CREACIÓN DE AGENTES DE IA:

n8n, make, zapier, Relevance AI

Santiago Hernández Ramos

Tema 1 · Fundamentos de los Agentes de IA

Resumen del tema

En este primer tema se presentan los **fundamentos de los agentes de IA**: qué los define, en qué se distinguen de la automatización clásica y cuáles son sus cinco componentes esenciales — cerebro (LLM), memoria, conocimiento externo, herramientas e instrucciones (*prompt*) — junto con el flujo de trabajo típico para diseñarlos, ejecutarlos y mantenerlos en plataformas no-code como n8n, Make, Zapier o Relevance IA.

1 · ¿Qué es un agente de IA?

Concepto	Definición corta	Diferencia frente a la automatización clásica
Agente de IA	Sistema que **decide, actúa y se adapta** para lograr un objetivo utilizando IA.	No sigue un flujo rígido: **razona** y elige la mejor acción en cada paso.
LLM	“Large Language Model”; el “cerebro” que interpreta el contexto y genera la siguiente acción.	Sustituye las reglas fijas por **razonamiento probabilístico**.
Tareas	Acciones atómicas que el agente puede ejecutar (enviar correo, llamar API, leer base de datos).	Pueden cambiar dinámicamente según el contexto.

2 · Componentes imprescindibles del agente

Componente	¿Para qué sirve?	Opciones habituales
Cerebro (LLM)	Procesa el *prompt*, entiende objetivos y decide la siguiente tarea.	GPT-4o, GPT-3.5-Turbo, Gemini 1.5 Pro, Azure OpenAI, Claude 3 Opus.
Memoria	Guarda la conversación o estados para que el agente "recuerde".	**Corto plazo** (context window del LLM) · **Largo plazo** (Bases de datos Vectoriales: Pinecone, Chroma, Weaviate).
Conocimiento externo	Datos que el agente necesita pero no están en la memoria.	SQL, Google Sheets, Salesforce, Notion, SharePoint, APIs propias.
Herramientas / Integraciones	Permiten que el agente actúe en el mundo real.	Email, Slack, WhatsApp, GitHub, Zapier/N8N HTTP Request, navegadores headless.
Prompt / Instrucciones	Lenguaje natural que describe el rol, las metas y las restricciones.	Sistema (definir rol) · Usuario (objetivo) · Ejemplos (one-shot / few-shot).

3 · Flujo de trabajo típico (alto nivel)

1. **Recepción de objetivo** → El usuario envía un *prompt* inicial.
2. **Razonamiento del LLM** → Evalúa contexto + memoria.
3. **Selección de herramienta** → Elige la mejor integración para el siguiente paso.
4. **Ejecución de la tarea** → Llama a la aplicación externa.
5. **Actualización de memoria** → Registra resultado y continúa hasta completar el objetivo.

Página de Notas del Tema

Esta página está pensada para que puedas anotar ideas clave, dudas y reflexiones importantes sobre el tema anterior.

Tema 2 · Zapier: Primer Agente de IA para correos y organización personal

Resumen del Tema

En este tema descubrirás cómo Zapier convierte tus ideas en flujos automáticos sin escribir código: desde crear tu primer Zap con un disparador y una acción, explorar plantillas que inspiran, y pasar a los Agentes de IA que razonan y actúan por ti; hasta construir formularios y tablas para captar datos limpios, lanzar chatbots que responden con conocimiento propio y, por último, usar Canvas para describir en texto un proceso y ver cómo se generan al instante los formularios, Zaps y agentes que lo ejecutan.

---

1 · Panel de control

Área	Qué contiene	Pistas rápidas
Barra "What do you want to automate?"	Prompt → busca plantillas o genera Zaps	Escribe lo que quieres crear
Zaps	Listado de flujos de automatización	Activa/Pausa con el conmutador
Tables	Mini-bases de datos internas	Guarda contexto para IA
Interfaces	Formularios y front-ends	Conecta con Tables o Zaps
Chatbots	IA conversacional	Configura tono y fuentes
Canvas	Lienzo visual + Lenguaje natural	Ideal para prototipos
AI Agents	Zaps + memoria + razonamiento	Evolución de los flujos de automatización
Templates	Casos pre-configurados	Filtra por app o resultado

2 · Crear un Zap

Home ▶️ Zaps ▶️ **Create** ▶️ **New Zap**

1. **Asistente IA**
    - Describe el flujo en lenguaje natural.
    - Propone Trigger + Actions.
    - Ajusta y haz clic **Try it**.
2. **Manual**
    - Añade **Trigger** primero.
    - Luego **Actions** (∞ pasos en planes de pago).
    - Usa + para añadir filtros o más acciones

> **Tip:** Nombra tu Zap en la cabecera (ej. “Leads → CRM”) para localizarlo rápido.

3 · Disparadores (Triggers)

Qué configuras	Para qué sirve	Ejemplos útiles
Aplicación	Lo que vigila el Zap	Google Sheets, Gmail, Discord, Webhooks
Evento	Situación que lo “despierta”	*New Spreadsheet Row*, *New Email*, *Message Posted*
Cuenta	Credenciales con acceso mínimo	Conecta solo el workspace o carpeta necesaria
Detalles	Recurso o filtro concreto	Hoja “Pedidos 2025”, canal “#soporte”, URL del webhook
Prueba	Carga datos reales de ejemplo	Verifica que aparecen los campos que necesitas

> En planes **Starter+** puedes añadir varios triggers en un mismo Zap.
> El trigger **Schedule** lanza el flujo según hora, día o frecuencia que definas.

4 · Acciones (Actions)

Paso	¿Qué define?	Tips rápidos
App + Evento	Qué tarea ejecuta	Drive → *Create File*, Trello → *Add Card*
Mapear datos	Llenar campos con valores del trigger	Haz clic en **Insert Data** o escribe "{{" para ver variables
Test	Ensayo con datos reales	Asegura formato correcto antes de publicar

5 · Probar y publicar un Zap

1. **Test Trigger / Test Action** – botones de prueba en cada paso · Ejecuta con datos reales y muestra *Data In / Data Out*.
2. **Publish** – botón verde (esquina sup. der.) · Activa el flujo y crea la versión 1.
3. **Interruptor On / Off** – lista de Zaps · Pausa o reactiva sin perder la configuración.
4. **Zap Runs** – abre el Zap ▶️ Zap Runs (barra lateral) · Cronología de cada ejecución con opción **Re-run** tras corregir errores.

> Con plan Free el panel puede tardar ~5 min en reflejar nuevas ejecuciones.

6 · Agentes de IA en Zapier – Visión general

Componente	Equivalente en Zaps	Particularidad
Trigger	Disparador	Misma lógica (ej. "New Row in Sheets").
Instructions	Acciones	Descritos 100 % en lenguaje natural.
Chat	—	Consola que muestra el pensamiento de la IA y permite feedback en vivo.

- Los agentes pueden **razonar y decidir** qué apps/acciones usar en cada ejecución.
- Se configuran con un *prompt* extenso; cuanto más contexto, mejor desempeño.

7 · Límites útiles en el plan Free (Agentes)

Recurso	Límite	Implicación
Agent activities	400/mes	Cada ejecución completa cuenta como 1.
Polling delay	~5–10 min	Los triggers no son instantáneos.
Instrucciones	Ilimitadas	Puedes encadenar muchas acciones en un solo agente.

> Aprovecha un solo agente con varias acciones para evitar la restricción de 2-pasos de los Zaps Free.

8 · Disparador del agente

Paso	Qué eliges	Preguntas clave
App	Google Sheets, Gmail, Notion…	¿Dónde ocurre el evento?
Evento	*New Row*, *New Email*, *Webhook*…	Debe reflejar el cambio que "despierta" al agente.
Cuenta	OAuth mínimas credenciales	Usa cuentas de servicio si es posible.
Recurso	Hoja, bandeja, base, colección…	Confirma ruta y permisos.
Prueba (Test Trigger)	Verifica acceso + datos muestra	Ajusta si falla: sheet vacío, credencial caducada…

> Marca el icono ⚠️ rojo como "resuelto" al completar todos los campos.

9 · Instrucciones y herramientas

1. **Describe la lógica paso a paso** en lenguaje natural.
2. Inserta herramientas con **/** → selecciona *App · Acción*.
3. Deja *"Let the agent decide"* en los parámetros que pueda deducir (hoja, columna, drive…).

4. **Sé explícito:** nombra hojas, columnas y formatos para evitar preguntas en el chat.

Ejemplo de sintaxis	Explica
`/google_sheets.find_many_rows`	Lee histórico.
`/google_sheets.format_row`	Cambia color o estilo.
`/gmail.create_draft`	Redacta respuesta sin enviarla.

> **Menos es más:** cada agente debería resolver **una** tarea concreta; usa *Pods* para agrupar tareas relacionadas.

10 · Probar y publicar el agente

Botón	Dónde	Resultado
Test Agent	Config ▶️ Test agent	Simula ejecución y abre un chat interactivo.
Approve / Reject	Dentro del chat	Continua o ajusta instrucciones.
Activate	Interruptor ▶️ On	Empieza a escuchar triggers reales.
Activity Log	Agents ▶️ All Activity	Lista ejecuciones, coste en *activities*, errores.

