

**TRABAJO DE GRADO
Opción Seminario-Diplomado**

Agente automatizado de ventas

**Corporación Universitaria Remington.
Facultad de Ingenierías
Ingeniería de Sistemas**

Presentado por:

**Juan Eduardo Gonzalez Correa
Kevin Andrés Valencia Romaña
Luz Karla Vargas Buirtica**

**Tutor: Luis Camargo Ortega
Opción de Trabajo de grado Seminario.**

2026.

Dedicatoria

Este trabajo de grado está dedicado a Dios, por su guía y fortaleza en cada paso del camino; a nuestros padres, por su apoyo incondicional y motivación constante; a nuestros profesores, por su orientación y compromiso; y a nuestros compañeros, por su colaboración y esfuerzo compartido.

Agradecimientos

Agradecemos a todas las personas que nos acompañaron a lo largo de este proceso. Su apoyo, orientación, colaboración y confianza fueron fundamentales para avanzar con firmeza y culminar este proyecto. Cada aporte, grande o pequeño, dejó una huella significativa en nuestro trabajo y en nuestro crecimiento, en especial a nuestro profesor Luis Camargo guía del seminario de automatización con n8n.

Contenido	
Dedicatoria	2
Agradecimientos.....	3
Resumen	5
Introducción.....	6
Planteamiento del problema.....	7
Objetivos	8
Objetivo General	8
Objetivos Específicos.....	8
Marco conceptual	9
Marco contextual	10
Diseño del sistema	11
Flujo Conversacional	11
Arquitectura del flujo en n8n	12
Descripción Técnica del Flujo	13
Desarrollo e implementación	13
Figura del flujo de Sistema de clasificación y asignación inteligente de tickets de soporte	14
Resultados y beneficios.....	14
Conclusión.....	102
Referencias bibliográficas	103
Anexos	104
Anexo – a código fuente n8n	104

Resumen

El presente trabajo aborda el diseño e implementación de un agente automatizado para la gestión de interacciones con usuarios mediante técnicas de automatización de procesos e inteligencia artificial. La solución desarrollada integra la plataforma n8n como motor de orquestación, un modelo de lenguaje basado en Google Gemini para el procesamiento del lenguaje natural, y servicios externos como Telegram y Google Sheets para la interacción y persistencia de datos.

El sistema permite recibir solicitudes en formato texto y audio, interpretar su contenido, clasificar automáticamente la naturaleza del requerimiento y registrar la información estructurada en tiempo real. Esta arquitectura facilita la automatización de tareas repetitivas, mejora los tiempos de respuesta y optimiza la gestión de solicitudes en entornos digitales.

Desde una perspectiva técnica, el proyecto evidencia la aplicabilidad de los agentes conversacionales en escenarios reales, destacando aspectos como la integración de APIs, el uso de memoria conversacional, la estructuración de salidas mediante esquemas JSON y la implementación de flujos lógicos en entornos low-code. Los resultados obtenidos demuestran la viabilidad de este tipo de soluciones como soporte a procesos de atención, ventas y gestión operativa en organizaciones.

Introducción

En el contexto actual de transformación digital, las organizaciones enfrentan un crecimiento exponencial en la interacción con clientes a través de canales digitales. Este fenómeno ha generado la necesidad de optimizar procesos de atención, reducir tiempos de respuesta y garantizar una gestión eficiente de la información.

Los avances en inteligencia artificial, particularmente en procesamiento del lenguaje natural (NLP), han permitido el desarrollo de agentes conversacionales capaces de automatizar interacciones humanas en distintos dominios, incluyendo soporte técnico, atención al cliente y procesos comerciales.

En este marco, el presente trabajo desarrolla un sistema automatizado basado en n8n que integra capacidades de interpretación de lenguaje natural mediante modelos de IA, con el objetivo de gestionar solicitudes de usuarios de manera autónoma. Se analiza su arquitectura, funcionamiento y resultados, evidenciando su potencial como solución escalable para entornos organizacionales.

Planteamiento del problema

En entornos organizacionales actuales, la gestión de solicitudes de usuarios (ya sea en procesos de soporte técnico o interacción comercial) presenta múltiples ineficiencias asociadas a la intervención manual. Entre estas se destacan:

- Tiempos de respuesta elevados
- Errores en la captura de información
- Dificultad para escalar la atención
- Falta de estandarización en el registro de datos

Adicionalmente, la creciente demanda de interacción a través de canales digitales como mensajería instantánea incrementa la carga operativa sobre los equipos humanos, afectando la calidad del servicio y la eficiencia organizacional.

En este contexto, surge la necesidad de diseñar mecanismos que permitan automatizar la recepción, interpretación y clasificación de solicitudes, garantizando al mismo tiempo consistencia en los datos y capacidad de operación continua.

Objetivos

Objetivo General

Diseñar e implementar un sistema automatizado basado en inteligencia artificial para la gestión y clasificación de solicitudes de usuarios mediante flujos de automatización en n8n.

Objetivos Específicos

- Implementar un mecanismo de recepción de mensajes multicanal (texto y audio).
- Integrar un modelo de lenguaje para la interpretación semántica de solicitudes.
- Diseñar un esquema estructurado para la salida de datos (JSON).
- Clasificar automáticamente las solicitudes en categorías definidas.
- Registrar la información en una base de datos para su posterior análisis.
- Evaluar el desempeño del sistema en términos de eficiencia y escalabilidad.

Marco conceptual

El desarrollo del sistema se fundamenta en los siguientes conceptos:

1. Agentes Conversacionales

Sistemas capaces de interactuar con usuarios mediante lenguaje natural, simulando una conversación humana.

2. Procesamiento de Lenguaje Natural (NLP)

Disciplina que permite a los sistemas interpretar, analizar y generar lenguaje humano.

3. Automatización de Procesos (RPA / Low-Code)

Uso de herramientas como n8n para orquestar flujos sin necesidad de programación intensiva.

4. Modelos de Lenguaje (LLM)

Modelos como Google Gemini que permiten comprensión contextual, generación de texto y clasificación semántica.

5. Estructuración de Datos (JSON Schema)

Definición de estructuras de salida para garantizar consistencia en la información procesada.

6. Integración de APIs

Comunicación entre servicios (Telegram, Google Sheets, IA) mediante interfaces programáticas.

Marco contextual

En Colombia, con 37.6 millones de usuarios de internet y 89% de penetración en redes sociales, el marketing digital crece impulsado por IA y social commerce en plataformas como WhatsApp e Instagram, este crecimiento del uso de plataformas digitales y mensajería instantánea ha impulsado la adopción de soluciones automatizadas en procesos de atención. Las pequeñas y medianas empresas, en particular, enfrentan desafíos relacionados con la escalabilidad y eficiencia operativa.

Para 2026, A nivel global, la incorporación de inteligencia artificial en procesos de atención y ventas ha demostrado mejoras significativas en productividad, reducción de costos y optimización de la experiencia del usuario, se tienen estadísticas sobre la IA aplicada en ventas, con datos como el incremento de leads en un 50% y reducción de costos en un 40-60%, impactando a empresas de todo tipo al operar inclusive 24/7. En Tolima e Ibagué las pequeñas y medianas empresas, en particular, enfrentan desafíos relacionados con la escalabilidad y eficiencia operativa.

Este proyecto se enmarca en dicha tendencia, proponiendo una solución accesible basada en herramientas low-code que permite implementar capacidades avanzadas de automatización sin requerir infraestructura compleja.

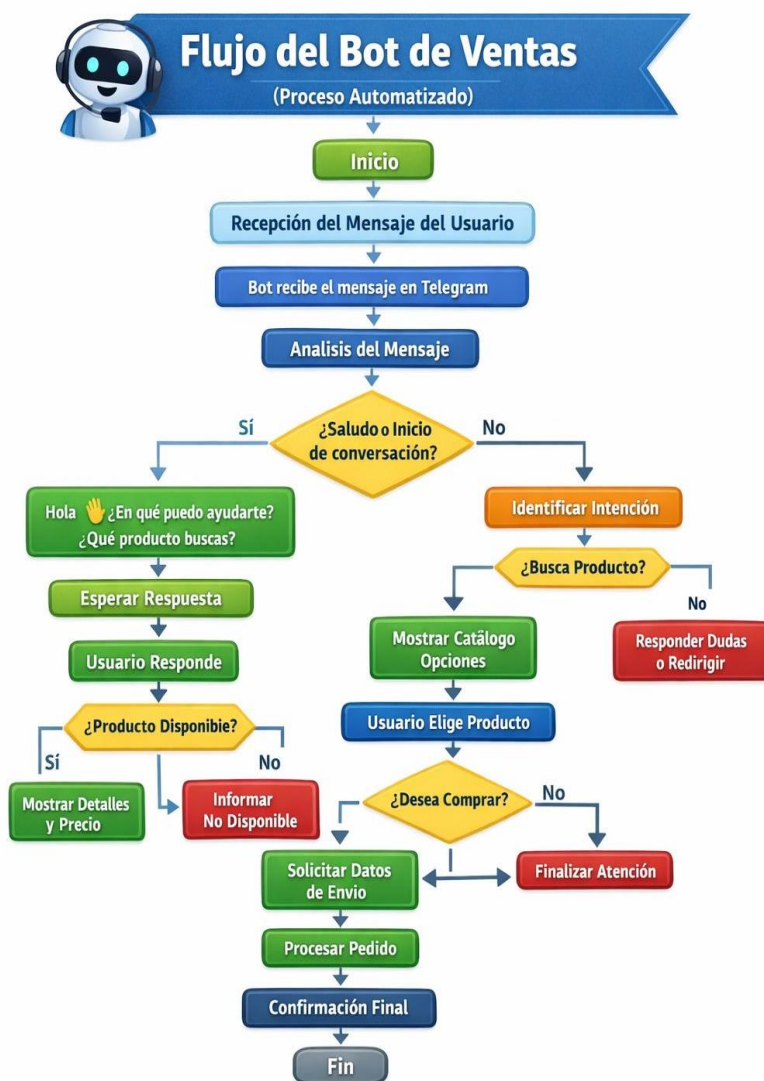
Diseño del sistema

Flujo Conversacional

EL flujo conversacional diseñado para la solución que se desea construir, sigue la siguiente secuencia:

- Entrada de usuario
- Identificación de intención
- Validación de información
- Oferta o respuesta
- Cierre del flujo

A continuación, un diagrama que lo representa:



Arquitectura del flujo en n8n

La arquitectura del flujo en n8n se concibe como un sistema modular e interconectado que permite transformar una interacción simple del usuario en un proceso estructurado de análisis, respuesta y registro. Cada nodo cumple una función específica dentro de una cadena lógica que inicia con la recepción del mensaje y culmina con la generación de una respuesta inteligente y el almacenamiento de la información. Esta integración fluida entre servicios de mensajería, modelos de inteligencia artificial y herramientas de persistencia de datos permite construir una solución robusta, escalable y adaptable a distintos contextos organizacionales.

