

# GLY NNE

MA

MDX

SAD

Moto20 / 20

+

-

0





MANUAL DE USUARIO - GLYNNE FRAMEWORK

El GLYNNE Framework es una arquitectura avanzada creada para integrar modelos de inteligencia artificial en los procesos empresariales de manera modular, escalable y totalmente adaptable. Su propósito es permitir a los desarrolladores y a las organizaciones construir ecosistemas de agentes cognitivos capaces de pensar, analizar, proponer y actuar por sí mismos dentro de los sistemas corporativos.

GLYNNE no es una herramienta más para ejecutar modelos de lenguaje. Es una infraestructura viva, pensada para que las empresas implementen inteligencias personalizadas que aprendan, se adapten y colaboren entre sí. Más que un framework de IA, GLYNNE es el inicio de una nueva forma de entender cómo la inteligencia artificial se integra con el ADN de las organizaciones.

**Este proceso no es solo técnico. Está impulsado por una visión transformadora de lo que puede ser una institución que aprende, mejora y se adapta a través de la tecnología. Es el resultado de una convicción profunda: que la automatización no debe deshumanizar el trabajo, sino potenciar el talento humano, liberar recursos operativos y permitirnos enfocarnos en tareas estratégicas, creativas y de impacto. Esta es la hoja de ruta hacia GLYNNE FW más eficiente, más coherente y, sobre todo, preparada para escalar y evolucionar con autonomía y propósito.**



**GLY  
NNE**

## **Quiénes somos**

GLYNNE es una empresa de ingeniería de software e inteligencia artificial con un enfoque en automatización empresarial a gran escala.

Nuestro trabajo no se centra únicamente en desarrollar herramientas, sino en diseñar sistemas que piensan, aprenden y evolucionan junto con la empresa. En GLYNNE creemos que la automatización no debe ser una meta, sino un medio para construir autonomía operativa, reducir cuellos de botella y liberar el verdadero potencial de los equipos humanos.

## **Nuestra visión**

Nuestra misión es convertir la inteligencia artificial en una infraestructura natural dentro de la empresa, de modo que los procesos rutinarios, repetitivos o lentos sean asumidos por sistemas inteligentes, mientras el talento humano se enfoca en la creatividad, la estrategia y la innovación. No queremos reemplazar personas: queremos crear entornos donde las personas y las máquinas colaboren de forma inteligente y armoniosa.

## **OBJETIVO GENERAL**

El objetivo principal del GLYNNE Framework es facilitar la implementación de modelos de lenguaje (LLMs) dentro de los flujos de trabajo empresariales, permitiendo:

- Crear agentes inteligentes personalizados, cada uno con su rol, personalidad y propósito.
- Conectar modelos de inteligencia artificial a sistemas reales de gestión y operación.
- Automatizar tareas, análisis y reportes de forma conversacional.
- Implementar un entorno de IA adaptable, escalable y conectado a frontends modernos.

En esencia, GLYNNE busca democratizar la construcción de inteligencias artificiales empresariales. Permite que cualquier desarrollador cree agentes capaces de analizar, decidir y evolucionar.

## **ESTRUCTURA DEL PROYECTO**

El framework está organizado en módulos que garantizan claridad, flexibilidad y orden. La estructura principal es la siguiente:

```
/  
|   main.py (API principal con FastAPI y Uvicorn)  
|   CLI.py (Interfaz de línea de comandos del framework)  
|   core/  
|       imports.py (Importaciones y componentes base del sistema)  
|       panel.py (Núcleo de configuración y orquestación)  
|       init.py  
|       user/  
|           panel.py (Configuración personalizada del usuario)  
|       .env (Variables de entorno)  
|       .gitignore (Archivos ignorados por Git)  
└       requirements.txt (Dependencias requeridas)
```

Descripción de los módulos:

#### **main.py**

Contiene la aplicación FastAPI, que actúa como puente entre el modelo de inteligencia artificial y el frontend. Permite enviar mensajes, recibir respuestas y gestionar la interacción con los agentes.

### **Descripción de los módulos:**

#### **main.py**

Contiene la aplicación FastAPI, que actúa como puente entre el modelo de inteligencia artificial y el frontend. Permite enviar mensajes, recibir respuestas y gestionar la interacción con los agentes.

#### **CLI.py**

Permite ejecutar el framework desde la línea de comandos. Desde aquí puedes probar modelos, inicializar configuraciones o ejecutar scripts de mantenimiento.

#### **core/imports.py**

Centraliza las librerías principales y las dependencias del sistema. Aquí se define la conexión con LangChain, Groq y LangGraph.

#### **core/panel.py**

Es el motor del framework. Orquesta la lógica que une el modelo, la memoria, el contexto y la respuesta final del agente.

#### **user/panel.py**

Es el espacio del usuario. Aquí defines cómo piensan tus agentes: qué modelo usan, qué tono tienen, cuál es su rol y cómo responden.

# INSTALACIÓN DEL FRAMEWORK

Antes de iniciar, asegúrate de tener un entorno virtual configurado.

