



De mirar el pasado a
anticipar el futuro



Machine Learning con Python para la Toma de Decisiones Empresariales

Jorge Israel Frometa Moya

Agentes de IA: La Nueva Frontera de la Analítica de Negocios.



- De modelos que responden a sistemas que actúan y razonan.

Concepto clave: El paso del "Analista que usa herramientas" al "Orquestador de Agentes".

¿Qué es un Agente de IA y en qué se diferencia de un Chatbot?



- **Chatbot (Sesión 10):** Recibe un texto y genera una respuesta.
- **Agente (Sesión 11):** Tiene **herramientas (Tools)**. Puede acceder a bases de datos, ejecutar código de Python y tomar decisiones sobre qué paso dar a continuación para resolver un objetivo.
- **Ejemplo en el proyecto:** Un agente no solo dice "hay poco stock", sino que accede a stock_analysis.py, calcula la edad del inventario y propone una solución.

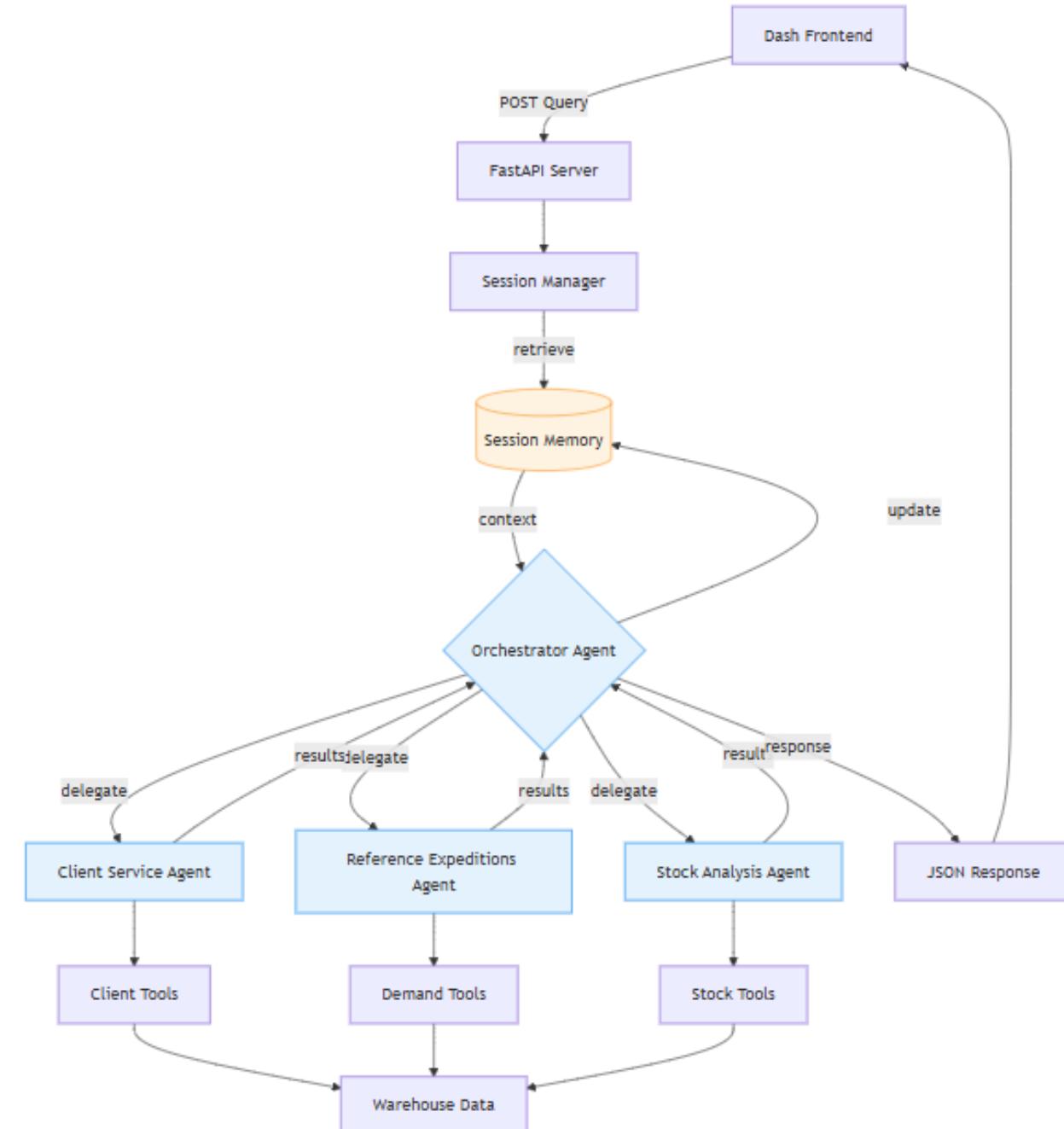
Arquitectura Multitarea (Multi-Agent System)

- **El Orquestador:** Es el cerebro central que recibe la duda del usuario y decide a qué experto llamar.