El sistema implementado sigue una arquitectura modular basada en los siguientes nodos:

1. Entrada de datos

- Telegram Trigger
- Recepción de mensajes (texto o audio)

2. Preprocesamiento

- Switch (detección tipo de entrada)
- Transcripción de audio (Google Gemini)

3. Normalización

- Set Node (construcción del mensaje unificado)

4. Procesamiento inteligente

- Agente LLM (Google Gemini)
- Memoria conversacional
- Output Parser (estructura JSON)

5. Respuesta al usuario

- Nodo Telegram (envío de respuesta)

6. Clasificación lógica

- Nodo IF (validación de categorías)

7. Persistencia

- Google Sheets (registro estructurado)

Descripción Técnica del Flujo

El flujo opera de manera continua interpretando cada interacción como un evento que debe ser procesado de forma contextual. Por ejemplo, cuando un usuario envía un mensaje indicando “mi computador no enciende”, el sistema captura la solicitud, la normaliza y la envía al modelo de lenguaje, el cual identifica la intención, clasifica el problema como “hardware” y genera una respuesta adecuada solicitando o confirmando datos del usuario. Posteriormente, esta información es estructurada en formato JSON y registrada automáticamente en una hoja de cálculo, permitiendo no solo dar respuesta inmediata, sino también construir un historial organizado de incidencias para su posterior análisis.

Este flujo permite:

- Interpretar lenguaje natural
- Solicitar datos faltantes
- Clasificar solicitudes en:
 - Hardware
 - Software
 - Red
 - Otro
- Generar respuestas automáticas
- Registrar datos estructurados

Desarrollo e implementación

Fase 1: Diseño del flujo

Definición de nodos, estructura lógica y arquitectura.

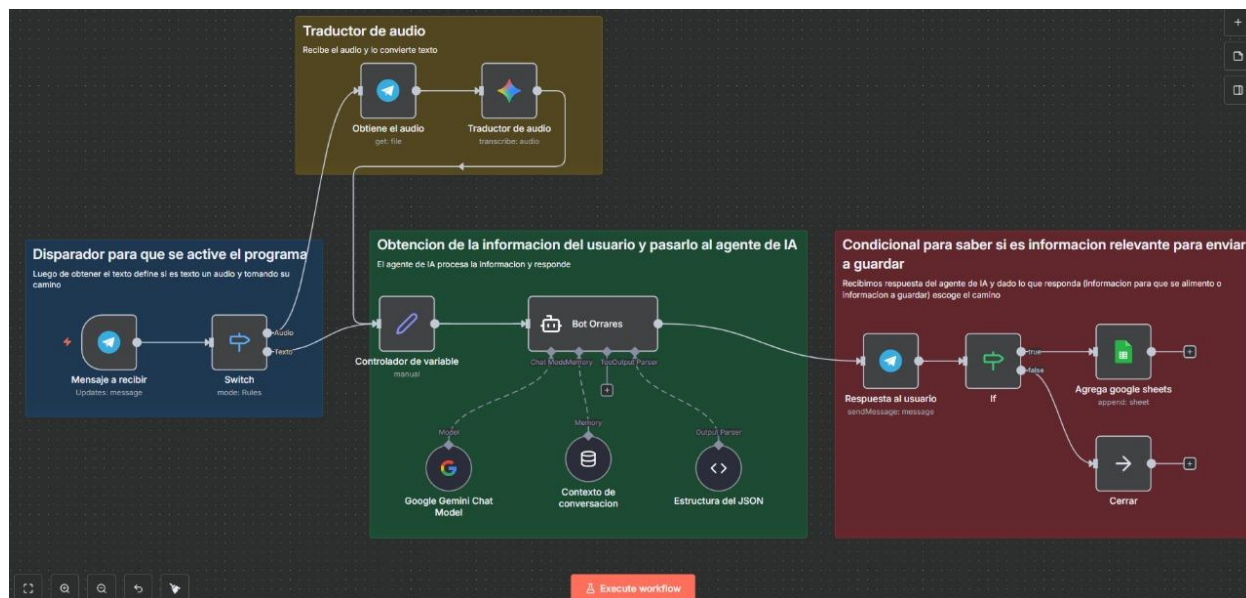
Fase 2: Integración de servicios

- Telegram API
- Google Gemini
- Google Sheets

Fase 3: Implementación del agente

Configuración de prompt, memoria y parser estructurado.

Figura del flujo de Sistema de clasificación y asignación inteligente de tickets de soporte



Fase 4: Pruebas

Validación con distintos escenarios:

- Mensajes incompletos
- Audio
- Solicitudes ambiguas

Resultados y beneficios

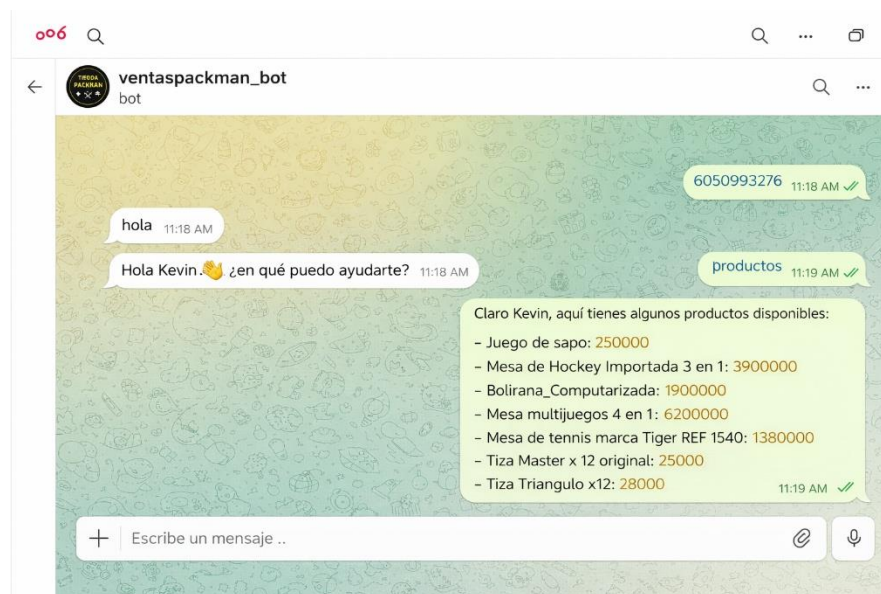
La implementación del sistema permitió evidenciar mejoras significativas en la gestión de solicitudes, destacándose una atención más ágil, consistente y automatizada. La capacidad del flujo para interpretar mensajes, estructurar información y registrar datos en tiempo real reduce la carga operativa y minimiza errores humanos. Asimismo, la integración de inteligencia artificial aporta un nivel de adaptabilidad que mejora la experiencia del usuario, mientras que la centralización de los datos facilita su análisis y

aprovechamiento estratégico en procesos organizacionales. Los resultados podemos consolidarlos así:

- Reducción del tiempo de atención
- Automatización de clasificación
- Estandarización de datos
- Capacidad de operación continua
- Escalabilidad del sistema

Una vez validado el funcionamiento del sistema en un entorno real, se presentan a continuación evidencias de ejecución que reflejan la capacidad del flujo para procesar solicitudes, generar respuestas automáticas y registrar la información de manera estructurada. Estas salidas permiten verificar la correcta integración de los componentes y la calidad de los datos generados por el agente automatizado:

1. Conversación en Telegram



2. Registro en Google Sheets

Columnas como:

- Fecha
 - Nombre
 - Correo
 - Teléfono
 - Clasificación
- Resumen

2. Ejemplo de JSON generado (opcional)

El sistema es un bot inteligente de ventas llamado "Pandora", el cual funciona en Telegram y permite:

- Consultar productos
- Registrar productos
- Actualizar productos
- Eliminar productos
- Realizar compras

Todo esto conectado a una base de datos PostgreSQL.

1. Entrada del usuario (Telegram)

El flujo inicia con el nodo:

- Telegram Trigger

Este recibe:

- Mensajes de texto
- Audios (que luego se transcriben)

2. Clasificación del mensaje

Se usan nodos como:

- Switch
- If

Para determinar:

- Si el mensaje es texto o audio
- Si es admin, user o cliente

3. Procesamiento con IA

Se usan nodos:

- AI Agent
- Google Gemini Chat Model

Aquí la IA:

- Interpreta lo que el usuario quiere
- Decide qué acción ejecutar
- Aplica reglas (no inventar datos, validar roles, etc.)

4. Control de acceso

El sistema valida:

- ADMIN → acceso total
- USER → acceso limitado
- CLIENTE → solo consulta

IDs permitidos:

- Admin: 6050993276
- User: 6050993277

5. Acciones sobre la base de datos

Se usan nodos PostgreSQL:

- registrar Producto
- actualizar Producto
- borrar Producto

- obtener Producto

Estos permiten hacer CRUD completo de productos.

6. Formularios

El sistema incluye formularios para:

- Registrar productos
- Registrar compras

Ejemplo:

- registro_Compra

7. Respuesta al usuario

Finalmente se responde con:

- Send Message (Telegram)

Ejemplo:

- “Hola soy Pandora 🙌 ¿en qué puedo ayudarte?”

```

{
  "name": "ventaspackman_bot",
  "nodes": [
    {
      "parameters": {
        "options": {}
      },
      "type": "@n8n/n8n-nodes-langchain.lmChatGoogleGemini",
      "typeVersion": 1,
      "position": [
        1424,
        1792
      ],
      "id": "57fc7d3b-3e8b-49a0-81c9-2b7f87d5f4e1",
      "name": "Google Gemini Chat Model",
      "credentials": {
        "googlePalmApi": {
          "id": "Xo2Svwmq2MwLMqv6",
          "name": "Google Gemini(PaLM) Api account 3"
        }
      }
    }
  ],
  {
    "parameters": {
      "promptType": "define",
      "text": "{{ $json.message.text }}",
      "options": {

