecieron, además, umbrales de calidad para la etapa de validación integral: una cobertura de código

mínima de entre 70% y 80% en los componentes críticos, una tasa de éxito de los agentes de al menos 80% sobre el set de casos de prueba definido, y una tasa máxima de alucinación del 15% medida con LangSmith.

Discusión

Los resultados muestran que la propuesta es viable. El ciclo end-to-end demostrado valida que es posible traducir una instrucción en lenguaje natural en una acción real ejecutada sobre una herramienta que la PYME ya utiliza, con el resultado persistido como memoria organizacional. Esto cumple el objetivo central del proyecto: ofrecer una capa operativa inteligente y no un simple asistente conversacional.

Frente a herramientas de automatización como Zapier o Make, TaskFlow AI agrega contexto, delegación inteligente y persistencia; frente a frameworks técnicos como LangGraph o CrewAI, agrega una capa de producto utilizable por personas no técnicas. El Workspace Compartido constituye el principal diferencial, ya que aporta memoria corporativa real, mientras que el control del consumo de tokens responde a la necesidad de previsibilidad económica de las PYMEs.

El alcance funcional se acota a los flujos de ventas y logística. Asimismo, los umbrales de calidad de la IA se encuentran en etapa de validación y deben confirmarse con un volumen mayor de casos representativos.

Conclusión

TaskFlow AI aborda una necesidad concreta: permitir que las PYMEs adopten inteligencia artificial de forma operativa, coordinada y controlada. El MVP desarrollado demuestra el diferencial de la propuesta, al convertir una instrucción en lenguaje natural en una acción real, coordinada por múltiples agentes y respaldada por una memoria persistente. El valor del proyecto reside en reducir tareas repetitivas, mejorar la trazabilidad, mantener la continuidad operativa y dar previsibilidad sobre el consumo de IA.

Como trabajo futuro, el proyecto contempla ampliar el catálogo de integraciones y de

agentes, consolidar el dashboard de observabilidad y el rate limiting con datos reales de uso, y profundizar la medición de la calidad de la IA. El MVP definido constituye una base suficiente para validar el diferencial de la plataforma en un contexto PYME concreto.

Agradecimientos

Los autores agradecen a los docentes de la Cátedra de Proyecto Final de la Universidad Tecnológica Nacional Facultad Regional Córdoba, y en particular a la profesora Sabrina Berreta, por su guía y acompañamiento durante el desarrollo del proyecto

Referencias

- [1] LangChain Inc., “LangGraph Documentation.”. (Acceso: Julio 2026). Disponible en: <https://langchain-ai.github.io/langgraph/>
- [2] Supabase Inc., “Supabase Documentation — Row Level Security & pgvector.”. (Acceso: Julio 2026) Disponible en: <https://supabase.com/docs>
- [3] S. Ramírez, “FastAPI Documentation.”. (Acceso: Julio 2026). Disponible en: <https://fastapi.tiangolo.com/>
- [4] LangChain Inc., “LangSmith Documentation.”. (Acceso: Julio 2026). Disponible en: <https://docs.smith.langchain.com/>
- [5] Anthropic, “Claude.” Utilizado como asistencia en la redacción y revisión del presente artículo; el análisis y las conclusiones son de elaboración propia del equipo. (Acceso: Agosto 2026)

Datos de Contacto

Morinigo Tomas. Universidad Tecnológica Nacional, Facultad Regional Córdoba.
morinigotomas1@gmail.com

Ochoa Rodriguez Francisco. Universidad Tecnológica Nacional, Facultad Regional Córdoba.
franochoarodriguez@gmail.com

Torres Daniele Fabrizio. Universidad Tecnológica Nacional, Facultad Regional Córdoba.
fabrotorres3@gmail.com

Majlis Gonzalo Ezequiel. Universidad Tecnológica Nacional, Facultad Regional Córdoba.
gonzalo.ezequiel.majlis@gmail.com

Mendez Guido. Universidad Tecnológica Nacional, Facultad Regional Córdoba.
guidomendez08@gmail.com

PLANILLA DE CATALOGACIÓN DE PROYECTO

AÑO	2026	CURSO Y NRO. DE GRUPO	5K4 - G8
NOMBRE DEL SISTEMA / PROYECTO			
Taskflow AI			
CATEGORÍA (Solución de Negocio/ Producto/ Proyecto de Impacto Social)			
Producto			
HERRAMIENTAS Y TECNOLOGÍAS UTILIZADAS			
ÁMBITO DE APLICACIÓN		NOMBRE Y VERSIÓN	
ENTORNO DE DESARROLLO		Visual Studio Code / Node.js 22 + npm 10 / Next.js 15.5.18 + React 19.2.4 + TypeScript 5.9.3 + Tailwind CSS 4.2.4 / Python 3.11 + FastAPI 0.139 / Docker 28 + Docker Compose 2.37 / AWS EC2	
REPOSITORIOS Y VERSIONADO		Git 2.50 / GitHub / GitHub Projects / GitHub Actions / Pull Requests / Husky 9.1.7	
BASE DE DATOS		Supabase (PostgreSQL + pgvector; cliente supabase-py 2.31) / Redis 7 para coordinación de colas	
COMUNICACION INTERNA		Discord / WhatsApp / Google Meet / Google Calendar / GitHub	
CAPACITACION		Documentación oficial de Next.js, React, FastAPI, Supabase, AWS Bedrock, LangChain y LangGraph / GitHub Skills / asistencia de IA para desarrollo	
PRUEBAS DE SISTEMA		Vitest 4.1.5 / React Testing Library 16.3.2 / Playwright 1.59.1 / Pytest 9.1.1 / pruebas unitarias, de integración y End-to-End / ESLint 9.39 / Ruff 0.15 / MyPy 2.3	
GESTIÓN DEL PROYECTO		Scrum + Kanban / Sprints / Historias de Usuario / GitHub Projects / GitHub Issues / Pull Requests / GitHub Actions	
DOCUMENTACIÓN		Markdown / README / SDD (Spec Driven Development) / OpenAPI 3 + Swagger UI / DBML + DER / diagramas SVG	
MODELOS		Amazon Bedrock: Anthropic Claude Haiku 4.5 / Claude Sonnet 4.5 / Claude Opus 4.6 / Amazon Titan Text Embeddings V2 / LangChain 1.3.13 + LangGraph 1.2.9 / LangSmith para observabilidad	