








# ARKAIOS - Integración con Gemini IA

## Descripción

Sistema completo para integrar con la IA de Google Gemini, incluyendo interfaz web moderna, servidor backend con memoria persistente y launcher automático.

## Características

-  **Interfaz Web Moderna:** UI responsive con tema oscuro
-  **Memoria Persistente:** Sistema de memoria a corto y largo plazo
-  **Subida de Archivos:** Soporte para imágenes y documentos
-  **Modo ROOT:** Para tareas administrativas
-  **CORS Configurado:** Funciona localmente y online
-  **Estado de Conexión:** Indicador visual del estado del servidor
-  **Drag & Drop:** Interfaz intuitiva para archivos

## Instalación

### Requisitos Previos

- Python 3.7+ instalado
- Clave API de Google Gemini (obtener desde [Google AI Studio](#))

### Pasos de Instalación

1. **Descargar archivos:** Coloca todos los archivos en una carpeta
2. **Ejecutar el launcher:** Haz doble clic en `iniciador+ia.bat`

3. **Configurar API:** Edita el archivo `.env` que se creará automáticamente
4. **¡Listo!:** La interfaz se abrirá automáticamente

## Configuración

### Archivo `.env`

```
GEMINI_API_KEY=tu_clave_api_real_aqui
GEMINI_MODEL=gemini-2.0-flash-exp
MEMORY_DIR=data/memory
MEMORY_MAX_TURNS=8
MEMORY_SUMMARY_EVERY=6
```

### Variables de Configuración

- `GEMINI_API_KEY` : Tu clave de API de Gemini (OBLIGATORIO)
- `GEMINI_MODEL` : Modelo a usar (por defecto: gemini-2.0-flash-exp)
- `MEMORY_DIR` : Directorio para almacenar la memoria
- `MEMORY_MAX_TURNS` : Número de turnos recientes en contexto
- `MEMORY_SUMMARY_EVERY` : Cada cuántos turnos actualizar el resumen

## Estructura de Archivos

```
proyecto/
├── iniciador+ia.bat          # 🚀 Launcher principal
├── server_gemini.py         # 🛠️ Servidor backend
├── magic_gemini.html       # 🎨 Interfaz web
├── .env                    # ⚙️ Configuración (se crea automáticamente)
├── uploads/               # 📁 Archivos subidos
└── data/memory/          # 🧠 Datos de memoria persistente
```

## Uso

### Iniciar el Sistema

1. Ejecuta `iniciador+ia.bat`
2. El servidor se iniciará en `http://127.0.0.1:8000`
3. La interfaz web se abrirá automáticamente

### Funcionalidades de la Interfaz

- **Chat:** Escribe mensajes y recibe respuestas de Gemini
- **Archivos:** Arrastra y suelta o selecciona archivos para adjuntar
- **Modo ROOT:** Activa para tareas administrativas
- **Ping:** Verifica la conexión con el servidor
- **Limpiar:** Borra el historial y memoria de la conversación

### Endpoints del API

- `GET /`: Interfaz web principal

- `POST /chat` : Enviar mensajes a Gemini
- `POST /upload` : Subir archivos
- `POST /clear` : Limpiar memoria de conversación
- `GET /health` : Estado del servidor
- `GET /memory` : Información de memoria

## Correcciones Realizadas

### Problemas Solucionados

1. **Error tipográfico:** `sommary_safe` → `summary_safe`
2. **Decorador Flask:** Corregido `@app.post` → `@app.route`
3. **Manejo de CORS:** Agregado soporte completo para CORS
4. **Verificación de archivos:** Mejorada la verificación de dependencias
5. **Interfaz mejorada:** Agregado estado de conexión y mejor UX
6. **Manejo de errores:** Mejor gestión de excepciones
7. **Launcher mejorado:** Verificación completa de dependencias y configuración

## Solución de Problemas

### El servidor no inicia

- Verifica que Python esté instalado
- Asegúrate de que las dependencias estén instaladas: `pip install flask`  
`python-dotenv`
- Revisa que el puerto 8000 no esté ocupado

## Error de API Key

- Verifica que tu clave de Gemini API sea correcta
- Asegúrate de que esté configurada en el archivo `.env`
- Verifica que tengas créditos disponibles en tu cuenta de Google AI

## Problemas de conexión

- Usa el botón "Ping" para verificar la conexión
- Revisa que no haya un firewall bloqueando el puerto 8000
- Verifica que todos los archivos estén en el mismo directorio



## Notas Técnicas

- **Puerto:** El servidor usa el puerto 8000 por defecto
- **Memoria:** Se guarda automáticamente en `data/memory/`
- **Archivos:** Se almacenan temporalmente en `uploads/`
- **Formatos soportados:** Imágenes (PNG, JPG, WebP), PDFs, texto
- **Límite de archivos:** 10MB por archivo



## Soporte

Si encuentras problemas:

1. Verifica que todos los archivos estén presentes
2. Revisa la configuración del archivo `.env`
3. Usa el botón "Ping" para diagnosticar la conexión
4. Revisa la ventana del servidor backend para mensajes de error

---

✨ ¡Arkaio está listo para usar con Gemini IA! ✨