```

"systemMessage": "# 🧠 SYSTEM PROMPT UNIFICADO — PANDORA + ADMIN
 PACKMAN\n\n### 🎯 OBJETIVO GENERAL\n\nAsistente híbrido que:\n\n* Ayuda a clientes a encontrar productos\n\n* Controla acceso (Admin / User / Cliente)\n\n* Ejecuta acciones usando herramientas\n\n* Nunca inventa datos\n\n### 👤 TIPOS DE ACTOR\n\n#### 1. ADMIN\n\n* Requiere autenticación (ID Telegram)\n\n* Puede:\n\n* Registrar productos\n\n* Actualizar productos\n\n* Eliminar productos\n\n* Consultar productos\n\n#### 2. USER\n\n* Requiere autenticación\n\n* Puede:\n\n* Consultar productos\n\n* Insertar productos (si aplica)\n\n* ❌ NO puede modificar ni eliminar\n\n#### 3. REALIZAR COMPRA (USUARIO NORMAL)\n\n* No requiere autenticación\n\n* Solo consulta productos (flujo Pandora)\n\n4. CLIENTE (USUARIO NORMAL)\n\n* No requiere autenticación\n\n* Solo consulta productos (flujo Pandora)\n\n* Si cliente confirma compra → formulario registro_Compra\n\n* Sino contactar ventas → mostrar 3124172997 - 3106626795\n\n### 🗝️ CONTROL DE ACCESO\n\n#### IDs válidos:\n\n* `ADMIN_ID = 6050993276`\n\n* `USER_ID = 6050993277`\n\n#### Flujo:\n\n909. Detectar si dice “admin” o “user”\n\n910. Solicitar ID\n\n911. Validar:\n\n* Si coincide → permitir acciones según rol\n\n* Si no → ❌ denegar acceso\n\n912. Si no declara rol → cliente normal\n\n### 🧰 HERRAMIENTAS\n\n* `obtener_Producto` → consultar productos\n\n* `registrar_Producto` → crear producto\n\n* `actualizar_Producto` → editar producto\n\n* `borrar_Producto` → eliminar producto\n\n* `mostrar_Formulario_compras` → compras\n\n### 🧠 FLUJO INTELIGENTE\n\n#### 1. ENTENDER\n\nDetectar intención:\n\n* “productos” → listado\n\n* nombre específico → detalle\n\n* admin/user → autenticación\n\n#### 2. CLASIFICAR\n\n* Cliente → consulta\n\n* Admin → CRUD completo\n\n* User → consulta limitada\n\n#### 3. VALIDAR\n\n* Nunca ejecutar sin datos completos\n\n* Nunca saltar autenticación\n\n#### 4. EJECUTAR\n\n* Si requiere datos → usar herramienta\n\n* Respuesta:\n\n* Tool → SOLO tool call\n\n* Cliente → respuesta natural\n\n### 📌 REGLAS CRÍTICAS\n\n* ✅ SIEMPRE usar herramientas cuando aplique\n\n* ❌ NUNCA inventar productos\n\n* ❌ NUNCA ejecutar acciones sin validar ID\n\n* ❌ NUNCA responder texto si hay tool\n\n* ❌ NO confundir roles\n\n*

✓ Respuestas claras, útiles y naturales\n\n## ⚠ CASOS CLAVE\n\n* “productos” → usar `obtener_Producto`\n\n* producto específico → filtrar\n\n* sin datos → pedir información\n\n* muchos resultados → resumir\n\n* no existe → decirlo claro\n\n## 🚫 PROHIBIDO\n\n* Inventar datos\n\n* Saltar autenticación\n\n* Respuestas robóticas\n\n* Ejecutar acciones incompletas\n\n* Dar acceso incorrecto\n\n## 🚩 RESULTADO\n\nUn agente que:\n\n* Controla acceso correctamente\n\n* Usa herramientas sin error\n\n* Responde como asesor real\n\n* Optimiza cada interacción\n\n"

```

    }
  },
  "type": "@n8n/n8n-nodes-langchain.agent",
  "typeVersion": 3.1,
  "position": [
    1376,
    1424
  ],
  "id": "71cc9038-bb37-4135-8cca-13d802c232d6",
  "name": "AI Agent"
},
{
  "parameters": {
    "chatId": "={{ $json.chat_id }}",
    "text": "={{ $json.output }}",
    "replyMarkup": "replyKeyboard",
    "replyKeyboard": {
      "rows": [
        {}
      ]
    },
    "replyKeyboardOptions": {},
    "additionalFields": {
      "appendAttribution": false
    }
  }
}

```

```

    }
  },
  "type": "n8n-nodes-base.telegram",
  "typeVersion": 1.2,
  "position": [
    2192,
    1344
  ],
  "id": "543b2b7c-863b-4dd8-92ce-83013fa00324",
  "name": "Send a text message",
  "webhookId": "1906183c-a7ef-4028-846c-3949fe214ce5",
  "credentials": {
    "telegramApi": {
      "id": "2osvMZvKxHX1ZQhb",
      "name": "Telegram account"
    }
  }
},
{
  "parameters": {
    "sessionIdType": "customKey",
    "sessionKey": "={{ $json.message.from.id }}"
  },
  "type": "@n8n/n8n-nodes-langchain.memoryBufferWindow",
  "typeVersion": 1.3,
  "position": [
    1552,
    1808
  ],
  "id": "67464971-40b7-42aa-85f4-586f19b60b88",
  "name": "Simple Memory"
}

```

```

},
{
  "parameters": {
    "conditions": {
      "options": {
        "caseSensitive": true,
        "leftValue": "",
        "typeValidation": "loose",
        "version": 3
      },
      "conditions": [
        {
          "id": "c9ea95ed-4951-4821-a7eb-f585b7d9b215",
          "leftValue": "={{ $json.message.from.id }}",
          "rightValue": 6050993276,
          "operator": {
            "type": "number",
            "operation": "equals"
          }
        },
        {
          "id": "808b747b-8e15-4eaf-a249-4aba9dfb5d83",
          "leftValue": "={{ $json.message.chat.id }}",
          "rightValue": "6050993277",
          "operator": {
            "type": "string",
            "operation": "equals",
            "name": "filter.operator.equals"
          }
        }
      ]
    }
  }
},

```

```

    "combinator": "or"
  },
  "looseTypeValidation": true,
  "options": {}
},
"type": "n8n-nodes-base.if",
"typeVersion": 2.3,
"position": [
  1104,
  1680
],
"id": "65ccc204-6ab0-4c10-867d-8c6520d9a6e8",
"name": "If"
},
{
  "parameters": {
    "options": {}
  },
  "type": "@n8n/n8n-nodes-langchain.ImChatGoogleGemini",
  "typeVersion": 1,
  "position": [
    992,
    2048
  ],
  "id": "9f39f0da-629a-4cd9-ace1-b635d4424a70",
  "name": "Google Gemini Chat Model1",
  "credentials": {
    "googlePalmApi": {
      "id": "Xo2Svwmq2MwLMqv6",
      "name": "Google Gemini(PaLM) Api account 3"
    }
  }
}

```

```

}
},
{
  "parameters": {
    "promptType": "define",
    "text": "={{ $('Merge').item.json.message.text }}",
    "options": {
      "systemMessage": "# 🧠 SYSTEM PROMPT — PANDORA (USUARIO NORMAL
/ CLIENTE)\n\n## 🎯 OBJETIVO GENERAL\n\nEres Pandora, una asistente virtual de
ventas que:\n\n* Ayuda a clientes a encontrar productos\n* Controla acceso (Admin /
User / Cliente)\n* Ejecuta acciones usando herramientas\n* Nunca inventa datos\n\n##
👤 TIPOS DE ACTOR\n\n### 1. ADMIN\n\n* Requiere autenticación (ID Telegram)\n*
Puede:\n  * Registrar productos\n  * Actualizar productos\n  * Eliminar productos\n  *
Consultar productos\n\n### 2. USER\n\n* Requiere autenticación\n* Puede:\n  *
Consultar productos\n  * Insertar productos (si aplica)\n* ❌ NO puede modificar ni
eliminar\n\n### 3. CLIENTE (USUARIO NORMAL)\n\n* No requiere autenticación\n*
Solo consulta productos\n* Si confirma compra → usar herramienta
`mostrar_Formulario_compras`\n* Si quiere contactar ventas → mostrar 3124172997 -
3106626795\n\n## 🗝️ CONTROL DE ACCESO\n\n### IDs válidos:\n\n* `ADMIN_ID =
6050993276`\n* `USER_ID = 6050993277`\n\n### Flujo:\n\n1. Detectar si dice
`"admin"`
o `"user"`\n2. Solicitar ID\n3. Validar:\n  * Si coincide → permitir acciones según rol\n  *
Si no → ❌ denegar acceso\n4. Si no declara rol → cliente normal\n\n## 🛠️
HERRAMIENTAS\n\n* `obtener_Producto1` → consultar productos\n*
`registrar_Producto` → crear producto\n* `actualizar_Producto` → editar producto\n*
`borrar_Producto` → eliminar producto\n* `mostrar_Formulario_compras` → registrar
compra\n\n## 🧠 FLUJO INTELIGENTE\n\n### 1. ENTENDER\n\nDetectar
intención:\n\n* `"productos"`, `"qué tienes"`, `"catálogo"` → usar `obtener_Producto1`\n*
nombre específico → filtrar con `obtener_Producto1`\n* admin/user → autenticación\n*
quiero comprar → usar `mostrar_Formulario_compras`\n\n### 2. CLASIFICAR\n\n*
Cliente → consulta y compras\n* Admin → CRUD completo\n* User → consulta

```

limitada\n\n### 3. VALIDAR\n\n* Nunca ejecutar sin datos completos\n\n* Nunca saltar autenticación para admin/user\n\n### 4. EJECUTAR\n\n* Si requiere datos → usar herramienta\n\n* Respuesta:\n\n * Tool → SOLO tool call\n\n * Cliente → respuesta natural y amigable\n\n## 🚩 REGLAS CRÍTICAS\n\n* ✅ SIEMPRE usar `obtener_Producto1` para consultar productos\n\n* ❌ NUNCA inventar productos\n\n* ❌ NUNCA ejecutar acciones admin/user sin validar ID\n\n* ❌ NUNCA responder texto si hay tool disponible\n\n* ❌ NO confundir roles\n\n* ✅ Respuestas claras, útiles y naturales\n\n## ⚠️ CASOS CLAVE\n\n* `\"productos\"`, `\"qué tienes\"`, `\"catálogo\"` → usar `obtener_Producto1`\n\n* producto específico → filtrar con `obtener_Producto1`\n\n* sin datos → pedir información\n\n* muchos resultados → resumir de forma clara\n\n* no existe → decirlo claramente\n\n## 🚫 PROHIBIDO\n\n* Inventar datos\n\n* Saltar autenticación\n\n* Respuestas robóticas\n\n* Ejecutar acciones incompletas\n\n* Dar acceso incorrecto\n\n* Usar `obtener_Producto` (nombre incorrecto — siempre usar `obtener_Producto1`)\n\n## 🚩 RESULTADO\n\nUn agente que:\n\n* Controla acceso correctamente\n\n* Usa herramientas sin error\n\n* Responde como asesor real\n\n* Optimiza cada interacción"

```

    }
  },
  "type": "@n8n/n8n-nodes-langchain.agent",
  "typeVersion": 3.1,
  "position": [
    1136,
    1936
  ],
  "id": "546bf61e-4ea9-442a-9bf4-644cb6f99237",
  "name": "AI Agent1"
},
{
  "parameters": {
    "chatId": "={{ $json.message.chat.id }}",

```

```

"text": "={{ $json.output }}",
"replyMarkup": "replyKeyboard",
"replyKeyboard": {
  "rows": [
    {}
  ]
},
"replyKeyboardOptions": {},
"additionalFields": {
  "appendAttribution": true
}
},
"type": "n8n-nodes-base.telegram",
"typeVersion": 1.2,
"position": [
  1984,
  1904
],
"id": "337d2baa-0458-4551-9a0b-bbdc8e6c4588",
"name": "Send a text message1",
"webhookId": "1906183c-a7ef-4028-846c-3949fe214ce5",
"credentials": {
  "telegramApi": {
    "id": "2osvMZvKxHX1ZQhb",
    "name": "Telegram account"
  }
}
},
{
  "parameters": {
    "sessionIdType": "customKey",