> Si el chat pide aclaraciones, añade la información al *prompt* y vuelve a probar antes de activar.

11 · Pods: organiza tus agentes

- **Crea un Pod** para un dominio funcional (p.ej. *Asistente personal*, *Operaciones de ventas*).
- Arrastra agentes al Pod; comparten panel de actividad y quota mensual.
- Ventajas:
  - **Visibilidad**: un solo registro de logs.
  - **Límites**: fácil ver cuántas *activities* consumen en conjunto.
  - **Mantenimiento**: desactivar o duplicar grupos enteros.

12 · Interfaces & Tables

Componente	Rol	Buenas prácticas
Interface	UI no-code (form, page, portal)	Usa campos *Required* para entrada limpia.
Table	Mini-base de datos interna	Define tipos (Date, Number, Enum) antes de enlazar.
Zap	"Pegamento" opcional	Sincroniza Table ⇄ Google Sheets, CRM, etc.

Pasos rápidos para un formulario:

1. **Crear Interface** ▶️ **Form**.
2. Ajustar campos y marcar obligatorios.
3. **Connect to Table** (una fila por envío).
4. Copiar URL pública → compartir.

13 ·Chatbot vs. Agente

- *Chatbot*: interfaz conversacional; responde en tiempo real; no ejecuta flujos complejos salvo que los encadene con Zaps.
- *Agente*: orquesta acciones, toma decisiones, puede usar chat como canal pero va más allá (razona, llama APIs, escribe en hojas…).

14 · Instrucciones del chatbot

Campo	Qué pones	Recomendaciones
Greeting	Saludo estático o dinámico	Ej. *«¿Qué dudas tienes sobre la privacidad de OpenAI?»*
Directives	Prompt con rol, tono y reglas	• Define objetivo del bot. • Estilo (formal/cercano). • Reglas si la pregunta está fuera de tema.
Cómo crearlas	Escribe a mano *o* usa ChatGPT > "Mejora estas instrucciones…"	Pide la salida "en caja de texto" para copiar-pegar.

> **Tip:** Sé explícito. Ej.: "Si la pregunta no es clara, pide detalles; si es ajena al tema, responde que solo cubres privacidad."

15 · Base de conocimiento

Fuente	Cómo añadirla	Uso típico
Web URL	Pega enlace (p.ej. Política de privacidad)	Rápido para docs públicos largos.
Files	PDF, DOCX, CSV… drag-and-drop	Manuales internos, FAQ PDF.
Table	Selecciona tabla Zapier	Datos dinámicos recogidos vía formularios.

- **Fallback**: Mensaje personalizado cuando no encuentra respuesta.
- **Logic** (opcional): muestra botón o lanza Zap si detecta palabras clave.
- **Settings** ▶️ **Creativity**: deslizador 0-100; baja si necesitas respuestas precisas.

16 · Probar y publicar el chatbot

Acción	Dónde	Resultado
Test Chat	Config ▶ Try it	Conversa y verifica restricciones.
Link público	Share ▶ Public link	Acceso inmediato vía URL.
Embed	Share ▶ *Add to website*	Genera snippet `<script>` para **chat fijo** o **pop-up**.
Integraciones	Integrations ▶ Gmail / Messenger …	Crea Zap ↔ genera y envía respuesta.

> Para sitios estáticos copia el *snippet* en HTML antes de `</body>`; tu bot aparecerá en la esquina como widget.

17 · Canvas – Automatiza "todo" con Lenguaje Natural

Paso	Qué hace Canvas	Ventajas
Describe el proceso	Prompt: "Recibo CV, extrae datos, puntúa, guarda…"	No-code, visión global.
Plan Preview	Muestra interfaces, Zaps, Tables, Agents propuestos	Edita antes de construir.
Build	Crea y conecta componentes	Ahorra configuración manual.
Edit Graph	Haz clic en nodos para afinar	Cambia prompts, cuentas, pasos.

Límites del plan Free

- Canvas no verifica tus *quotas*: si un Zap supera 2 pasos tendrás que ajustarlo o pasar a Starter.
- Los nodos "multi-step Zap" o "premium app" mostrarán aviso de upgrade.

> Usa Canvas para prototipos complejos: diseña, revisa con tu equipo y luego convierte cada bloque a la versión que encaje con tu plan.

Página de Notas del Tema

Esta página está pensada para que puedas anotar ideas clave, dudas y reflexiones importantes sobre el tema anterior.

Tema 3 · Make: Atiende a tus clientes por WhatsApp y Telegram con un Agente de IA

Resumen del tema

En este tema construirás de principio a fin un agente de IA en Make: configurarás un bot de Telegram como disparador, usarás webhooks para integrar aplicaciones externas, incorporarás modelos de Hugging Face y OpenAI mediante prompts estructurados, aplicarás routers y filtros para decidir rutas del flujo, extraerás datos con Text Parser, consultarás reportes en Google Drive y, finalmente, enviarás respuestas personalizadas al cliente, todo controlando costes y sin escribir código.

1 · Panel de control

Área (barra lateral)	Para qué sirve	Puntos esenciales
Organization	Facturación, roles y seguridad	Puedes añadir varios equipos y controlar permisos escenario a escenario.
Teams	Sub-grupos dentro de la organización	Mantiene separados proyectos de clientes o departamentos.
Scenarios	Lista, búsqueda y creación de flujos	Equivale a los *Zaps* de Zapier. Todo arranca aquí.
Templates	Galería de escenarios pre-configurados	Miles disponibles. Busca antes de reinventar la rueda.
More ⋯	Conexiones, Webhooks, API Keys, Devices, Data Stores	Suele configurarse dentro del editor; aquí los ves de un vistazo.

2 · Crear un escenario

1. **Scenarios ▶️ Create new**
2. Elige **Start from scratch** o una *template*.
3. En el lienzo:
    - Haz clic en + para añadir el **primer módulo** (Trigger).
    - Busca la app o servicio y selecciona el evento que inicia el flujo.
    - Repite + para añadir **acciones** o **routers** (ramas).
4. Sin límite de módulos, pero en plan gratuito cada ejecución no puede exceder **5 min**.

Atajo	Acción rápida
`Ctrl + S`	Guardar versión del escenario
`Ctrl + Arrastrar`	Duplicar módulo
Clic derecho ▶️ Delete	Eliminar un módulo
AI Assistant	Redacta en lenguaje natural el escenario (experimenta; varía su calidad)

3 · Tipos de módulos

Tipo	Icono en editor	Función	Ejemplos habituales
Trigger	● con borde grueso	Dispara el escenario	“Watch Emails”, “New Row”, Webhook custom.
Action	● simple	Opera sobre datos	“Create Item”, “Send Message”.
Iterator / Aggregator	∞	Repite o agrupa registros	Paginar API, sumar totales.
Router	⅄	Divide la ejecución en ramas	Filtrar por condiciones, A/B paths.
Tools	⚙️	Procesa datos	Formateo de fechas, JSON parse, funciones de texto.

4 · Configurar módulos

1. **Conexión**: autoriza la cuenta (OAuth 2, API Key, etc.).
2. **Campos obligatorios**: Make sugiere *Search* o pegar IDs (ej. ID del Form).
3. **Mapeo de datos**:
    - Ejecuta **Run once** en el Trigger para capturar una muestra.
    - Así verás la **burbuja** con los datos reales y podrás arrastrar los campos correctos.
4. **Notas y versiones**: usa *Notes* para colaborar y *Version history* para revertir.

5 · Probar y publicar

Paso	Qué hace	Buenas prácticas
Run once	Simula el flujo con datos de prueba	Verifica que cada módulo muestra ✅ en verde.
Save	Guarda borrador	Make crea un nuevo *Revision*.
Scheduling	Activa el escenario	Plan gratuito: mínimo cada **15 min**; planes de pago desde 1 min o *instant* (webhooks).
Status	ON / OFF	Mantén OFF en producción hasta completar pruebas.

6 · Webhooks (conectar apps "no soportadas")

Aspecto	Detalle clave
Creación	More ⋯ ▶️ Webhooks ▶️ + ▶️ Custom Webhook.
URL única	Make genera una URL HTTPS; compártela con el desarrollador externo.
Métodos	Acepta GET/POST (JSON, Form-Data, query params).
Escucha instantánea	Escenarios via Webhook se ejecutan en tiempo real (sin espera de 15 min).
IP Whitelist	(Opcional) Restringe quién puede llamar al Webhook.
Prueba	Usa `Run once` y envía una petición desde herramientas tipo *curl* o *hoppscotch*. Make "aprende" la forma del payload y muestra los campos listos para mapear.

7 · Límites del plan gratuito

- **Ejecuciones**: máx. 1000 operaciones/mes (puede variar).
- **Tiempo por run**: 5 min.
- **Frecuencia programada**: 15 min mínimo (excepto Webhooks).

- **Máximo de escenarios activos**: 2 (recomendación: archiva los que no uses).