Especialistas del Proyecto:

- **Client Service Agent:** Experto en niveles de servicio.
- **Reference Expeditions Agent:** Experto en demanda y previsión.
- **Stock Analysis Agent:** Experto en optimización de almacén.

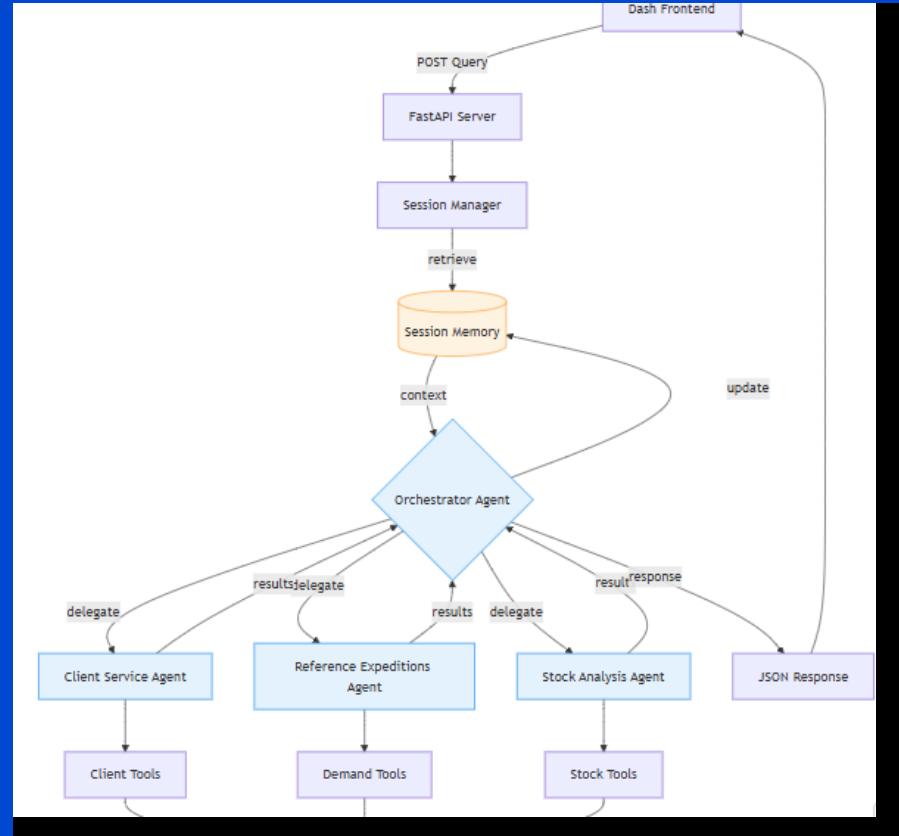
Beneficio: Evita que el modelo se confunda (alucinaciones) al darle un rol y herramientas específicas a cada agente.