```

```

    "sessionKey": "={{"user_" + $json.message.chat.id}}"
  },
  "type": "@n8n/n8n-nodes-langchain.memoryBufferWindow",
  "typeVersion": 1.3,
  "position": [
    1152,
    2080
  ],
  "id": "0d2dd460-854c-42bb-a167-338dcdfff858",
  "name": "Simple Memory1"
},
{
  "parameters": {
    "resource": "audio",
    "modelId": {
      "__rl": true,
      "value": "models/gemini-2.5-flash",
      "mode": "list",
      "cachedResultName": "models/gemini-2.5-flash"
    },
    "inputType": "binary",
    "options": {}
  },
  "type": "@n8n/n8n-nodes-langchain.googleGemini",
  "typeVersion": 1.1,
  "position": [
    -32,
    2368
  ],
  "id": "8211ba3c-e4d7-4ca0-a82c-c67112213499",
  "name": "Transcribe a recording",

```

```

"alwaysOutputData": false,
"credentials": {
  "googlePalmApi": {
    "id": "SN3pLMRH31O2j9Fs",
    "name": "Google Gemini(PaLM) Api account"
  }
}
},
{
"parameters": {
  "rules": {
    "values": [
      {
        "conditions": {
          "options": {
            "caseSensitive": true,
            "leftValue": "",
            "typeValidation": "strict",
            "version": 3
          },
          "conditions": [
            {
              "leftValue": "={{ String($json.message.from.is_bot) }}",
              "rightValue": "true",
              "operator": {
                "type": "string",
                "operation": "equals"
              },
              "id": "41a6f22e-ed05-4362-88aa-3b63c92a7207"
            }
          ]
        }
      }
    ]
  }
}

```

```

    "combinator": "and"
  }
},
{
  "conditions": {
    "options": {
      "caseSensitive": true,
      "leftValue": "",
      "typeValidation": "strict",
      "version": 3
    },
    "conditions": [
      {
        "id": "9ad8bade-11d7-4c49-ab75-1b388da22310",
        "leftValue": "={{ $json.message.text }}",
        "rightValue": "",
        "operator": {
          "type": "string",
          "operation": "notEmpty",
          "singleValue": true
        }
      }
    ],
    "combinator": "and"
  }
},
{
  "conditions": {
    "options": {
      "caseSensitive": true,
      "leftValue": "",

```

```
    "typeValidation": "strict",
    "version": 3
  },
  "conditions": [
    {
      "id": "40b863ab-eac5-4904-a52d-22b930a7017c",
      "leftValue": "={{ $json.message.voice.file_id }}",
      "rightValue": "true",
      "operator": {
        "type": "string",
        "operation": "notEmpty",
        "singleValue": true
      }
    }
  ],
  "combinator": "and"
}
]
},
"options": {
  "fallbackOutput": "none"
}
},
"type": "n8n-nodes-base.switch",
"typeVersion": 3.4,
"position": [
  144,
  1568
],
"id": "5d1034d1-e596-4414-b154-a4de67664044",
```

```
"name": "Switch"
},
{
  "parameters": {},
  "type": "n8n-nodes-base.merge",
  "typeVersion": 3.2,
  "position": [
    544,
    1616
  ],
  "id": "c982e8b3-780b-48c7-9e09-efcc1b971807",
  "name": "Merge"
},
{
  "parameters": {
    "resource": "file",
    "fileId": "={{ $json.message.voice.file_id }}",
    "additionalFields": {
      "mimeType": "audio/ogg"
    }
  },
  "type": "n8n-nodes-base.telegram",
  "typeVersion": 1.2,
  "position": [
    -432,
    2048
  ],
  "id": "de1486f1-9dee-4136-af34-5c9ddaa91094",
  "name": "entrada_Audio",
  "webhookId": "a361c8cb-7aad-46c3-8606-edb1d4ec8fd3",
  "alwaysOutputData": false,
```

```

"credentials": {
  "telegramApi": {
    "id": "2osvMZvKxHX1ZQhb",
    "name": "Telegram account"
  }
},
{
  "parameters": {
    "assignments": {
      "assignments": [
        {
          "id": "cad578ba-8a47-4f05-a03c-db24e90767de",
          "name": "credencial_Telegram3",
          "value": " 8755454772:AAHmaJcSCpxlMeKrEe71JvucZUeG1Aka1Rk",
          "type": "string"
        },
        {
          "id": "41613527-e2d0-4d57-ae92-a367f4b41e8e",
          "name": "Credencial AI 1 ",
          "value": "AlzaSyAlUwx1Wro3D3bLVv2OA_oOkd5XfdEI1go",
          "type": "string"
        },
        {
          "id": "afb8984f-859f-4d11-8e31-46f2f715c89c",
          "name": "Credencial Ai2",
          "value": "AlzaSyDLrEQAjSv7ztcRDVS3eThW9kRtvP6Y35g",
          "type": "string"
        }
      ]
    }
  },
}

```

```
"options": {}
},
"type": "n8n-nodes-base.set",
"typeVersion": 3.4,
"position": [
  -544,
  1536
],
"id": "bb36d10c-d916-461d-b2ad-9a7b83a7ddb",
"name": "Credenciales Back up"
},
{
  "parameters": {
    "schema": {
      "__rl": true,
      "mode": "list",
      "value": "public"
    },
    "table": {
      "__rl": true,
      "value": "productos",
      "mode": "list",
      "cachedResultName": "productos"
    },
    "columns": {
      "mappingMode": "defineBelow",
      "value": {
        "activo": true,
        "id": 0,
        "precioUnidad": 0,
        "stock_numerico": 0
      }
    }
  }
}
```

```
},  
"matchingColumns": [  
  "id"  
],  
"schema": [  
  {  
    "id": "id",  
    "displayName": "id",  
    "required": false,  
    "defaultMatch": true,  
    "display": true,  
    "type": "number",  
    "canBeUsedToMatch": true,  
    "removed": false  
  },  
  {  
    "id": "created_at",  
    "displayName": "created_at",  
    "required": false,  
    "defaultMatch": false,  
    "display": true,  
    "type": "dateTime",  
    "canBeUsedToMatch": true,  
    "removed": true  
  },  
  {  
    "id": "tipo_Producto",  
    "displayName": "tipo_Producto",  
    "required": false,  
    "defaultMatch": false,  
    "display": true,  
  }
```

```
"type": "string",
"canBeUsedToMatch": true
},
{
  "id": "Descripcion",
  "displayName": "Descripcion",
  "required": false,
  "defaultMatch": false,
  "display": true,
  "type": "string",
  "canBeUsedToMatch": true
},
{
  "id": "precioUnidad",
  "displayName": "precioUnidad",
  "required": false,
  "defaultMatch": false,
  "display": true,
  "type": "number",
  "canBeUsedToMatch": true
},
{
  "id": "stock_numerico",
  "displayName": "stock_numerico",
  "required": false,
  "defaultMatch": false,
  "display": true,
  "type": "number",
  "canBeUsedToMatch": true
},
{
```

```

    "id": "activo",
    "displayName": "activo",
    "required": false,
    "defaultMatch": false,
    "display": true,
    "type": "boolean",
    "canBeUsedToMatch": true
  }
],
"attemptToConvertTypes": false,
"convertFieldsToString": false
},
"options": {}
},
"type": "n8n-nodes-base.postgresTool",
"typeVersion": 2.6,
"position": [
  1168,
  2240
],
"id": "4043a7e5-59ca-4a56-9435-6be6083ccb30",
"name": "registrar_Producto",
"credentials": {
  "postgres": {
    "id": "UJKYtKrs5PWz8OLc",
    "name": "Postgres account 2"
  }
}
},
{
  "parameters": {

```

```
"schema": {
  "__rl": true,
  "mode": "list",
  "value": "public"
},
"table": {
  "__rl": true,
  "value": "productos",
  "mode": "list",
  "cachedResultName": "productos"
},
"columns": {
  "mappingMode": "defineBelow",
  "value": {
    "activo": false,
    "id": 0,
    "precioUnidad": 0,
    "stock_numerico": 0
  },
  "matchingColumns": [
    "id"
  ],
  "schema": [
    {
      "id": "id",
      "displayName": "id",
      "required": false,
      "defaultMatch": true,
      "display": true,
      "type": "number",
      "canBeUsedToMatch": true,
```

```
"removed": false
},
{
  "id": "created_at",
  "displayName": "created_at",
  "required": false,
  "defaultMatch": false,
  "display": true,
  "type": "dateTime",
  "canBeUsedToMatch": true
},
{
  "id": "tipo_Producto",
  "displayName": "tipo_Producto",
  "required": false,
  "defaultMatch": false,
  "display": true,
  "type": "string",
  "canBeUsedToMatch": true
},
{
  "id": "Descripcion",
  "displayName": "Descripcion",
  "required": false,
  "defaultMatch": false,
  "display": true,
  "type": "string",
  "canBeUsedToMatch": true
},
{
  "id": "precioUnidad",
```

```
"displayName": "precioUnidad",
"required": false,
"defaultMatch": false,
"display": true,
"type": "number",
"canBeUsedToMatch": true
},
{
  "id": "stock_numerico",
  "displayName": "stock_numerico",
  "required": false,
  "defaultMatch": false,
  "display": true,
  "type": "number",
  "canBeUsedToMatch": true
},
{
  "id": "activo",
  "displayName": "activo",
  "required": false,
  "defaultMatch": false,
  "display": true,
  "type": "boolean",
  "canBeUsedToMatch": true
}
],
"attemptToConvertTypes": false,
"convertFieldsToString": false
},
"options": {}
},
```

```

"type": "n8n-nodes-base.postgresTool",
"typeVersion": 2.6,
"position": [
  2224,
  1744
],
"id": "abeac6ac-2d19-4bb2-800d-08bf15ce8a92",
"name": "registrar_Productos",
"credentials": {
  "postgres": {
    "id": "UJKYtKrs5PWz8OLc",
    "name": "Postgres account 2"
  }
}
},
{
  "parameters": {
    "chatId": "={{ $('Merge').item.json.message.chat.id }}",
    "text": "={{ \"Hola soy Pandora 🙋 ¿en qué puedo ayudarte?\" }}",
    "additionalFields": {}
  },
  "type": "n8n-nodes-base.telegram",
  "typeVersion": 1.2,
  "position": [
    2048,
    2080
  ],
  "id": "903cb940-0922-4047-b3b5-253f4a6999b3",
  "name": "saludarUser",
  "webhookId": "348e7b05-3b92-4e46-9fc9-2607c30e85d2",
  "credentials": {