8 · Estrategias para crear agentes de IA

Enfoque	¿En qué consiste?	Pros	Contras
Generador automático (p. ej. en Zapier)	Describes el objetivo en lenguaje natural y la plataforma construye todo el flujo.	Montaje ultrarrápido, ideal para prototipos.	Poco control sobre prompts, modelos y credenciales.
Flujos manuales en Make	Seleccionas tú cada módulo (trigger + acciones + modelos IA).	Máximo control sobre prompts, modelos, datos y costes.	Requiere entender la lógica de cada servicio.

9 · Telegram Bot como disparador

1. **Crea el bot**

```
Telegram → @BotFather → /newbot
```

   - Define **Nombre visible** y **Username** terminado en `_bot`.
   - Copia el **Token** que devuelve BotFather.

2. **Conecta en Make**
   - **+ módulo** ▶️ *Telegram* ▶️ **Watch Updates** (Trigger).
   - En *Create connection* pega el *Token* → *Save*.
   - Make genera un **Webhook** que Telegram usará internamente.

3. **Añade el bot a un canal/grupo**
   - Crea **Canal** o **Grupo** privado.
   - Añade el bot y márcalo **Administrator** (sin permiso para añadir admins).
   - Ya recibes mensajes en tiempo real (no hay ventana de 15 min).

Campo clave del trigger	Contenido típico
`chat.title`	Nombre del canal/grupo
`from.username`	Remitente humano
`text`	Mensaje escrito
`date`	Marca temporal UNIX

10 · Conectar servicios de IA (Hugging Face)

Paso	Detalle
Cuenta	Regístrate en `huggingface.co` (plan free).
Token	Perfil ▶ *Settings* ▶ *Access Tokens* ▶ **New token** (`Read + Inference`).
Módulo	+ ▶ *Hugging Face – Chat Completion*. Pega el token.
Modelo	Copia el repo-ID (ej. `deepseek-ai/deepseek-llm-7b-chat`) y pégalo en *Model*.
Límites free tier	~30 K tokens/mes → evita bucles de prueba excesivos.

11 · Diseña prompts estructurados

Rol	Uso recomendado	Ejemplo breve
assistant	Instrucciones de sistema: reglas, formato, tono.	"Devuelve solo JSON { nombre, pregunta } …"
user	Texto del usuario (mapeado desde `text`).	`{{Telegram.text}}`

Plantilla mínima (DeepSeek / GPT-like)

```
[
  { "role": "assistant", "content":
    "Eres un ayudante… Devuelve JSON con {nombre, pregunta}. …" },
  { "role": "user", "content": "{{text}}" }
]
```

> **Tip**: Pide siempre un formato fácil de procesar (JSON, CSV) para que el siguiente módulo pueda usar **Parse JSON** o rellenar celdas sin expresiones regulares complicadas.

12 · Procesamiento y control de calidad

1. **Parse JSON** módulo ▶️ Extrae `nombre` y `pregunta`.
2. **If / Router**
   - Si `nombre == "preguntar"` → Responde “Necesito tu nombre completo…”.
   - Else → Continúa con lógica de negocio (consultar base, generar respuesta…).
3. **Throttle / Error Handler** para evitar bloqueos si supera cuota de tokens.

13 · Routers y control de flujo

Elemento	Ícono	¿Qué hace?	Buenas prácticas
Router	∧ (Y-shaped)	Divide la ejecución en N ramas según reglas lógicas.	- Coloca justo tras el módulo que produce los datos a evaluar. - Usa **Labels** descriptivos (“sin nombre”, “con nombre”).
Condiciones	⚙️ en la línea	Filtra por `contains`, `does not contain`, `matches regex`, etc.	Comparar siempre campos simples (IDs, flags, cadenas cortas) para minimizar fallos.
Filtros en enlaces	Clic en el conector ▶️ *Set up a filter*	Evita módulos innecesarios avalanchas de operaciones.	Prefiere un solo router con varias ramas antes que routers anidados.
Tips			- Prueba cada rama con **Run once** enviando datos que cumplan / no cumplan la condición. - Documenta la lógica con **Notes** para compañeros de equipo.

14 · Procesamiento de texto y extracción de valores

Herramienta	¿Para qué sirve?	Cómo se usa rápido
Text Parser – Match Pattern	Extraer partes de un string mediante *expresiones regulares* (regex).	1. Selecciona el **texto origen** (p. ej. respuesta del LLM). 2. Pega la regex (`{"name":"([^"]+)","pregunta":"([^"]+)"}`) en *Pattern*. 3. El output se expone como `$1` , `$2` , …
Regex asistido	Pedir a ChatGPT "genera regex para capturar X y Y"	Valida en sites tipo *regex101.com* antes de usar.
Alternativas	- *Parse JSON* si el texto ya es JSON puro. - *Text functions* (split, trim) para cadenas sencillas.	

15 · Consultar bases de conocimiento y filtrar resultados

Escenario	Módulos sugeridos	Clave del éxito
Docs en Google Drive	1. *Google Docs – List documents* (en carpeta `Reportes`). 2. **Filtro**: `Name contains {{NombreCliente}}` . 3. *Google Docs – Get document content* usando `Document ID` .	Mantén convención de nombres ("plan {{Nombre}}") para que el filtro sea fiable.
Airtable / Datastores	Query `Filter by formula` o `ID == {{Nombre}}` .	Indexa por un campo único (email, phone, slug).
Objetivo	Minimizar llamadas: lista → filtra → procesa **solo** el documento pertinente.	

16 · Generar respuestas personalizadas con IA

Paso	Configuración	Recomendaciones
Modelo	Hugging Face *Chat Completion* `deepseek-ai/deepseek-llm-7b-chat` (o tu favorito)	Verifica limits de tokens mensuales (≈30 k en gratuito).
Prompt (rol = assistant)	- Contexto: "A continuación te paso el plan…" - Instrucciones: "Responde solo en JSON {respuesta, razon}…"	- Sé explícito en formato para facilitar el parseo. - Indica qué hacer si la pregunta no encaja con el plan.
Variables dinámicas	`{{Documento.textContent}}` → plan completo. `{{$2}}` → pregunta extraída.	Mantén los replacers lo más cortos posible (evita pegar gigabytes de texto).
Post-procesado	- *Procesa JSON* de la respuesta. - *Telegram Send message* con `respuesta`.	Implementa **handler de errores** por si el modelo devuelve texto mal formado.

17 · Publicar y monitorizar tu agente

Acción	Dónde se hace	Detalles clave
Activar ejecución inmediata	Editor ▶️ toggle **ON**	Trigger = Webhook/Telegram → se ejecuta al instante (no 15 min).
Historial de runs	Scenario ▶️ *Run history*	Revisa bulbos en rojo/amarillo; abre cada paso para ver el payload.
Errores y reintentos	*Incomplete executions*	Decide: *Re-run*, *Ignore* o *Fix data & replay*.
Límites	Plan free: 1 000 ops/mes & 5 min/run	Usa *Routers* y *Filtros* para no malgastar operaciones.

18 · Migrar de Hugging Face a OpenAI (ChatGPT)

Diferencia	Hugging Face (DeepSeek)	OpenAI ChatGPT
Conexión	*Hugging Face – Chat Completion* + token HF	*OpenAI – Create Chat Completion* + API Key
Model ID	`deepseek-ai/deepseek-llm-7b-chat` (u otro)	Ej.: `gpt-4o-mini`, `gpt-3.5-turbo`
Campos de salida	`mapable_message` → texto completo	`result` → texto completo
Precio	Gratuito hasta agotar créditos (≈30 k tokens)	Pago por tokens; desde ≈ $0.0005/1k tokens
Velocidad	Media	Muy rápida (latencia baja)

Pasos de sustitución rápida

1. Crea API Key en *platform.openai.com* ▶️ Projects ▶️ API Keys.
2. Añade módulo **OpenAI – Create Chat Completion** y pega la key.
3. Copia tu *prompt* existente (rol = assistant / user).
4. Cambia referencias: `result` en lugar de `mapable_message`.
5. Ajusta filtros/regex para leer desde `result`.

19 · Costes & mejores prácticas con la API de OpenAI

Consejos	Por qué importan
Modelo "mini" para prototipos	4o-mini cuesta ~⅓ que GPT-4o.
Top-p y temperature bajos (0.2–0.4)	Evitan respuestas aleatorias y repetidos errores de formato JSON.
Máx Tokens	Limita a 512–1024 para no sobrepagar.
Alertas de gasto	Billing ▶️ Usage limits → pon avisos a $2 y $3.
Rotar la API Key	Si compartes el escenario con otros editores, crea una key nueva y revoca la anterior.

Página de Notas del Tema

Esta página está pensada para que puedas anotar ideas clave, dudas y reflexiones importantes sobre el tema anterior.

Tema 4 · Relevance AI: Agente de IA que transcribe, resume y traduce tus reuniones

Resumen del tema

En este tema descubrirás cómo utilizar **Relevance AI** para construir, sin escribir código, un agente capaz de transcribir el vídeo de una reunión, extraer la información esencial, generar un acta bilingüe (español-inglés) y enviarla automáticamente por correo electrónico.