1. Ingresa a la carpeta venv.
2. Dentro de la carpeta venv entra a la carpeta bin.
3. Ejecuta el siguiente comando para activar el entorno virtual:
4. **source activate**
5. Una vez activado, regresa a la carpeta raíz del proyecto.
6. Ejecuta el siguiente comando para instalar todas las dependencias:
7. **pip install -r requirements.txt**

Si por alguna razón este comando no funciona, puedes instalar los módulos manualmente con:

**pip install python-dotenv langchain langchain-groq langgraph fastapi pydantic uvicorn**

## EJECUCIÓN DEL FRAMEWORK

Una vez instalado, puedes ejecutar el framework de dos formas:

1. Ejecutar el modelo directamente desde el CLI:
2. `python CLI.py`
3. Esto probará el modelo configurado en `user/panel.py`.
4. Ejecutar el servidor para conectarlo al frontend:
5. **uvicorn main:app --host 0.0.0.0 --port 8000**

Esto iniciará el servidor localmente en:

<http://0.0.0.0:8000>

Puedes probar el funcionamiento abriendo en tu navegador la ruta:

<http://0.0.0.0:8000/chat>

Desde ahí podrás interactuar con el agente en tiempo real.

## MODELOS DISPONIBLES

GLYNNE es compatible con modelos de lenguaje de última generación, especialmente los disponibles en Groq. Entre ellos:

- Llama-3.1-8B-Instruct: eficiente y económico, ideal para chatbots.
- Llama-3.3-70B-Versatile: modelo potente y profundo, recomendado para entornos empresariales.
- Mixtral-8x7B: especializado en razonamiento y análisis complejo.
- Gemma-7B-It: ligero y rápido, ideal para entornos móviles.
- Llama-Guard-7B: enfocado en moderación y seguridad de contenido.

ajustable para cualquier modelo LLM

## **PARÁMETROS DE CONFIGURACIÓN**

Cada modelo puede configurarse con parámetros que definen su personalidad:

temperature: nivel de creatividad (de 0.1 a 1.0).

max\_tokens: cantidad máxima de tokens por respuesta.

top\_p: control de diversidad de salida.

Por ejemplo, un agente analítico tendrá una temperatura baja, mientras que uno creativo podrá tener una temperatura alta para generar ideas.

## **CREACIÓN DE NUEVOS AGENTES**

Cada agente que creas representa una nueva inteligencia dentro del ecosistema.

Para agregar un nuevo agente:

1. Entra a la carpeta user.

2. Crea un nuevo archivo, por ejemplo **panel2.py**.

3. Dentro de panel2.py, configura un nuevo modelo y un nuevo prompt.

4. Puedes importar este nuevo agente en main.py para exponerlo por otro endpoint o reemplazar el existente.

Cada panel actúa como un cerebro independiente, con su propia memoria y propósito. Así puedes tener múltiples agentes en un mismo entorno: uno para auditorías, otro para ventas, otro para soporte técnico, etc.

## **CREACIÓN DE NUEVOS AGENTES**

Cada agente que creas representa una nueva inteligencia dentro del ecosistema.

Para agregar un nuevo agente:

1. Entra a la carpeta user.

2. Crea un nuevo archivo, por ejemplo **panel2.py**.

3. Dentro de panel2.py, configura un nuevo modelo y un nuevo prompt.

4. Puedes importar este nuevo agente en main.py para exponerlo por otro endpoint o reemplazar el existente.

Cada panel actúa como un cerebro independiente, con su propia memoria y propósito. Así puedes tener múltiples agentes en un mismo entorno: uno para auditorías, otro para ventas, otro para soporte técnico, etc.

Este ejemplo muestra cómo enviar un mensaje desde el frontend a un backend FastAPI. Se define la URL del API (API\_URL) usando una variable de entorno o un valor por defecto. La función enviarMensaje hace un POST al endpoint /chat enviando un JSON con user\_id, rol y mensaje. Si la respuesta es exitosa, se parsea como JSON y se devuelve; si hay algún error de red o del servidor, se capturea y se muestra en la consola. Esto permite integrar fácilmente el chat del usuario con el agente definido en el backend.

js

Copiar código

```
const API_URL = process.env.NEXT_PUBLIC_API_URL || 'http://0.0.0.0:8000';

const enviarMensaje = async ({ user_id, rol, mensaje }) => {
  try {
    const res = await fetch(`${API_URL}/chat`, {
      method: 'POST',
      headers: { 'Content-Type': 'application/json' },
      body: JSON.stringify({ user_id, rol, mensaje }),
    });

    if (!res.ok) throw new Error(`HTTP ${res.status}: ${res.statusText}`);

    const data = await res.json();
    return data; // { rol, mensaje, respuesta }

  } catch (err) {
    console.error('✖ Error al conectar con el backend:', err);
    throw err;
  }
};
```



## Contacto y Recursos Oficiales

Si deseas implementar GLYNNE en tu empresa, colaborar en el desarrollo del ecosistema o solicitar una demostración corporativa, utiliza los siguientes canales oficiales:

RecursoEnlace

**Sitio Web Oficial**

<https://www.glynne.ai>

**Correo de contacto**

alexglynne7@gmail.com

**Soporte Técnico y Colaboraciones**

Wpp +57 3123455328

**Documentación Técnica**

Disponible próximamente

**Sede Principal**

Bogotá, Colombia

GLYNNE no es una herramienta.

Es el comienzo de una nueva especie digital,  
donde cada empresa puede crear sus propios cerebros,  
y donde cada sistema puede aprender, decidir y evolucionar.  
La inteligencia ya no se programa.

Se cultiva.

**GLYNNE Technologies S.A.S.**

Creando ecosistemas inteligentes para el futuro empresarial.



**GLYNNE S.A.S**

Crra 2 a No 1-24 sur  
Madrid Cundinamarca  
Nit:9019665123



---

**GLYNNE — Alexander Quiroga**  
Gerente principal  
whatsApp - 3123455328  
alexglynne7@gmail.com

**GLYNNE** — Arquitectura Inteligente para Empresas Inteligentes.  
Desarrollamos autonomía. Construimos futuro.

# GIGA NATION