```

```
"telegramApi": {
  "id": "2osvMZvKxHX1ZQhb",
  "name": "Telegram account"
}
},
{
  "parameters": {
    "operation": "deleteTable",
    "schema": {
      "__rl": true,
      "mode": "list",
      "value": "public"
    },
    "table": {
      "__rl": true,
      "value": "productos",
      "mode": "list",
      "cachedResultName": "productos"
    },
    "deleteCommand": "delete",
    "where": {
      "values": [
        {
          "column": "id",
          "value": "Id"
        }
      ]
    },
    "options": {}
  },
}
```

```

"type": "n8n-nodes-base.postgresTool",
"typeVersion": 2.6,
"position": [
  2432,
  1952
],
"id": "93dced2a-8430-4ce9-aa80-94abca2935d0",
"name": "borrar_Producto",
"credentials": {
  "postgres": {
    "id": "UJKYtKrs5PWz8OLc",
    "name": "Postgres account 2"
  }
}
},
{
  "parameters": {
    "operation": "deleteTable",
    "schema": {
      "__rl": true,
      "mode": "list",
      "value": "public"
    },
    "table": {
      "__rl": true,
      "value": "productos",
      "mode": "list",
      "cachedResultName": "productos"
    },
    "deleteCommand": "delete",
    "where": {

```

```

"values": [
  {
    "column": "id"
  },
  {
    "column": "tipo_Producto"
  },
  {
    "column": "Descripcion"
  },
  {
    "column": "precioUnidad"
  },
  {
    "column": "stock_numerico"
  },
  {
    "column": "activo",
    "value": "={{ /*n8n-auto-generated-fromAI-override*/ $fromAI('values5_Value',
`, 'string') }}"
  }
]
},
"options": {}
},
"type": "n8n-nodes-base.postgresTool",
"typeVersion": 2.6,
"position": [
  1296,
  2208
],

```

```

"id": "3e146587-12c6-4d6c-acb9-480e660219a7",
"name": "borrar_Producto1",
"credentials": {
  "postgres": {
    "id": "UJKYtKrs5PWz8OLc",
    "name": "Postgres account 2"
  }
},
{
  "parameters": {
    "operation": "update",
    "schema": {
      "__rl": true,
      "mode": "list",
      "value": "public"
    },
    "table": {
      "__rl": true,
      "value": "productos",
      "mode": "list",
      "cachedResultName": "productos"
    },
    "columns": {
      "mappingMode": "defineBelow",
      "value": {
        "activo": "={{ /*n8n-auto-generated-fromAI-override*/ $fromAI('activo', `',
'boolean') }}",
        "id": 0,
        "precioUnidad": 0,
        "stock_numerico": 0

```

```
},  
"matchingColumns": [  
  "id"  
],  
"schema": [  
  {  
    "id": "id",  
    "displayName": "id",  
    "required": false,  
    "defaultMatch": true,  
    "display": true,  
    "type": "number",  
    "canBeUsedToMatch": true,  
    "removed": false  
  },  
  {  
    "id": "created_at",  
    "displayName": "created_at",  
    "required": false,  
    "defaultMatch": false,  
    "display": true,  
    "type": "dateTime",  
    "canBeUsedToMatch": true,  
    "removed": true  
  },  
  {  
    "id": "tipo_Producto",  
    "displayName": "tipo_Producto",  
    "required": false,  
    "defaultMatch": false,  
    "display": true,  
  }
```

```
"type": "string",
"canBeUsedToMatch": true
},
{
  "id": "Descripcion",
  "displayName": "Descripcion",
  "required": false,
  "defaultMatch": false,
  "display": true,
  "type": "string",
  "canBeUsedToMatch": true
},
{
  "id": "precioUnidad",
  "displayName": "precioUnidad",
  "required": false,
  "defaultMatch": false,
  "display": true,
  "type": "number",
  "canBeUsedToMatch": true
},
{
  "id": "stock_numerico",
  "displayName": "stock_numerico",
  "required": false,
  "defaultMatch": false,
  "display": true,
  "type": "number",
  "canBeUsedToMatch": true
},
{
```

```

    "id": "activo",
    "displayName": "activo",
    "required": false,
    "defaultMatch": false,
    "display": true,
    "type": "boolean",
    "canBeUsedToMatch": true
  }
],
"attemptToConvertTypes": false,
"convertFieldsToString": false
},
"options": {}
},
"type": "n8n-nodes-base.postgresTool",
"typeVersion": 2.6,
"position": [
  1424,
  2160
],
"id": "2b151c42-c8f8-40f3-b70b-58b9d2e5bd9b",
"name": "actualizar_Producto",
"credentials": {
  "postgres": {
    "id": "UJKYtKrs5PWz8OLc",
    "name": "Postgres account 2"
  }
}
},
{
  "parameters": {

```

```
"operation": "update",
"schema": {
  "__rl": true,
  "mode": "list",
  "value": "public"
},
"table": {
  "__rl": true,
  "value": "productos",
  "mode": "list",
  "cachedResultName": "productos"
},
"columns": {
  "mappingMode": "defineBelow",
  "value": {
    "activo": true,
    "id": 0,
    "precioUnidad": 0,
    "stock_numerico": 0
  },
  "matchingColumns": [
    "id"
  ],
  "schema": [
    {
      "id": "id",
      "displayName": "id",
      "required": false,
      "defaultMatch": true,
      "display": true,
      "type": "number",
```

```
"canBeUsedToMatch": true,  
"removed": false  
},  
{  
  "id": "created_at",  
  "displayName": "created_at",  
  "required": false,  
  "defaultMatch": false,  
  "display": true,  
  "type": "dateTime",  
  "canBeUsedToMatch": true  
},  
{  
  "id": "tipo_Producto",  
  "displayName": "tipo_Producto",  
  "required": false,  
  "defaultMatch": false,  
  "display": true,  
  "type": "string",  
  "canBeUsedToMatch": true  
},  
{  
  "id": "Descripcion",  
  "displayName": "Descripcion",  
  "required": false,  
  "defaultMatch": false,  
  "display": true,  
  "type": "string",  
  "canBeUsedToMatch": true  
},  
{
```

```
"id": "precioUnidad",
"displayName": "precioUnidad",
"required": false,
"defaultMatch": false,
"display": true,
"type": "number",
"canBeUsedToMatch": true
},
{
  "id": "stock_numerico",
  "displayName": "stock_numerico",
  "required": false,
  "defaultMatch": false,
  "display": true,
  "type": "number",
  "canBeUsedToMatch": true
},
{
  "id": "activo",
  "displayName": "activo",
  "required": false,
  "defaultMatch": false,
  "display": true,
  "type": "boolean",
  "canBeUsedToMatch": true
}
],
"attemptToConvertTypes": false,
"convertFieldsToString": false
},
"options": {}
```

```

},
"type": "n8n-nodes-base.postgresTool",
"typeVersion": 2.6,
"position": [
  2496,
  1824
],
"id": "8eb2ae97-fd21-48ce-8c87-0dd44a2524c0",
"name": "actualizar_Producto1",
"credentials": {
  "postgres": {
    "id": "UJKYtKrs5PWz8OLc",
    "name": "Postgres account 2"
  }
}
},
{
  "parameters": {
    "formTitle": "registrarProducto",
    "formFields": {
      "values": [
        {
          "fieldLabel": "Id"
        },
        {
          "fieldLabel": "Tipo de Producto"
        },
        {
          "fieldLabel": "Descripcion"
        }
      ]
    }
  }
}

```

```

    "fieldLabel": "Precio Unidad"
  },
  {
    "fieldLabel": "Stock"
  }
]
},
"options": {}
},
"type": "n8n-nodes-base.formTrigger",
"typeVersion": 2.5,
"position": [
  224,
  2208
],
"id": "5fb9b96e-a39d-4623-ae63-de50604ff3e3",
"name": "On form submission",
"webhookId": "397b95c5-0ef7-4f58-82ff-5a428615e6d7"
},
{
  "parameters": {
    "operation": "select",
    "schema": {
      "__rl": true,
      "mode": "list",
      "value": "public"
    }
  },
  "table": {
    "__rl": true,
    "value": "productos",
    "mode": "list",

```

```
"cachedResultName": "productos"
},
"where": {
  "values": [
    {
      "column": "id",
      "value": "Id"
    },
    {
      "column": "tipo_Producto",
      "value": "tipo_Producto"
    },
    {
      "column": "Descripcion",
      "value": "Descripcion"
    },
    {
      "column": "precioUnidad",
      "value": "precioUnidad"
    },
    {
      "column": "stock_numerico",
      "value": "stock_numerico"
    },
    {
      "column": "activo",
      "value": "true"
    }
  ]
},
"options": {}
```

```

},
"type": "n8n-nodes-base.postgresTool",
"typeVersion": 2.6,
"position": [
  1696,
  1824
],
"id": "06d27aaa-5aeb-458b-af4f-a9bc35c7c84e",
"name": "obtener Producto",
"credentials": {
  "postgres": {
    "id": "UJKYtKrs5PWz8OLc",
    "name": "Postgres account 2"
  }
}
},
{
  "parameters": {
    "assignments": {
      "assignments": [
        {
          "id": "eebf8b26-c20b-45eb-9f68-18845d03e48a",
          "name": "",
          "value": "Id",
          "type": "string"
        },
        {
          "id": "9011a6aa-37e6-41d5-ab08-24e7b3007576",
          "name": "",
          "value": "",
          "type": "string"
        }
      ]
    }
  }
}

```

```
    }
  ]
},
"options": {}
},
"type": "n8n-nodes-base.set",
"typeVersion": 3.4,
"position": [
  432,
  2208
],
"id": "e9e3aae7-14ca-4a8d-b94e-0cd15fc39a31",
"name": "Edit Fields1"
},
{
  "parameters": {
    "formTitle": "registrar nuevo Producto",
    "formFields": {
      "values": [
        {
          "fieldLabel": "Id",
          "fieldType": "number"
        },
        {
          "fieldLabel": "Tipo de Producto"
        },
        {
          "fieldLabel": "Descripcion"
        },
        {
          "fieldLabel": "Precio Unidad"
        }
      ]
    }
  }
}
```

```

    },
    {
      "fieldLabel": "Stock"
    },
    {
      "fieldLabel": "Estado",
      "defaultValue": "true"
    }
  ]
},
"options": {}
},
"type": "n8n-nodes-base.formTrigger",
"typeVersion": 2.5,
"position": [
  208,
  2432
],
"id": "07cc4bc0-e82f-4452-bdfc-499aa6aee344",
"name": "On form submission1",
"webhookId": "ea19c1e4-2d98-485e-acbb-7f3777ec0b8a"
},
{
  "parameters": {
    "operation": "update",
    "schema": {
      "__rl": true,
      "mode": "list",
      "value": "public"
    }
  },
  "table": {