1 · Registro y planes

Opción	¿Qué hace?	Pistas rápidas
Continuar con Google	Crea la cuenta en un clic sin contraseñas adicionales.	Usa un correo corporativo si el agente va a actuar en nombre de la empresa.
Plan Free (100 créditos/día)	Permite probar todas las funciones básicas y la mayoría de modelos de IA.	• Los créditos se renuevan cada día. • Las llamadas a modelos *mínimo* consumen ≈ 0,23 créditos (modelo GPT-4o Mini).
Planes Pro / Business / Enterprise	Aumentan el límite de créditos y añaden Workforce, analítica avanzada y SLAs.	Calcula consumo: nº procesos × créditos × días. Así evitas sobrecostes.

2 · Panel de control principal

Sección	Para qué sirve	Comentario rápido
Templates	Galería de plantillas (escenarios preconfigurados).	Ideal para inspirarse o acelerar un proyecto.
Agents	Creador de agentes mediante *prompting*.	El agente decide cuándo y cómo usar tus Tools.
Tools	Flujos atómicos reutilizables (tareas concretas).	Piensa en "mini-zaps" que el agente invoca.
Workforce	Orquesta varios agentes en paralelo (solo planes de pago).	Útil para pipelines largos o roles especializados.
Knowledge Base	Tabla/carpeta con tus propios datos y archivos.	Sube PDFs, webs o CSVs para dar contexto factual.
Monitoring & Analytics	Métricas de uso y logs de llamadas.	Requiere plan Business+ para dashboards completos.
Snippets	Bloques de código opcionales (Python, JS…).	Potente, pero innecesario si buscas cero programación.

3 · Creación de una Tool

1. **New Tool ▶️ Describe en lenguaje natural**
    - El asistente *Inventor* hace preguntas aclaratorias y genera el esqueleto.
2. **Campos clave de la Tool**

Campo	Qué contiene	Buenas prácticas
Title	Nombre visible en el catálogo.	Empieza por verbo: “Generar acta…”, “Clasificar tickets…”.
Short description	Texto que el *agent* leerá para decidir si la llama.	Sé explícito: qué hace, con qué entradas y salidas.
Inputs	Parámetros que recibirá (texto, número, lista, archivo…).	Usa nombres sin espacios y en *snake_case* (ej. `transcripcion_reunion`).
Steps	Módulos encadenados (LLM, HTTP, Gmail, etc.).	Limita cada Tool a una responsabilidad clara.
Outputs	Datos devueltos al agente.	Devuelve solo lo necesario para evitar tokens extra.

4 · Variables dinámicas en prompts

- Referencia un **input** o la salida de un **step** con `{{ nombre_variable }}`.
- Ej.:

```
Analiza la transcripción "{{ transcripcion_reunion }}" y genera…
```

- Usa nombres autoexplicativos y documenta los formatos (ej. “texto largo”, “URL”, “JSON”).

5 · Módulo LLM (Large-Language-Model)

Ajuste	Opciones	Consejos de crédito & calidad
Provider	OpenAI, Google, Anthropic, Mistral…	Selecciona según coste y latencia en tu región.
Model	GPT-4o Mini, GPT-4o, Gemini 1.5 Flash…	Empieza con el *tier* económico; sube solo si la calidad no es suficiente.
Temperature	0 – 1	0-0.3 para respuestas deterministas (actas, informes).
Prompt	Instrucciones + contextos + style guide.	Incluye: formato de salida, idioma y restricciones de token.
Output variable	`prompt_completion` (editable)	Nómbrala con el resultado esperado: `acta_es_en`.

6 · Versionado, pruebas y despliegue

Acción	Menú	¿Qué hace?
Publish changes	Barra superior	Crea una nueva versión inmutable de la Tool.
Use ▶ Run Tool	Vista *Use*	Test manual con valores de ejemplo (sandbox).
Share ▶ Public link / Embed	Vista *Use*	Expón la Tool como widget externo o mini-app.
API tab	Vista *Use*	Copia el endpoint REST con Auth Token para integrarlo en otros sistemas.

7 · Buenas prácticas de diseño

- **Agrupa lógica**: una Tool = una función de negocio claramente definida.
- **Reduce pasos**: fusiona prompts (p. ej. *analiza* + *traduce*) para ahorrar créditos.
- **Controla tokens**: pide resúmenes breves y limita longitud de entrada con validaciones.

- **Nombra todo**: títulos, variables y versiones descriptivas facilitan el mantenimiento.
- **Reutiliza**: cualquier agente puede llamar a una Tool —preferible a duplicar lógica.

8 · Crear tu agente paso a paso

Inicio ▶️ **Agents** ▶️ *Describe your agent*

Paso rápido	Qué ocurre	Consejos
Escribe el objetivo en lenguaje natural	El asistente **Inventor** pregunta y genera un esqueleto con prompt, Tools y módulos.	Sé conciso: "Recibir vídeo → transcribir → generar acta bilingüe → enviar email".
Revisa el prompt generado	Define rol ("Eres un asistente de reuniones …") e instrucciones secuenciales.	Añade tu propia Tool con `using: Generar_acta_estructurada`.
Acepta o edita Tools sugeridas	El agente hereda todas las Tools listadas en el panel **Tools**.	Elimina las que no aporten; evita duplicar funciones que ya tienes en una Tool propia.
Publica	Crea versión *v1* del agente.	Cada *Publish changes* bloquea la versión: ideal para rollback.

9 · Panel de configuración del agente

Sección	Para qué sirve	Buenas prácticas
Prompt	Motor de comportamiento; puede usar variables `{{ }}`.	Mantén la lógica de alto nivel; delega tareas concretas a Tools.
Model	Selección del LLM que "razona".	Deja **Auto** para equilibrio coste-calidad; fuerza GPT-4o solo si imprescindible.
Tools	Lista de flujos que el agente puede invocar.	Marca "Let agent decide" en los campos de entrada para que los llene con contexto.
Knowledge	PDFs, webs o tablas que el agente puede consultar.	Adjunta documentación interna y referencia en el prompt ("Consulta tu base de conocimiento…").
Triggers	Eventos que lanzan la ejecución.	• Webhook (genérico). • Calendario/Drive/Email. • O bien dispara desde Make/Zapier vía API.
Variables	Texto/Número/JSON reutilizable en el prompt.	Útil para cabeceras fijas, disclaimers o firmas.
Advanced	Temperatura, mensaje de bienvenida, sub-agents.	Temperatura 0-0.3 para actas; evita encadenar agentes salvo necesidad extrema.

10 · Ejecución, depuración y créditos

- **Run ▶ New Task** → prueba manual con archivos o texto de muestra.
- **Logs**: cada paso muestra duración, input recortado y output; ideal para detectar prompts demasiado largos.
- **Créditos**: barra inferior indica los que quedan (p. ej. 11/100). Un run típico con GPT-4o Mini + Tool suele costar de 4-8 cr.
- **Itera rápido**:
  1. Ajusta prompt o quita módulos redundantes.
  2. *Publish changes* (crea v2).
  3. Re-Run y compara salidas.

- **Version rollback**: botón ⋯ ▶ **Revert to…** por si la nueva versión falla.

11 · Buenas prácticas específicas de agentes

1. **Divide y vencerás**: transcripción, análisis y envío deben vivir en Tools distintas; el agente solo orquesta.
2. **Usa "Let agent decide"** para inputs variables (archivos, destinatarios) y evita hard-codear datos.
3. **Ahorra créditos**: junta acciones compatibles en un único prompt y escoge el modelo más barato que cumpla calidad.
4. **Formatea en la Tool, no en el agente**: así reutilizas la misma acta en otros flujos.
5. **Tests breves**: recorta videos a 1-2 min al depurar; sube duración cuando el flujo sea estable.

Página de Notas del Tema

Esta página está pensada para que puedas anotar ideas clave, dudas y reflexiones importantes sobre el tema anterior.

Tema 5 · n8n: Automatiza el control de gastos de tu empresa

Resumen del tema

En este tema aprenderás a instalar y ejecutar n8n en tu propio equipo mediante Docker, conectar tu cuenta de Google con OAuth 2.0, y dominar la interfaz de la herramienta para crear y gestionar workflows. Verás cómo configurar un disparador que detecte nuevas filas en Google Sheets, aplicar lógica condicional con nodos **IF**, actualizar celdas automáticamente cuando se superen ciertos umbrales de gasto y, por último, probar, activar y monitorizar tu flujo desde el registro de ejecuciones. Todo ello sentará las bases para, en las siguientes clases, incorporar capacidades de inteligencia artificial dentro de tus automatizaciones.

---

1 · ¿Qué es n8n y cómo puedo usarlo?

Opción	¿Qué hace?	Cuándo elegirla
Auto-hosted (código abierto)	Descargas el código de GitHub o descargas la imagen Docker y la ejecutas en tu máquina o servidor.	Necesitas control total, no quieres costes recurrentes.
n8n Cloud (servicio gestionado)	La empresa se encarga del hosting, backups y actualizaciones.	Prefieres pagar por simplicidad y soporte.
Repositorio GitHub	Acceso al código fuente (TypeScript) y a los tickets de soporte de los usuarios.	Revisar cambios, abrir PR o adaptar el programa.