El Stack Tecnológico del Proyecto

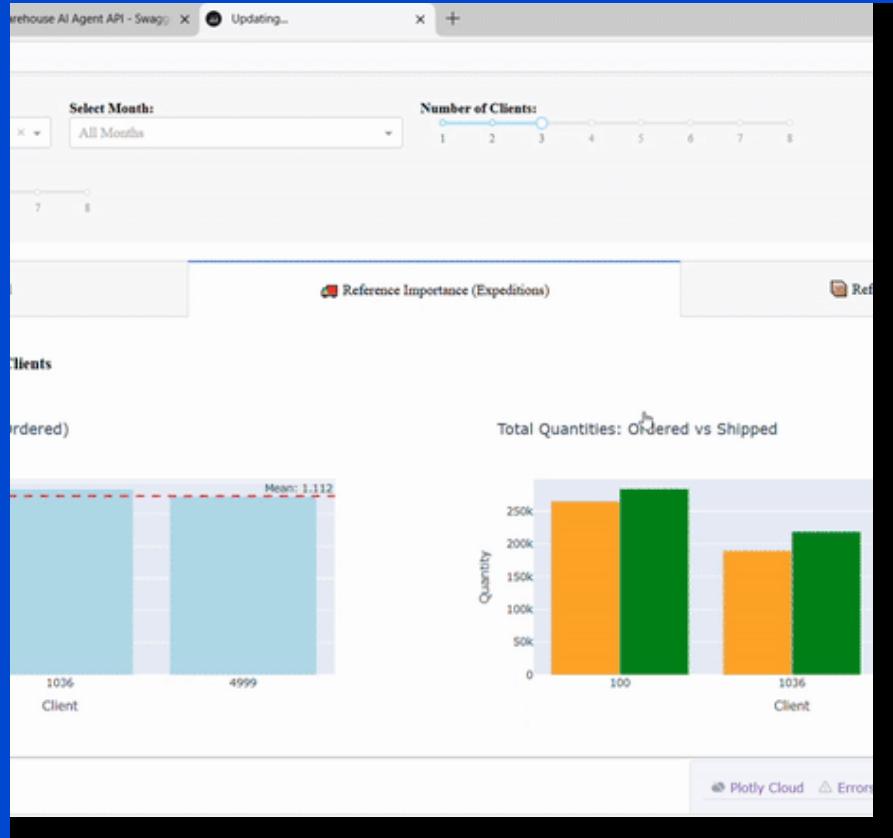


https://github.com/jfrometa88/capstone_project_new



- **Backend (FastAPI)**: El motor que gestiona la lógica de los agentes.
- **Frontend (Plotly Dash)**: La interfaz visual donde el negocio ve los datos y habla con la IA.
- **Eficacia Operativa**: Uso de Docker para asegurar que el sistema funcione igual en cualquier servidor, aislando la IA del Dashboard para mayor estabilidad.

La Interfaz Híbrida

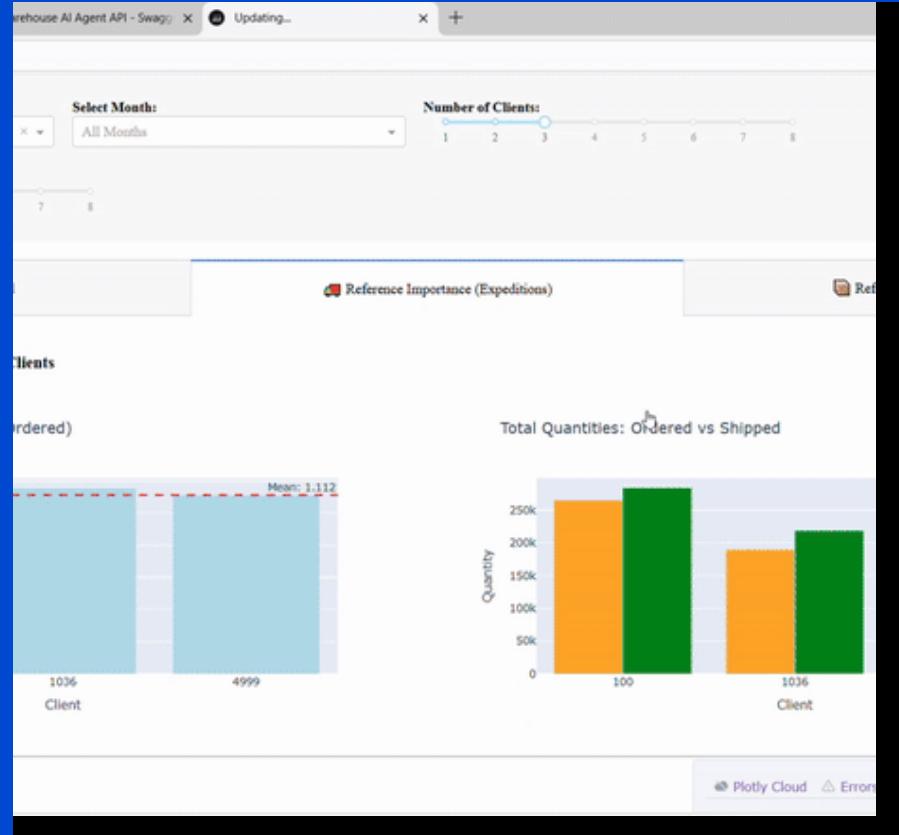


<http://deploy.tlsi.es:8601/>

El Dashboard ofrece lo mejor de los dos mundos:

- **Visualización Clásica:** Filtros interactivos de año/mes y gráficos de barras (lo que el jefe ya conoce).
- **Asistente Inteligente:** La capacidad de preguntar en lenguaje natural: "*¿Qué clientes tuvieron peor nivel de servicio este trimestre?*".

Aspectos relevantes de la arquitectura



- **Modularidad:** Si queremos añadir un "Agente de Costes de Transporte", solo tenemos que crear la herramienta y registrarla en el orquestador. No hay que programar todo el sistema de nuevo.
- **Contexto Masivo:** Al usar Gemini 2.5 Flash, el sistema puede "leer" grandes volúmenes de datos de expediciones sin perder el hilo de la conversación.
- **Conexión con Datos Reales:** Los agentes no "inventan" los niveles de servicio; los calculan usando las funciones de Python integradas en common/utils.