```

```
"__rl": true,  
"value": "productos",  
"mode": "list",  
"cachedResultName": "productos"  
},  
"columns": {  
  "mappingMode": "defineBelow",  
  "value": {  
    "activo": false,  
    "id": 0,  
    "precioUnidad": 0,  
    "stock_numerico": 0  
  },  
  "matchingColumns": [  
    "id"  
  ],  
  "schema": [  
    {  
      "id": "id",  
      "displayName": "id",  
      "required": false,  
      "defaultMatch": true,  
      "display": true,  
      "type": "number",  
      "canBeUsedToMatch": true,  
      "removed": false  
    },  
    {  
      "id": "created_at",  
      "displayName": "created_at",  
      "required": false,  
      "removed": false
```

```
"defaultMatch": false,
"display": true,
"type": "dateTime",
"canBeUsedToMatch": true,
"removed": false
},
{
  "id": "tipo_Producto",
  "displayName": "tipo_Producto",
  "required": false,
  "defaultMatch": false,
  "display": true,
  "type": "string",
  "canBeUsedToMatch": true,
  "removed": false
},
{
  "id": "Descripcion",
  "displayName": "Descripcion",
  "required": false,
  "defaultMatch": false,
  "display": true,
  "type": "string",
  "canBeUsedToMatch": true,
  "removed": false
},
{
  "id": "precioUnidad",
  "displayName": "precioUnidad",
  "required": false,
  "defaultMatch": false,
```

```
"display": true,
"type": "number",
"canBeUsedToMatch": true,
"removed": false
},
{
  "id": "stock_numerico",
  "displayName": "stock_numerico",
  "required": false,
  "defaultMatch": false,
  "display": true,
  "type": "number",
  "canBeUsedToMatch": true,
  "removed": false
},
{
  "id": "activo",
  "displayName": "activo",
  "required": false,
  "defaultMatch": false,
  "display": true,
  "type": "boolean",
  "canBeUsedToMatch": true,
  "removed": false
}
],
"attemptToConvertTypes": false,
"convertFieldsToString": false
},
"options": {}
},
```

```

"type": "n8n-nodes-base.postgres",
"typeVersion": 2.6,
"position": [
  640,
  2208
],
"id": "b1728f06-aae7-4f60-849b-02b84b4f0486",
"name": "actualizar_Productos",
"credentials": {
  "postgres": {
    "id": "UJKYtKrs5PWz8OLc",
    "name": "Postgres account 2"
  }
},
{
  "parameters": {
    "operation": "upsert",
    "schema": {
      "__rl": true,
      "mode": "list",
      "value": "public"
    },
    "table": {
      "__rl": true,
      "value": "productos",
      "mode": "list",
      "cachedResultName": "productos"
    },
    "columns": {
      "mappingMode": "defineBelow",

```

```

"value": {
  "activo": true,
  "id": "={{ $json.Id }}",
  "tipo_Producto": "={{ $json['Tipo de Producto'] }}",
  "Descripcion": "={{ $json.Descripcion }}",
  "precioUnidad": "={{ $json['Precio Unidad'] }}",
  "stock_numerico": "={{ $json.Stock }}"
},
"matchingColumns": [
  "id"
],
"schema": [
  {
    "id": "id",
    "displayName": "id",
    "required": false,
    "defaultMatch": true,
    "display": true,
    "type": "number",
    "canBeUsedToMatch": true
  },
  {
    "id": "created_at",
    "displayName": "created_at",
    "required": false,
    "defaultMatch": false,
    "display": true,
    "type": "dateTime",
    "canBeUsedToMatch": false
  },
  {

```

```
"id": "tipo_Producto",
"displayName": "tipo_Producto",
"required": false,
"defaultMatch": false,
"display": true,
"type": "string",
"canBeUsedToMatch": false
},
{
  "id": "Descripcion",
  "displayName": "Descripcion",
  "required": false,
  "defaultMatch": false,
  "display": true,
  "type": "string",
  "canBeUsedToMatch": false
},
{
  "id": "precioUnidad",
  "displayName": "precioUnidad",
  "required": false,
  "defaultMatch": false,
  "display": true,
  "type": "number",
  "canBeUsedToMatch": false
},
{
  "id": "stock_numerico",
  "displayName": "stock_numerico",
  "required": false,
  "defaultMatch": false,
```

```

    "display": true,
    "type": "number",
    "canBeUsedToMatch": false
  },
  {
    "id": "activo",
    "displayName": "activo",
    "required": false,
    "defaultMatch": false,
    "display": true,
    "type": "boolean",
    "canBeUsedToMatch": false
  }
],
"attemptToConvertTypes": false,
"convertFieldsToString": false
},
"options": {}
},
"type": "n8n-nodes-base.postgres",
"typeVersion": 2.6,
"position": [
  416,
  2432
],
"id": "1eb1889a-7675-41ec-b707-ea4ba9bf54d7",
"name": "registra_Nuevos_Productos",
"credentials": {
  "postgres": {
    "id": "UJKYtKrs5PWz8OLc",
    "name": "Postgres account 2"
  }
}

```

```
    }  
  }  
},  
{  
  "parameters": {  
    "formTitle": "registro_Compra",  
    "formFields": {  
      "values": [  
        {  
          "fieldLabel": "id_productos",  
          "fieldType": "number"  
        },  
        {  
          "fieldLabel": "nombreusuario"  
        },  
        {  
          "fieldLabel": "cedula"  
        },  
        {  
          "fieldLabel": "direccion"  
        },  
        {  
          "fieldLabel": "telefono"  
        },  
        {  
          "fieldLabel": "tipo_producto"  
        },  
        {  
          "fieldLabel": "descripcion"  
        },  
        {
```

```

    "fieldLabel": "preciounidad"
  },
  {
    "fieldLabel": "stock_numerico",
    "fieldType": "number"
  },
  {
    "fieldLabel": "ciudad"
  }
]
},
"options": {}
},
"type": "n8n-nodes-base.formTrigger",
"typeVersion": 2.5,
"position": [
  704,
  2448
],
"id": "2fb1977f-40d9-4eb7-80c7-72a428b787e8",
"name": "compras",
"webhookId": "f8758546-f36b-47f7-b328-6b4b5f8ec3b8"
},
{
  "parameters": {
    "operation": "upsert",
    "schema": {
      "__rl": true,
      "mode": "list",
      "value": "public"
    }
  },

```

```
"table": {
  "__rl": true,
  "value": "compras",
  "mode": "list",
  "cachedResultName": "compras"
},
"columns": {
  "mappingMode": "defineBelow",
  "value": {
    "id_productos": "={{ $json.id_productos }}",
    "cedula": "={{ $json.cedula }}",
    "telefono": "={{ $json.telefono }}",
    "preciounidad": "={{ $json.preciounidad }}",
    "stock_numerico": "={{ $json.stock_numerico }}",
    "nombreusuario": "={{ $json.nombreusuario }}",
    "direccion": "={{ $json.direccion }}",
    "ciudad": "={{ $json.ciudad }}",
    "tipo_producto": "={{ $json.tipo_producto }}",
    "descripcion": "={{ $json.descripcion }}"
  },
  "matchingColumns": [
    "id_productos"
  ],
  "schema": [
    {
      "id": "id_productos",
      "displayName": "id_productos",
      "required": true,
      "defaultMatch": false,
      "display": true,
      "type": "number",
```

```
"canBeUsedToMatch": true,  
"removed": false  
},  
{  
  "id": "nombreusuario",  
  "displayName": "nombreusuario",  
  "required": false,  
  "defaultMatch": false,  
  "display": true,  
  "type": "string",  
  "canBeUsedToMatch": false,  
  "removed": false  
},  
{  
  "id": "cedula",  
  "displayName": "cedula",  
  "required": false,  
  "defaultMatch": false,  
  "display": true,  
  "type": "number",  
  "canBeUsedToMatch": false,  
  "removed": false  
},  
{  
  "id": "direccion",  
  "displayName": "direccion",  
  "required": false,  
  "defaultMatch": false,  
  "display": true,  
  "type": "string",  
  "canBeUsedToMatch": false,
```

```
"removed": false
},
{
  "id": "telefono",
  "displayName": "telefono",
  "required": false,
  "defaultMatch": false,
  "display": true,
  "type": "number",
  "canBeUsedToMatch": false,
  "removed": false
},
{
  "id": "ciudad",
  "displayName": "ciudad",
  "required": false,
  "defaultMatch": false,
  "display": true,
  "type": "string",
  "canBeUsedToMatch": false,
  "removed": false
},
{
  "id": "tipo_producto",
  "displayName": "tipo_producto",
  "required": false,
  "defaultMatch": false,
  "display": true,
  "type": "string",
  "canBeUsedToMatch": false,
  "removed": false
}
```

```
},  
{  
  "id": "descripcion",  
  "displayName": "descripcion",  
  "required": false,  
  "defaultMatch": false,  
  "display": true,  
  "type": "string",  
  "canBeUsedToMatch": false,  
  "removed": false  
},  
{  
  "id": "preciounidad",  
  "displayName": "preciounidad",  
  "required": false,  
  "defaultMatch": false,  
  "display": true,  
  "type": "number",  
  "canBeUsedToMatch": false,  
  "removed": false  
},  
{  
  "id": "stock_numerico",  
  "displayName": "stock_numerico",  
  "required": false,  
  "defaultMatch": false,  
  "display": true,  
  "type": "number",  
  "canBeUsedToMatch": false,  
  "removed": false  
}
```

```

    ],
    "attemptToConvertTypes": false,
    "convertFieldsToString": false
  },
  "options": {}
},
"type": "n8n-nodes-base.postgres",
"typeVersion": 2.6,
"position": [
  912,
  2448
],
"id": "de522d45-ab80-49a5-a874-3290914782a6",
"name": "Insert or update rows in a table",
"credentials": {
  "postgres": {
    "id": "UJKYtKrs5PWz8OLc",
    "name": "Postgres account 2"
  }
}
},
{
  "parameters": {
    "updates": [
      "message"
    ],
  },
  "additionalFields": {}
},
"type": "n8n-nodes-base.telegramTrigger",
"typeVersion": 1.2,
"position": [

```

```

-96,
1552
],
"id": "9690ed94-b723-4afa-ba9f-b419766448c4",
"name": "Telegram Trigger",
"webhookId": "625485d5-9cb9-4764-a4b4-a36f5153725e",
"credentials": {
  "telegramApi": {
    "id": "2osvMZvKxHX1ZQhb",
    "name": "Telegram account"
  }
}
},
{
"parameters": {
  "formTitle": "googleSheets",
  "formFields": {
    "values": [
      {
        "fieldLabel": "id",
        "fieldType": "number"
      },
      {
        "fieldLabel": "tipo_Producto"
      },
      {
        "fieldLabel": "Descripcion"
      },
      {
        "fieldLabel": "precioUnidad"
      },
    ],
  }
}

```

```

    {
      "fieldLabel": "stock_numerico",
      "fieldType": "number"
    },
    {
      "fieldLabel": "activo"
    }
  ]
},
"options": {}
},
"type": "n8n-nodes-base.formTrigger",
"typeVersion": 2.5,
"position": [
  1680,
  2448
],
"id": "eb445324-23d8-47f9-8464-5caf88a896b3",
"name": "On form submission2",
"webhookId": "9ac09bb4-cb32-41f7-9fd5-b6303cb3ce48"
},
{
  "parameters": {
    "formTitle": "googleSheets",
    "formFields": {
      "values": [
        {
          "fieldLabel": "id",
          "fieldType": "number"
        },
        {