2 · Instalación local paso a paso (Docker Desktop)

Paso	Comando / Acción	Notas rápidas
1. Instalar **Docker Desktop**	Descarga desde docker.com y sigue el asistente	Requiere hardware de virtualización activado.
2. Obtener la imagen	`docker pull n8nio/n8n` *(o botón **Pull** en Docker Desktop)*	+100 M descargas — es la vía oficial.
3. Lanzar el contenedor	`docker run --name n8n -p 5678:5678 -v ~/n8n_data:/home/node/.n8n ...`	Usa el **mismo** puerto externo e interno (5678 por defecto).
4. Variables de entorno más comunes	`N8N_BASIC_AUTH_ACTIVE=true` `N8N_BASIC_AUTH_USER=admin` `N8N_BASIC_AUTH_PASSWORD=******` `WEBHOOK_URL=http://localhost:5678/`	Protege la instancia y evita registrar la cuenta a mano.
5. Persistencia	Monta un volumen para que los workflows sobrevivan a reinicios.	Cualquier ruta local sirve; redirígela a tu carpeta de proyectos.

3 · Interfaz principal de n8n

Sección	¿Para qué sirve?
Workflows	Lista, busca y etiqueta tus flujos de automatización.
Credentials	Gestiona tokens secretos para apps externas.
Executions	Historial, logs y estado (éxito / error).
Templates	Copia-pega plantillas de la galería oficial. Selecciona → "Use template" → Ctrl + V en tu canvas.
Insights	Métricas de uso: nº de ejecuciones, ratio de error, tiempo ahorrado.
Help	Docs, tutoriales y soporte comunitario.

4 · Crear un Workflow desde cero

Acción	Ruta rápida	Detalle
Crear	Overview ▶️ **Create workflow**	Se abre un canvas en blanco.
Renombrar y etiquetar	Click en el título	Usa tags para filtrar en la vista Workflows.
Añadir nodo	Doble click en canvas *o* " + "	Empieza por un **Trigger**; después arrastra líneas para encadenar acciones.
Notas (Sticky Notes)	Click derecho ▶️ Add Node ▶️ **Notes**	Documenta pasos, requisitos o URLs.
Ejecuciones	Pestaña **Executions** (en el editor)	Ejecuta manualmente, inspecciona payloads y errores paso a paso.

5 · Tipos de nodos (categorías clave)

Categoría	Ejemplos destacados	¿Qué hacen?
Trigger (Inicio)	**Manual Trigger**, **Cron**, **Webhook**, "New Row in Google Sheets"	Disparan el flujo cuando ocurre un evento o a una hora programada.
Action / Integration	Gmail → "Send Email", Slack → "Post Message", Google Drive → "Upload File"	Interactúan con APIs externas; la mayoría requieren **Credentials**.
Transform	**Set**, **Merge**, **Item Lists**, **IF**, **Filter**	Manipulan datos: formateo, condicionales, deduplicados, mapeos.
Flow Control	**Split In Batches**, **Wait**, **Loop**, **Merge**	Dirigen la ruta de ejecución, sincronizan ramas o iteran arrays.
Core	**Webhook**, **Execute Command**, **Code (JavaScript/Python)**	Funciones nativas potentes; permiten lógica avanzada o integraciones propias.
Human-in-the-Loop	Gmail / Outlook "Send & Wait for Reply", Slack / Telegram "Send Message"	Detienen el flujo hasta que un usuario responda (aprobaciones, revisiones).

> **Pista rápida**: si no encuentras un nodo, escribe en el buscador del panel **exactamente** la app o acción; n8n ofrece sinónimos y filtros por categoría.

6 · Conectar apps externas de Google (OAuth 2.0)

> **Ruta rápida:** Node ▶ *Credentials* ▶ **Create new** ▶ Sigue la guía

Paso	¿Dónde?	Clave para no perderte
1. Crear proyecto	Google Cloud Console ▶️ "Select project" ▶️ **New Project**	Un proyecto por integración mantiene permisos acotados.
2. Pantalla de consentimiento	APIs & Services ▶️ OAuth consent screen	Marca *External* → añade sólo correos de prueba si no vas a publicar.
3. Crear ID de cliente	APIs & Services ▶️ Credentials ▶️ **Create** ▶️ OAuth Client ID (Web)	Copia **Client ID** y **Client Secret** al instante; Google ya no mostrará el secret pasado un tiempo.
4. URIs de redirección	Campo "Authorized redirect URIs"	Usa la que te muestra n8n (`http://localhost:5678/rest/oauth2-credential/callback`).
5. Habilitar APIs	APIs & Services ▶️ **Enable APIs & services**	Activa sólo lo necesario: Google Sheets, Drive, Gmail, etc.
6. Autorizar cuenta	Botón **Connect OAuth** en n8n	Selecciona la cuenta → Concede permisos granulares.

7 · Nodo IF — lógica condicional

Ajuste	Valor clave	Comentario
Rules combinadas	`AND` / `OR`	Selecciona `OR` si cualquiera de las condiciones debe cumplirse.
Type coercion	*Enable Type Conversion*	Fuerza que "12345" se trate como número para comparaciones.
Operadores numéricos	`isGreaterThan` , `isLowerThan` , `between`	No distingue decimales/enteros; usará JS `Number()` .
Always Output Data	*Off* por defecto	Actívalo si necesitas una salida vacía con estructura fija.

8 · Probar y publicar el workflow

Acción	Botón	Qué vigilar
Guardar	💾 *Save*	Sin guardar no podrás activar.
Test Workflow	▶️ (en modo *test*)	Usa los últimos datos recuperados; no se agenda el trigger.
Activate	**Activate** (switch 🟢)	El trigger se ejecuta según el intervalo; aparece banner de advertencia.
Executions log	Pestaña **Executions**	Inspecciona cada nodo, tiempo de ejecución, payload y errores.
Desactivar	switch ⚪	Detiene el cron pero conserva el workflow.

> **Depuración rápida:** si ves filas "incorrectas" revisa la **Key Column** y duplica el workbook para no afectar datos reales durante pruebas.

Página de Notas del Tema

Esta página está pensada para que puedas anotar ideas clave, dudas y reflexiones importantes sobre el tema anterior.

Tema 6 · Introduce IA en tus automatizaciones con n8n

Resumen del tema

En este tema aprenderás a integrar inteligencia artificial local en tus flujos de n8n: primero instalarás modelos LLM con **Ollama** en tu propio equipo y los conectarás mediante credenciales para que n8n los invoque; después verás cómo usar **Basic LLM Chain** para generar correos adaptados a distintas situaciones (gastos normales o superiores a 10 000 €), filtrar la respuesta y enviarla automáticamente con el nodo **Gmail**; finalmente crearás un **formulario web** en n8n para que los empleados registren sus gastos, añadiendo cada registro a Google Sheets y desencadenando el workflow completo sin depender de servicios externos ni costes por llamada.

---

1 · Modelos LLM locales con Ollama

Paso	Comando / Opción	¿Qué hace?
Instalar	`ollama run <modelo>`	Descarga y deja listo el LLM (p. ej. `deepseek:7b`) dentro de tu equipo.
Ubicación por defecto	`http://localhost:11434`	API REST que exponen todos los modelos descargados.
Elegir tamaño	`*-tiny* ≈ 1 GB` / `*-7b* ≈ 5 GB` / `*-20b* ≈ 20 GB`	Ajusta consumo de RAM y velocidad. Elige el más grande que tu hardware permita sin ralentizar.
Arranque automático	Icono de la llama en la bandeja del SO	Verifica que Ollama está corriendo antes de lanzar flujos.
Actualizar modelo	`ollama pull <modelo>`	Re-descarga la versión más reciente.

2 · Conectar Ollama a n8n

Inicio ▶️ **Credentials** ▶️ *New* ▶️ **Ollama**

Campo	Valor típico	Comentario rápido
Base URL	`http://host.docker.internal:11434`	Necesario cuando n8n corre dentro de Docker. En instalaciones "nativas" usa `http://localhost:11434`.
Test	*¡Green!*	Confirma que el contenedor ve la API de Ollama.

3 · Nodo Ollama Chat Model

Pestaña	Opción	¿Para qué sirve?
Parameters	**Model**	Lista desplegable con todos los modelos descargados.
	Temperature	Variabilidad en la respuesta (0 – 1). Valores bajos → respuestas consistentes.
	Top-p / Top-k	Filtrado de tokens; afina creatividad frente a precisión.
	Threads	Nº de hilos CPU (mejora velocidad).
	GPU Layers	Número de capas a descargar a GPU (si tienes soporte CUDA/Metal).
Notes	Texto libre	Útil para documentar el nodo en el canvas.

> 💡 **Atajo**: Duplica un nodo ya configurado para reutilizar credenciales y modelo.

4 · Cadena básica (Basic LLM Chain)

Sección	Clave	Explicación
Input Source	*Define below*	Permite escribir el *prompt* directamente.
Prompt	Texto + Variables `{{$json.field}}`	Describe el rol y añade los valores de tu flujo.

Buenas prácticas de prompt

1. Define **rol + objetivo** en la primera frase.
2. Lista variables como "Nombre: {{name}}" para que el modelo las reemplace.
3. Añade instrucciones «**Sé breve y profesional**» si el modelo es pequeño (< 10 B parámetros).