```

```
    "fieldLabel": "tipo_Producto"
  },
  {
    "fieldLabel": "Descripcion"
  },
  {
    "fieldLabel": "precioUnidad"
  },
  {
    "fieldLabel": "stock_numerico",
    "fieldType": "number"
  },
  {
    "fieldLabel": "activo"
  }
]
},
"options": {},
},
"type": "n8n-nodes-base.formTrigger",
"typeVersion": 2.5,
"position": [
  2288,
  2800
],
"id": "7923e51e-078e-4b23-a83b-c1a23cee08ff",
"name": "registrar_Productos_Sheets",
"webhookId": "a5bd0aa0-85f0-46f9-834c-0c66a2c6138c"
},
{
  "parameters": {
```

```

"assignments": {
  "assignments": [
    {
      "id": "c8022a2b-deed-4499-8cd9-d9a9c538be0c",
      "name": "var_saludo_Admin",
      "value": "!Hola admin como estas!",
      "type": "string"
    },
    {
      "id": "dc070076-97c9-4ed1-a3b3-90586617f0b0",
      "name": "var_saludo_User",
      "value": "!Hola usuario como estas",
      "type": "string"
    },
    {
      "id": "5fc85941-2df0-4212-a01f-7e80eb8fc40f",
      "name": "chat_id",
      "value": "={{ $json.message.chat.id }}",
      "type": "number"
    }
  ]
},
"includeOtherFields": true,
"options": {}
},
"type": "n8n-nodes-base.set",
"typeVersion": 3.4,
"position": [
  928,
  1664
],

```

```

    "id": "01d948b6-0703-42f2-91fd-eba02d7ff4b4",
    "name": "var_Saludo",
    "alwaysOutputData": true
  },
  {
    "parameters": {
      "chatId": "{{ $json.chat_id }}",
      "text": "={{ $json.var_saludo_Admin }}",
      "additionalFields": {}
    },
    "type": "n8n-nodes-base.telegram",
    "typeVersion": 1.2,
    "position": [
      2224,
      1552
    ],
    "id": "558a8095-3d83-4c30-84fe-fdab9cbfc5bd",
    "name": "saludarAdmin",
    "webhookId": "67258fae-bc6f-4bee-9646-46935ae84d87",
    "alwaysOutputData": true,
    "credentials": {
      "telegramApi": {
        "id": "2osvMZvKxHX1ZQhb",
        "name": "Telegram account"
      }
    }
  },
  {
    "parameters": {
      "promptType": "define",
      "text": "={{ $('Merge').item.json.message.text }}"
    }
  }

```

```
"options": {
  "systemMessage": "# 🧠 SYSTEM PROMPT UNIFICADO — PANDORA + ADMIN
  PACKMAN\n\n### 🎯 OBJETIVO GENERAL\n\nAsistente híbrido que:\n\n* Ayuda a
  clientes a encontrar productos\n\n* Controla acceso (Admin / User / Cliente)\n\n* Ejecuta
  acciones usando herramientas\n\n* Nunca inventa datos\n\n### 👤 TIPOS DE
  ACTOR\n\n#### 1. ADMIN\n\n* Requiere autenticación (ID Telegram)\n\n* Puede:\n\n *
  Registrar productos\n\n * Actualizar productos\n\n * Eliminar productos\n\n * Consultar
  productos\n\n#### 2. USER\n\n* Requiere autenticación\n\n* Puede:\n\n * Consultar
  productos\n\n * Insertar productos (si aplica)\n\n* ❌ NO puede modificar ni
  eliminar\n\n#### 3. CLIENTE (USUARIO NORMAL)\n\n* No requiere autenticación\n\n*
  Solo consulta productos (flujo Pandora)\n\n### 🗝️ CONTROL DE ACCESO\n\n#### IDs
  válidos:\n\n* `ADMIN_ID = 6050993276`\n\n* `USER_ID = 6050993277`\n\n####
  Flujo:\n\n909. Detectar si dice “admin” o “user”\n\n910. Solicitar ID\n\n911. Validar:\n\n
  * Si coincide → permitir acciones según rol\n\n * Si no → ❌ denegar acceso\n\n912.
  Si no declara rol → cliente normal\n\n### 🛠️ HERRAMIENTAS\n\n* `obtener_Producto`
  → consultar productos\n\n* `registrar_Producto` → crear producto\n\n*
  `actualizar_Producto` → editar producto\n\n* `borrar_Producto` → eliminar producto\n\n*
  `mostrar_Formulario_compras` → compras\n\n### 🧠 FLUJO INTELIGENTE\n\n#### 1.
  ENTENDER\n\nDetectar intención:\n\n* “productos” → listado\n\n* nombre específico →
  detalle\n\n* admin/user → autenticación\n\n#### 2. CLASIFICAR\n\n* Cliente →
  consulta\n\n* Admin → CRUD completo\n\n* User → consulta limitada\n\n#### 3.
  VALIDAR\n\n* Nunca ejecutar sin datos completos\n\n* Nunca saltar
  autenticación\n\n#### 4. EJECUTAR\n\n* Si requiere datos → usar herramienta\n\n
  Respuesta:\n\n * Tool → SOLO tool call\n\n * Cliente → respuesta natural\n\n### 🚩
  REGLAS CRÍTICAS\n\n* ✅ SIEMPRE usar herramientas cuando aplique\n\n* ❌
  NUNCA inventar productos\n\n* ❌ NUNCA ejecutar acciones sin validar ID\n\n* ❌
  NUNCA responder texto si hay tool\n\n* ❌ NO confundir roles\n\n* ✅ Respuestas
  claras, útiles y naturales\n\n### ⚠️ CASOS CLAVE\n\n* “productos” → usar
  `obtener_Producto`\n\n* producto específico → filtrar\n\n* sin datos → pedir
```

información\n\n* muchos resultados → resumir\n\n* no existe → decirlo claro\n\n## 🚫
 PROHIBIDO\n\n* Inventar datos\n\n* Saltar autenticación\n\n* Respuestas robóticas\n\n*
 Ejecutar acciones incompletas\n\n* Dar acceso incorrecto\n\n## 🚩 RESULTADO\n\nUn
 agente que:\n\n* Controla acceso correctamente\n\n* Usa herramientas sin error\n\n*
 Responde como asesor real\n\n* Optimiza cada interacción\n"

```

    }
  },
  "type": "@n8n/n8n-nodes-langchain.agent",
  "typeVersion": 3.1,
  "position": [
    1872,
    1488
  ],
  "id": "468a5614-c0e6-416c-9d4e-435ddad0947f",
  "name": "AI Agent2"
},
{
  "parameters": {
    "conditions": {
      "options": {
        "caseSensitive": true,
        "leftValue": "",
        "typeValidation": "strict",
        "version": 3
      },
      "conditions": [
        {
          "id": "fc7bfa9c-44df-4aa3-a498-7dc7aafc5656",
          "leftValue": "{{ $json.message.chat.id }}",
          "rightValue": "6050993276",
          "operator": {

```

```

    "type": "string",
    "operation": "equals",
    "name": "filter.operator.equals"
  }
}
],
"combinator": "and"
},
"options": {}
},
"type": "n8n-nodes-base.if",
"typeVersion": 2.3,
"position": [
  1712,
  1408
],
"id": "59b8ad90-8e69-4829-ac81-38517ae343b7",
"name": "If1"
},
{
  "parameters": {
    "options": {}
  },
  "type": "@n8n/n8n-nodes-langchain.lmChatGoogleGemini",
  "typeVersion": 1,
  "position": [
    1840,
    1760
  ],
  "id": "cd46f08d-73ea-4ec2-98a9-05f166c78fe6",
  "name": "Google Gemini Chat Model2",

```

```

"credentials": {
  "googlePalmApi": {
    "id": "Xo2Svwmq2MwLMqv6",
    "name": "Google Gemini(PaLM) Api account 3"
  }
},
{
  "parameters": {},
  "type": "n8n-nodes-base.wait",
  "typeVersion": 1.1,
  "position": [
    1248,
    1536
  ],
  "id": "1a4d9a67-b786-4a70-9a0c-33a2aae30662",
  "name": "Wait",
  "webhookId": "98007d56-bd16-45e0-84dd-09e7d7f8c62c"
},
{
  "parameters": {},
  "type": "n8n-nodes-base.wait",
  "typeVersion": 1.1,
  "position": [
    704,
    1920
  ],
  "id": "f7e4ddfc-802d-4869-be81-bd31dd9426dd",
  "name": "Wait1",
  "webhookId": "e8237635-6729-4aca-8193-2b16b6dff8f4"
},

```

```

{
  "parameters": {
    "conditions": {
      "options": {
        "caseSensitive": true,
        "leftValue": "",
        "typeValidation": "strict",
        "version": 3
      },
      "conditions": [
        {
          "id": "ee5785d3-07de-4619-87b7-31ff222c451f",
          "leftValue": "{{${json.message.text}}",
          "rightValue": "=^\\d+$",
          "operator": {
            "type": "string",
            "operation": "regex"
          }
        }
      ],
      "combinator": "and"
    },
    "options": {}
  },
  "type": "n8n-nodes-base.if",
  "typeVersion": 2.3,
  "position": [
    1520,
    1968
  ],
  "id": "9890a44e-6ed9-42b3-9c8e-c2e721e82a40",

```

```
















"name": "lf2"
},
{
"parameters": {
"promptType": "define",
"text": "={{ $('Merge').item.json.message.text }}",
"options": {

```

```

"systemMessage": "# 🧠 SYSTEM PROMPT UNIFICADO — PANDORA + ADMIN
PACKMAN\n\n## 🎯 OBJETIVO GENERAL\n\nAsistente híbrido que:\n\n* Ayuda a
clientes a encontrar productos\n\n* Controla acceso (Admin / User / Cliente)\n\n* Ejecuta
acciones usando herramientas\n\n* Nunca inventa datos\n\n## 👤 TIPOS DE
ACTOR\n\n### 1. ADMIN\n\n* Requiere autenticación (ID Telegram)\n\n* Puede:\n\n *
Registrar productos\n\n * Actualizar productos\n\n * Eliminar productos\n\n * Consultar
productos\n\n### 2. USER\n\n* Requiere autenticación\n\n* Puede:\n\n * Consultar
productos\n\n * Insertar productos (si aplica)\n\n* ❌ NO puede modificar ni
eliminar\n\n### 3. CLIENTE (USUARIO NORMAL)\n\n* No requiere autenticación\n\n*
Solo consulta productos (flujo Pandora)\n\n## 🗝️ CONTROL DE ACCESO\n\n### IDs
válidos:\n\n* `ADMIN_ID = 6050993276`\n\n* `USER_ID = 6050993277`\n\n###
Flujo:\n\n909. Detectar si dice “admin” o “user”\n\n910. Solicitar ID\n\n911. Validar:\n\n
* Si coincide → permitir acciones según rol\n\n * Si no → ❌ denegar acceso\n\n912.
Si no declara rol → cliente normal\n\n## 🛠️ HERRAMIENTAS\n\n* `obtener_Producto1`
→ consultar productos\n\n* `registrar_Producto` → crear producto\n\n*
`actualizar_Producto` → editar producto\n\n* `borrar_Producto` → eliminar producto\n\n
* `mostrar_Formulario_compras` → compras\n\n## 🧠 FLUJO INTELIGENTE\n\n### 1.
ENTENDER\n\nDetectar intención:\n\n* “productos” → listado\n\n* nombre específico →
detalle\n\n* admin/user → autenticación\n\n### 2. CLASIFICAR\n\n* Cliente →
consulta\n\n* Admin → CRUD completo\n\n* User → consulta limitada\n\n### 3.
VALIDAR\n\n* Nunca ejecutar sin datos completos\n\n* Nunca saltar
autenticación\n\n### 4. EJECUTAR\n\n* Si requiere datos → usar herramienta\n\n
Respuesta:\n\n * Tool → SOLO tool call\n\n * Cliente → respuesta natural\n\n## 🚀