5 · Enviar correos con el nodo Gmail

Campo	Uso típico	Detalles
To	`{{$json.email}}`	Dirección proveniente del flujo.
Subject	Texto + variable fecha (`{{$json.date}}`)	Ej. "Reporte procesado – {{$json.date}}".
Message	Salida del LLM (`{{$json.message}}`)	Puede ser **text/plain** o **text/html**.
Append n8n Attribution	*Off*	Quita la firma automática "sent via n8n".
Attachments	Lista de rutas o binarios	Adjunta archivos generados previamente.

6 · Formularios web (n8n Form)

Parámetro	Propósito	Sugerencia
Publish URL	Enlace temporal (testing) y productivo	Expuesto localmente; usa túnel o servidor público si lo compartirás fuera.
Auth	None / Basic / Token	Añade autenticación si manejas datos sensibles.
Field Types	Text, Email, Number, Date, Hidden	Campos *Hidden* ideales para IDs (`{{date.now}}`).
Submit Trigger	Activa workflow	El formulario se comporta como nodo disparador.

7 · Control de flujo y filtrado

Nodo	Caso de uso	Ejemplo rápido
IF	Ramas condicionales	`{{$json.total}} > 10000` → ruta de alerta.
Set / Transform	Renombrar o crear campos	Construir objeto limpio para el LLM.
Merge (By Index)	Combinar salidas paralelas	Une resultados de IA y datos originales.

8 · Rendimiento y costes

- **Locales = €0** : sin llamadas a API externas ni cuotas por token.
- **Hardware** : 7 B ≈ 5 GB RAM; 20 B requiere maquinaria de gama media.
- **Tiempo de respuesta** : aumenta linealmente con tamaño de modelo y complejidad de prompt.
- **Escalado** : para producción, considera mover Ollama a un servidor con GPU y exponer la URL en red interna.

Página de Notas del Tema

Esta página está pensada para que puedas anotar ideas clave, dudas y reflexiones importantes sobre el tema anterior.

Tema 7 · n8n: Construye Agentes de IA con memoria, herramientas y tus datos

Resumen del tema

En este tema aprenderás a convertir n8n en un entorno para crear agentes de IA completos — conversacionales, autónomos y conectados a tus propios datos. Partirás del nodo **Agent**, verás cómo elegir un modelo LLM compatible con herramientas (local con Ollama o vía API con OpenAI), añadir memoria para mantener el contexto, y equipar al agente con **tools** dinámicas (Gmail, Airtable, HTTP, etc.) que ejecuta según necesite. Después integrarás una base de datos real en Airtable, redactarás un **prompt de sistema** que delimite intenciones y reglas, depurarás las cadenas de pensamiento para ofrecer respuestas limpias y, por último, publicarás el chat tanto como URL pública como embebido en una web, aplicando buenas prácticas de seguridad, coste y despliegue.

1 · Estructura de un agente de IA

Componente	¿Qué es?	Buenas prácticas
Trigger	Nodo que inicia el workflow (Chat, e-mail, webhook, etc.).	El chat trigger es el más común, pero **cualquier nodo** puede ser la entrada.
Agent node	Recibe la entrada, la pasa al modelo, gestiona memoria y decide qué herramienta usar.	Añade un solo nodo Agent por agente para mantener la lógica centralizada.
Chat Model	Modelo LLM que decide las acciones.	Debe **soportar tools/functions**; de lo contrario no podrá invocar nodos.
Memory	Guarda contexto entre mensajes.	Usa "Simple Memory" para prototipos; bases de datos (Mongo, Postgres, Redis) para producción.
Tools	Conjunto de nodos de acción que el agente puede invocar dinámicamente.	Añade solo las herramientas necesarias; cada nodo puede exponer múltiples operaciones.

2 · Selecciona el modelo de IA (Chat Model)

Modelo	Tipo	Cuándo usarlo	Ventajas / Retos
Qwen3 · 8 B (Ollama)	Local, gratuito	POCs, portátiles sin GPU	Sin coste, sorprendentemente preciso; más lento en CPU.
Qwen3 · 14 B (Ollama)	Local, gratuito	Casos con razonamiento medio y GPU disponible	Mejor calidad que 8 B; alto consumo de RAM/CPU.
GPT-4o mini (OpenAI)	API, pago	Flujos en prod. con latencia baja	Muy rápido y soporta tools; coste por llamada.
GPT-4 Turbo/Enterprise	API, pago	Agentes críticos o complejos	Razonamiento superior; coste mayor.

Tips rápidos

- Requiere que el modelo anuncie “tools/functions” en la API.
- En Ollama activa la GPU si existe (`GPU: auto`); en CPU limita la concurrencia.
- Ajusta `maxTokens` , `temperature` , etc., desde el nodo para afinar respuestas.

3 · Añade memoria al agente

Opción de memoria	Persistencia	Escenario típico
Simple Memory	RAM (almacenada en el workflow)	Chats internos, prototipos, ≤ 10-20 mensajes.
Redis Cache	RAM + disco, baja latencia	Bots de atención con alta simultaneidad.
MongoDB / Postgres	Disco, consultas ricas	Historial a largo plazo, reporting y GDPR.

Parámetros clave

- **Session ID** (viene del trigger): agrupa mensajes de un mismo usuario.
- **Context Window Length**: nº de interacciones a recuperar. Suele bastar con 10-15 para mantener contexto.

4 · Configura herramientas (Tools)

1. **Añade nodo Tool** → elige el servicio (Gmail, Airtable, HTTP Request, Sheets, etc.).
2. Para cada **parámetro** marca **“Let the AI decide”** si quieres que el modelo lo rellene en tiempo de ejecución.
3. Describe la herramienta en el campo **Description**: el agente leerá esta instrucción para saber cuándo usarla.
4. Conecta varias herramientas al mismo Agent node; el modelo decide el orden y los datos que pasa a cada una.

Ejemplo rápido – Gmail Tool	Valor
Resource	`messages`
Operation	`getMany` (leer) / `send` (enviar)
Parámetros delegables	`limit`, `to`, `subject`, `message` → **Let the AI decide**

5 · Prueba y depura tu agente

- **Chat panel**: escribe consultas y observa cada paso en la barra lateral de ejecución.
- **Highlights de color**
  - Verdes = memoria actualizada.
  - Azules = llamadas al modelo LLM.
  - Morados = invocaciones de herramientas.
- **Reinicia sesión** para comenzar un chat limpio (genera nuevo Session ID).
- **Logs completados** muestran el prompt interno, la decisión de la herramienta y el output devuelto: ideal para troubleshooting.
- **Cost control**: en modelos de pago mide tokens; en locales observa consumo de CPU/GPU.

6 · Conecta bases de conocimiento externas

Paso	Qué debes hacer	Detalles clave
1. Elige la fuente	Airtable, Sheets, Postgres, etc.	Necesitas un endpoint o token con permisos de lectura (y opcionalmente escritura).
2. Crea la credencial	Settings ▶️ Credentials ▶️ "Personal Access Token"	Concede solo los scopes mínimos: `data.records:read/write`, `schema:read`.
3. Añade la Tool	Agent ▶️ Add Tool ▶️ *Airtable Tool*	Define: • `resource` = `records` o `schema` • `operation` = `search`, `get`, `create`, `update`. • Para filtros usa "Let the AI decide" y la IA construirá fórmulas Airtable.
4. Opciones avanzadas		• Crea **dos** tools sobre la misma base: una para `schema` (describe campos) y otra para `search`. • Limita `maxRecords` para evitar descargas masivas.

7 · Diseña un prompt de sistema robusto

Coloca las reglas inmutables en **System Prompt**. Ejemplo genérico:

```
Eres un asistente conectado a la tabla <NombreTabla>.
- Intenciones:
  1️⃣ Seguimiento: si el usuario da <campo clave>, busca y responde solo estado y fechas.
  2️⃣ Métricas: si solicita totales o promedios, usa funciones de agregación.
- Reglas:
  • Consulta siempre el esquema antes de cada operación.
  • No escanees todos los registros.
  • No reveles datos personales ni inventes información.
```

Buenas prácticas

- Separa claramente *intenciones* (lo que puede hacer) de *reglas* (lo que nunca debe violar).
- Actualiza el prompt si cambian columnas o tipos de consulta.
- Añade un pre-prompt al usuario ("Recuerda consultar el esquema…") si necesitas forzar un comportamiento en cada petición.

8 · Limpia la respuesta antes de mostrarla

Acción	Nodo recomendado	Expresión regular sugerida
Eliminar chain-of-thought	**Set / Edit Fields** → `Value=output.replace(/<zinc>[\s\S]*?<\/zinc>/g, '')`	`/<zinc>[\s\S]*?<\/zinc>/g`
Anonimizar PII	**Code** (JavaScript) o **Set**	Sustituye emails, teléfonos, etc., antes de enviar al usuario.

Coloca este nodo **después** del Agent y **antes** del canal de salida (Chat, e-mail, API).

9 · Publica tu chat

Modalidad	Cómo activarla	Uso típico
Hosted Chat	Chat node ▶️ "Make public" ▶️ URL	Soporte interno rápido; requiere exponer tu instancia n8n.
Embedded	Chat node ▶️ "Embed" ▶️ copia el `<script>`	Integra en cualquier web o portal sin backend adicional.

Tips

- Activa el workflow (⚡) para que la URL funcione.
- Personaliza `Initial Message`.
- Considera Nginx, ngrok o un VPS si tu n8n corre en localhost.

10 · Seguridad y despliegue

1. **Rotación de tokens**: usa variables de entorno y renueva claves cada 3-6 meses.
2. **Rate limiting**: limita llamadas al modelo o base de datos para evitar abusos.
3. **Monitoriza logs**: revisa el Execution Log de n8n; exporta a Grafana/ELK si es producción.
4. **Cost control** (modelos de pago): calcula `tokens * precio` por llamada y fija alertas.
5. **Fallback**: configura un mensaje de error amigable si la herramienta devuelve errores.

Página de Notas del Tema

Esta página está pensada para que puedas anotar ideas clave, dudas y reflexiones importantes sobre el tema anterior.

Tema 8 · RAG en Agentes de IA con **n8n**

Resumen del tema

En este tema aprenderás a construir un flujo completo en n8n que permita aplicar la técnica de Retrieval Augmented Generation (RAG) sobre tus propios documentos: detectar y descargar automáticamente contratos subidos a Google Drive, extraer su texto con OCR, fragmentarlo y convertirlo en *embeddings*, almacenarlos en una base de datos vectorial (Pinecone) y, finalmente, crear un agente de IA con un chat público que consulta esa base para responder preguntas personalizadas sobre las cláusulas, participantes y detalles de cada contrato, ya sea usando modelos locales vía Ollama o alternando a modelos de OpenAI para mayor precisión y velocidad.

1 · ¿Qué es RAG?

Concepto	¿En qué consiste?	Pistas prácticas
RAG – Retrieval Augmented Generation	El agente convierte la pregunta en *embeddings*, recupera fragmentos de contexto de una base vectorial y genera la respuesta usando ambos.	Ideal si tu base de conocimiento es grande y el LLM no puede procesarla entera.
Embeddings	Vectores numéricos que capturan el significado semántico de palabras/frases.	Palabras “gato” y “felino” quedan cerca en el espacio vectorial; “gato” y “coche” aparecen lejos.
Base de datos vectorial	Guarda los embeddings y permite búsquedas por similitud (cosine, dot-product, etc.).	Pinecone, Weaviate, Chroma son las más usadas.
Flujo genérico en n8n	1) Trigger → 2) Descarga → 3) OCR/ETL → 4) Split → 5) Embeddings → 6) Upsert vectorial → 7) Chat agent.	Mantén los nodos desacoplados para reusar componentes.

2 · Detectar y descargar nuevos documentos

Paso	Nodo n8n	Opción clave	Comentario rápido
Trigger	*Google Drive → Watch for file changes*	- **Change Type**: *File Created* - **Folder**: ID o ruta de la carpeta a vigilar. - **Poll Interval**: ajusta a tu SLA; ≥ 5 min en producción para reducir costes.	También permite filtrar por MIME (limita a `application/pdf` si solo quieres PDF).
Descarga	*Google Drive → Download file*	- **File ID**: usa la referencia `{{$json["id"]}}` del trigger.	Produce un binario `data` con el PDF listo para OCR.

3 · Extraer texto con OCR

Nodo	Configuración mínima	Resultado
Extract from File	- **Mode**: *Extract text from PDF* (sirve para imágenes, DOCX, etc.)	Devuelve el texto plano en `data.text`.
(opcional) **Edit Fields / Set**	Selecciona solo `data.text` y bórralo del binario si no lo necesitas.	Facilita que los nodos posteriores manipulen menos carga inútil.

4 · Dividir texto en chunks

Estrategia	Nodo recomendado	Parámetros sugeridos
División semántica recursiva	*Recursive Character Text Splitter*	- **Chunk Size**: `1000` – `1500` caracteres. - **Chunk Overlap**: `100` – `200` caracteres para no perder contexto.
Tokens fijos	*Token Text Splitter*	Útil si tu modelo de embedding limita por tokens, no por caracteres.
Por líneas o párrafos	*Character Text Splitter*	Pon **Separator** a `\n\n` (doble salto) para párrafos.

5 · Embeddings y bases vectoriales

Decisión	Recomendación	Detalle
Algoritmo de embeddings local (sin coste)	`nomic-embed-text (768 dims)` vía **Ollama**	↓ latencia, 0 €; descarga (~200 MB) con `ollama run nomic-embed-text`.
Algoritmo de embeddings SaaS	OpenAI Embeddings, Cohere, etc.	Mejor cobertura idiomática y calidad; conlleva coste por 1 K tokens.
Base de datos vectorial	**Pinecone** (plan free)	Serverless, escalable; índice = "base de datos", documento = "registro". Otros: Weaviate (self-host), Chroma (local).
Métrica	Cosine	Mantén coherente métrica ↔ embeddings; la mayoría usa coseno.
Dimensión	Debe coincidir con el modelo (p. ej. `768`)	Configúrala al crear el índice; no es editable después.

6 · Configurar Pinecone en n8n

Componente	Pasos clave	Buenas prácticas
Credenciales	1. Crea API Key en la consola Pinecone. 2. En n8n → Credentials → Pinecone: pega la key.	Trata la key como secreto; usa variables de entorno en n8n Self-host.
Crear índice	Desde la consola Pinecone: - **Name**: `tu-proyecto-vector` - **Dimension**: *igual a embeddings* - **Metric**: *cosine* - **Environment/Region**: cumple tus requisitos de residencia de datos.	El plan gratuito permite hasta 1 índice (sujeto a cambios).
Nodo Pinecone → Upsert	- **Index**: tu índice. - **Batch Size**: ≤ 200 (límite API).	Envía arrays `[{id, values, metadata}]`; id único por chunk.
Embeddings dentro del nodo	Selecciona **Embeddings: Ollama** → **nomic-embed-text**.	Delega la generación para no pasar vectores manualmente.
Default Data Loader	*Load data from previous step*; Schema → elige `text`.	Evita cargar binarios u otros campos que no irán a la base.

7 · Probar y activar tu workflow de ingesta

Paso	Qué revisar en el panel de ejecución	Éxito si…
Test workflow	Se ejecutan todos los nodos en orden.	Cada nodo muestra ✓ verde.
Splitter	Aparecen `chunk_0`, `chunk_1` …	Texto ≤ chunk size y con solape correcto.
Embeddings	Campo `values` con 768 números (ej.)	La longitud coincide con la dimensión del índice.
Pinecone Upsert	Respuesta `{"upsertedCount": n}`	`n` = nº de chunks.
Activación	Interruptor en “ON”.	El log registra nuevas ejecuciones al subir archivos.

Mejor práctica: Ajusta el intervalo del trigger (≥ 5 min en producción) para no sobrecargar la API de Drive ni tu plan de n8n.

8 · RAG en el agente (Vector QA Tool)

Elemento	Ajustes esenciales	Buenas prácticas
Vector QA Tool	**Max Documents** = 3–6 **Description** = breve uso (ayuda al agente).	Limita docs para no desbordar contexto.
Vector Store	Pinecone, modo *Get Documents*, índice correcto.	Solo lectura: evita sobreescribir tu índice.
Embeddings	Mismo modelo (`nomic-embed-text`) usado al indexar.	Consistencia = mejor recall.
LLM para re-ranking	Igual que el agente o uno ligero tipo `gpt-4o-mini`.	Reduce costes si usas un modelo más pequeño aquí.

9 · Pruebas de la conversación

Caso	Pista en la ejecución	Qué hacer si falla
No consulta Pinecone	Vector QA Tool no se llama.	Refuerza prompt: “Siempre consulta la base de conocimiento antes de responder.”
Respuestas vacías	`documents: []` en la salida.	Revisa split size/overlap o aumenta embeddings quality.
Lentitud	Nodo modelo > 10 s	Cambia a modelo más pequeño o baja `maxDocuments`.
Hallazgos irrelevantes	Texto recuperado no coincide con la pregunta.	Ajusta embedding model o métrica; prueba re-ranking con modelo mejor.

10 · Cambiar de modelo (local vs. OpenAI)

Escenario	Modelo	Pros	Contras
Demo rápida	`gpt-4o-mini`	Latencia baja, buena calidad.	Datos salen a la nube; coste por 1 K tokens.
Razonamiento avanzado	`o3-mini` (OpenAI)	Mejor comprensión de peticiones complejas.	Más caro y algo más lento.
Offline / privacidad total	`qwen-8b`, `llama3-8b` via Ollama	0 € y datos on-prem.	Requiere CPU/GPU local potente.

> **Tip:** Tras cambiar de LLM, reinicia el chat para limpiar la memoria de la sesión y asegurar que el nuevo modelo se use desde el primer mensaje.

Página de Notas del Tema

Esta página está pensada para que puedas anotar ideas clave, dudas y reflexiones importantes sobre el tema anterior.

Tema 9 · Agentes de Voz con n8n + ElevenLabs

Resumen del tema

En este tema aprenderás a dotar a tus agentes de IA creados en n8n de capacidades de voz reales mediante ElevenLabs: explorarás el panel “Conversational”, crearás un agente desde plantilla, escribirás su prompt y le añadirás una **Custom Tool** que lanza un webhook POST a tu workflow; después verás cómo exponer tu n8n local con Cloudflare Tunnel, mapear la pregunta recibida, aplicarle RAG y responder al agente por el nodo **Respond to Webhook**. Para optimizar coste y velocidad sustituirás GPT-4 por modelos gratuitos Google Gemini, ajustarás temperatura y prompt, y activarás el flujo en producción. Finalmente, descubrirás las dos formas de publicar el asistente: incrustar el widget de voz en cualquier página web y asignarle un número telefónico vía Twilio, de modo que clientes o empleados puedan consultarlo tanto desde el navegador como mediante una llamada.

---

1 · Flujo de llamadas de principio a fin

1. **Usuario habla** → Llega audio (o llamada telefónica) a ElevenLabs.
2. **ElevenLabs Agent**
    - Transcribe y decide si necesita datos externos.
    - Si hace falta, dispara un **Custom Tool** → Webhook POST.
3. **n8n Webhook Trigger**
    - Recibe `question` (o payload definido).
    - Orquesta RAG, DBs, API…
    - Devuelve la `answer` mediante **Respond to Webhook**.
4. **ElevenLabs** convierte texto → voz y contesta al usuario en tiempo real.

2 · Panel “Conversational” de ElevenLabs

Sección	¿Qué ves?	Para qué sirve
Dashboard	Minutos, costes, LLM usage	Vigila cuotas gratuitas y picos de tráfico.
Agents	Lista y plantillas (Soporte, Ventas…)	Crea o duplica agentes; ideal empezar con una plantilla.
Call History	Registro completo	Revisión de transcripciones para QA.
Knowledge Base	URLs, PDF, texto	Añade contexto ligero; para RAG profundo usa n8n.
Phone Numbers	Bring-your-own-number	Vincula números reales para llamadas entrantes / salientes.
Calls	Programar llamadas	Automatiza recordatorios o briefing diario.
Settings	Webhooks, secrets, límites	Webhook start/end, token API, concurrencia.

3 · Crear un agente de voz

Campo	¿Qué hace?	Tip clave
Name	Lo que ve el usuario	Ej. *Asistente Contratos*.
Languages	Idiomas que detecta y habla	Añade varios para multilingüe.
Greeting	Primer mensaje	Personaliza tono y menciona nombre si procede.
System Prompt	Personalidad + límites	Describe rol, tono, temas prohibidos; apóyate en generador automático y luego edita.
LLM	Modelo subyacente	Gemini 2 Flash (o GPT-4o, Claude-3…); revisa costes.
Temperature	Creatividad 0-1	0 = determinista; 0.7-0.9 suena más natural en voz.
Max Tokens	Longitud máxima	−1 para "sin límite razonable".
Knowledge	Docs para consultas simples	Deja vacío si usarás RAG externo.
Tools	Llamadas externas	Añade **Custom Tool** → **Webhook** (POST).
Voice	Locutor/a y ajustes	Elige voz, velocidad, pronunciación y latencia.
Advanced / Security	Timeouts, IP allowlist, concur.	Corta silencios > 7 s, límite 300 s por call, etc.
Widget	Snippet `<script>`	Incluye el agente de voz en tu web en un clic.

4 · Definir la Custom Tool (Webhook)

Paso	Valor recomendado
Name	`consulta_n8n`
Method	`POST`
URL	Endpoint HTTPS público de tu Webhook Trigger.
Request Schema	`{ "question": "{{user_message}}" }` (ajusta variables)
Response Handling	Espera respuesta JSON con `answer` para leer en voz alta.

> 🔍 **Prueba rápida**: En el editor del agente usa el "Test Tool" para enviar `"¿Qué dice la cláusula de confidencialidad?"` y confirma que n8n devuelve texto adecuado.

5 · n8n: nodos mínimos

Nodo	Ajustes clave	Comentario
Webhook (Trigger)	Method: **POST** · Respond: **Using Respond to Webhook**	Recibe `question`.
→ LLM / Function / RAG	Implementa búsqueda y genera `answer`	Usa tu receta RAG habitual.
→ Respond to Webhook	Return: **Input Data** o `{{ $json.answer }}`	Envía texto a ElevenLabs.

Pro-Tip: Coloca un **Set** entre medias para normalizar la salida:

```
{
  "answer": {{$json["generated_text"]}}
}
```

6 · Exponer n8n local a Internet con Cloudflare Tunnel

Acción	Comando / GUI	Notas
Descarga imagen	`docker pull cloudflare/cloudflared`	Solo una vez.
Lanza túnel	`docker run -d --name cf_tunnel ...` (script del curso)	Mapea `http://localhost:5678` → URL pública *.trycloudflare.com*.
Copia URL	Ver logs o Docker Desktop	Sustituye `localhost:5678` en tu Webhook.
Re-start	Si 502/timeout, `docker restart cf_tunnel`	Genera nueva URL.

> **Seguridad mínima**:
>
> - Habilita credenciales de n8n (usuario/contraseña o auth header).
> - Coordina con TI antes de exponer endpoints en producción.

---

7 · Buenas prácticas rápidas

- **Mantén la llamada humana**: saluda, confirma entendimiento y avisa si vas a tardar en procesar.
- **Gestión de errores**: si n8n devuelve fallo, responde con disculpa genérica y ofrece repetir.
- **Coste bajo control**: reduce temperatura y tokens, y usa filtrado de intents para invocar RAG solo cuando sea necesario.
- **Logs**: revisa `Call History` y guarda métricas en tu propio Data Lake si necesitas análisis avanzado.
- **Escalabilidad**: el webhook puede enrutar a colas o microservicios si esperas alto volumen.

8 · Configura la Custom Tool (Webhook) en ElevenLabs

Campo	Valor típico	¿Para qué sirve?
Name	`N8n` (o similar)	Identifica la integración externa.
Descripción	*"Usa esta herramienta cuando necesites consultar la BD de contratos"*	Guía al LLM sobre cuándo llamarla.
URL	`https://<tu-túnel>.trycloudflare.com/webhook/<id>`	Endpoint POST público de n8n. Comprueba en navegador que devuelve "supports POST".
Timeout	90 s *(ajustable)*	Tiempo máximo que esperará ElevenLabs la respuesta.
Body Params	`pregunta: {{user_message}}`	El agente extrae la pregunta y la envía como JSON.

> **Consejo**: si tu modelo/flujo tarda < 30 s reduce el timeout para liberar recursos.

9 · Ajusta tu workflow de n8n para entrada por voz

1. **Webhook (Trigger)** – POST, "Respond using Respond to Webhook".
2. **Set / Function** – `prompt = $json.pregunta` (mapea el parámetro).
3. **LLM / RAG / Búsqueda Vectorial** – genera `answer`.
4. **Respond to Webhook** – envía `{"answer": "..."}` a ElevenLabs.

Modificaciones clave frente a un chat clásico

- Quita nodos **Chat Input** y **Memoria** que esperen historiales de mensaje.
- Simplifica **System Prompt**: define rol, prohíbe alucinar y especifica formato.
- Mantén **Webhook (respuesta)** al final para cerrar la llamada HTTP.

10 · Acelera la respuesta con modelos Google Gemini

Opción	Ventaja	Paso rápido
Gemini 2 Flash 001	10 k+ peticiones gratuitas/mes · latencia baja	Crea API Key en Google AI Studio ▶️ pega en credenciales n8n.
Temp. 0.7 – 0.9	Sonido natural en voz	Ajusta si notas respuestas "robóticas".
Prompt fino	Evita "no tengo info" falsos	Prueba a eliminar reglas demasiado restrictivas.

> Si tu PC tiene GPU, un **modelo Ollama local** puede ser más veloz que Gemini; mide.

11 · Pasa tu flujo a producción

1. **Activa** el workflow ➜ n8n genera URL */webhook* **(sin `/test/`)**.
2. Sustituye la URL en la Custom Tool de ElevenLabs y **Guarda**.
3. Haz una llamada real en *Test Agent* → debe funcionar sin pulsar "Test Workflow".

12 · Incluye el agente en tu web

Paso	Detalle
Agents ▶️ **Widget**	Copia el `<script>` generado.
Pega en HTML	Ej. después de `<body>` o en CMS.
Prueba	Clic en el "botón flotante" → pide micro y habla.

El widget carga remotamente; no consume recursos de tu servidor.

13 · Conecta un número de teléfono (Twilio)

Requisito	Dónde encontrarlo	Nota
Número	Twilio ▶️ Phone Numbers ▶️ Buy Number	UE requiere datos de empresa.
Account SID	Twilio dashboard	Copia sin espacios.
Auth Token	Ídem	Guarda como secreto.
ElevenLabs ▶️ **Phone Numbers**	“Import from Twilio”, pega los datos	Asigna al agente deseado.

☎️ Las llamadas a números de otro país pueden ser caras; adquiere un número local.

Página de Notas del Tema

Esta página está pensada para que puedas anotar ideas clave, dudas y reflexiones importantes sobre el tema anterior.