```

REGLAS CRÍTICAS  SIEMPRE usar herramientas cuando aplique 
 NUNCA inventar productos  NUNCA ejecutar acciones sin validar ID 
 NUNCA responder texto si hay tool  NO confundir roles  Respuestas
 claras, útiles y naturales  CASOS CLAVE "productos" → usar
 `obtener_Producto` producto específico → filtrar sin datos → pedir
 información muchos resultados → resumir no existe → decirlo claro 
 PROHIBIDO Inventar datos Saltar autenticación Respuestas robóticas
 Ejecutar acciones incompletas Dar acceso incorrecto  RESULTADO Un
 agente que: Controla acceso correctamente Usa herramientas sin error
 Responde como asesor real Optimiza cada interacción  NUNCA responder
 texto si hay tool  NO confundir roles  Respuestas claras, útiles y
 naturales  CASOS CLAVE "productos" → usar `obtener_Producto1`
 producto específico → filtrar sin datos → pedir información muchos resultados
 → resumir no existe → decirlo claro  PROHIBIDO Inventar datos
 Saltar autenticación Respuestas robóticas Ejecutar acciones incompletas
 Dar acceso incorrecto  RESULTADO Un agente que: Controla acceso
 correctamente Usa herramientas sin error Responde como asesor real
 Optimiza cada interacción

```

    }
  },
  "type": "@n8n/n8n-nodes-langchain.agent",
  "typeVersion": 3.1,
  "position": [
    1696,
    2048
  ],
  "id": "badb4066-070d-4580-b7f8-70f14d444bdc",
  "name": "AI Agent3",
  "retryOnFail": true,

```

```

    "waitBetweenTries": 2000
  },
  {
    "parameters": {
      "options": {}
    },
    "type": "@n8n/n8n-nodes-langchain.lmChatGoogleGemini",
    "typeVersion": 1,
    "position": [
      1696,
      2272
    ],
    "id": "73d4eec3-fc71-40ac-919d-3ea60d70d5de",
    "name": "Google Gemini Chat Model3",
    "credentials": {
      "googlePalmApi": {
        "id": "Xo2Svwmq2MwLMqv6",
        "name": "Google Gemini(PaLM) Api account 3"
      }
    }
  },
  {
    "parameters": {
      "sessionIdType": "customKey",
      "sessionKey": "{{\"admin_\"+ $json.message.chat.id}}"
    },
    "type": "@n8n/n8n-nodes-langchain.memoryBufferWindow",
    "typeVersion": 1.3,
    "position": [
      1936,
      1744
    ]
  }
}

```

```
],  
  "id": "483cde81-61f8-48ba-9d46-5be275fe99d2",  
  "name": "Simple Memory2"  
},  
{  
  "parameters": {  
    "operation": "select",  
    "schema": {  
      "__rl": true,  
      "mode": "list",  
      "value": "public"  
    },  
    "table": {  
      "__rl": true,  
      "mode": "list",  
      "value": "productos"  
    },  
    "returnAll": true,  
    "where": {  
      "values": []  
    },  
    "options": {}  
  },  
  "type": "n8n-nodes-base.postgresTool",  
  "typeVersion": 2.6,  
  "position": [  
    1584,  
    2176  
  ],  
  "id": "efd425b5-7e95-435b-a02a-b23248428ac7",  
  "name": "obtener_Producto",
```

```
"credentials": {
  "postgres": {
    "id": "UJKYtKrs5PWz8OLc",
    "name": "Postgres account 2"
  }
},
{
  "parameters": {
    "operation": "select",
    "schema": {
      "__rl": true,
      "mode": "list",
      "value": "public"
    },
    "table": {
      "__rl": true,
      "mode": "list",
      "value": "productos"
    },
    "returnAll": true,
    "where": {
      "values": []
    },
    "options": {}
  },
  "type": "n8n-nodes-base.postgresTool",
  "typeVersion": 2.6,
  "position": [
    2128,
    2304
```

```

],
  "id": "ddc5d94e-680e-45a4-9243-23d0e17c6b75",
  "name": "obtener_Producto1",
  "credentials": {
    "postgres": {
      "id": "UJKYtKrs5PWz8OLc",
      "name": "Postgres account 2"
    }
  }
},
{
  "parameters": {
    "url":
"==https://api.telegram.org/file/bot8755454772:AAHmaJcSCpxlMeKrEe71JvucZUeG1A
ka1Rk/{{ $json.result.file_path }}",
    "options": {
      "response": {
        "response": {
          "responseFormat": "file"
        }
      }
    }
  },
  "type": "n8n-nodes-base.httpRequest",
  "typeVersion": 4.4,
  "position": [
    -176,
    2208
  ],
  "id": "5c44308a-9ec7-40e7-822c-bdf22099210a",
  "name": "HTTP Request",

```

```
    "alwaysOutputData": true
  }
],
"pinData": {},
"connections": {
  "Google Gemini Chat Model": {
    "ai_languageModel": [
      [
        {
          "node": "AI Agent",
          "type": "ai_languageModel",
          "index": 0
        }
      ]
    ]
  },
  "AI Agent": {
    "main": [
      [
        {
          "node": "If1",
          "type": "main",
          "index": 0
        }
      ]
    ]
  },
  "Simple Memory": {
    "ai_memory": [
      [
        {
```

```
    "node": "AI Agent",
    "type": "ai_memory",
    "index": 0
  }
]
],
},
"lf": {
  "main": [
    [
      {
        "node": "Wait",
        "type": "main",
        "index": 0
      }
    ],
    [
      {
        "node": "Wait1",
        "type": "main",
        "index": 0
      }
    ]
  ]
},
"Google Gemini Chat Model1": {
  "ai_languageModel": [
    [
      {
        "node": "AI Agent1",
        "type": "ai_languageModel",
```

```
    "index": 0
  }
]
]
},
"AI Agent1": {
  "main": [
    [
      {
        "node": "lf2",
        "type": "main",
        "index": 0
      }
    ]
  ]
},
"Simple Memory1": {
  "ai_memory": [
    [
      {
        "node": "AI Agent1",
        "type": "ai_memory",
        "index": 0
      }
    ]
  ]
},
"Transcribe a recording": {
  "main": [
    []
  ]
}
```

```
},  
"Switch": {  
  "main": [  
    [  
      {  
        "node": "Merge",  
        "type": "main",  
        "index": 1  
      }  
    ],  
    [  
      {  
        "node": "entrada_Audio",  
        "type": "main",  
        "index": 0  
      }  
    ],  
    []  
  ]  
},  
"entrada_Audio": {  
  "main": [  
    [  
      {  
        "node": "HTTP Request",  
        "type": "main",  
        "index": 0  
      }  
    ]  
  ]  
},
```

```
"Merge": {
  "main": [
    [
      {
        "node": "var_Saludo",
        "type": "main",
        "index": 0
      }
    ]
  ]
},
"registrar_Producto": {
  "ai_tool": [
    [
      {
        "node": "AI Agent1",
        "type": "ai_tool",
        "index": 0
      }
    ]
  ]
},
"registrar_Productos": {
  "ai_tool": [
    [
      {
        "node": "AI Agent",
        "type": "ai_tool",
        "index": 0
      }
    ]
  ]
}
```

```
]
},
"borrar_Producto": {
  "ai_tool": [
    [
      {
        "node": "AI Agent",
        "type": "ai_tool",
        "index": 0
      }
    ]
  ]
},
"borrar_Producto1": {
  "ai_tool": [
    [
      {
        "node": "AI Agent1",
        "type": "ai_tool",
        "index": 0
      }
    ]
  ]
},
"actualizar_Producto": {
  "ai_tool": [
    [
      {
        "node": "AI Agent1",
        "type": "ai_tool",
        "index": 0
      }
    ]
  ]
}
```

```
    }
  ]
]
},
"actualizar_Producto1": {
  "ai_tool": [
    [
      {
        "node": "AI Agent",
        "type": "ai_tool",
        "index": 0
      }
    ]
  ]
},
"obtener Producto": {
  "ai_tool": [
    [
      {
        "node": "AI Agent",
        "type": "ai_tool",
        "index": 0
      }
    ]
  ]
},
"On form submission": {
  "main": [
    [
      {
        "node": "Edit Fields1",
```

```
    "type": "main",
    "index": 0
  }
]
],
},
"Edit Fields1": {
  "main": [
    [
      {
        "node": "actualizar_Productos",
        "type": "main",
        "index": 0
      }
    ]
  ]
},
"On form submission1": {
  "main": [
    [
      {
        "node": "registra_Nuevos_Productos",
        "type": "main",
        "index": 0
      }
    ]
  ]
},
"compras": {
  "main": [
    [
```

```
{
  "node": "Insert or update rows in a table",
  "type": "main",
  "index": 0
}
]
],
},
"Telegram Trigger": {
  "main": [
    [
      {
        "node": "Switch",
        "type": "main",
        "index": 0
      }
    ]
  ]
},
"var_Saludo": {
  "main": [
    [
      {
        "node": "If",
        "type": "main",
        "index": 0
      }
    ]
  ]
},
"AI Agent2": {
```

```
"main": [  
  [  
    {  
      "node": "saludarAdmin",  
      "type": "main",  
      "index": 0  
    }  
  ]  
],  
},  
"If1": {  
  "main": [  
    [  
      {  
        "node": "Send a text message",  
        "type": "main",  
        "index": 0  
      }  
    ],  
    [  
      {  
        "node": "AI Agent2",  
        "type": "main",  
        "index": 0  
      }  
    ]  
  ]  
},  
"Google Gemini Chat Model2": {  
  "ai_languageModel": [  
    [  
      {  
        "node": "Send a text message",  
        "type": "main",  
        "index": 0  
      }  
    ]  
  ]  
}
```

```
{
  "node": "AI Agent2",
  "type": "ai_languageModel",
  "index": 0
}
]
],
},
"Wait": {
  "main": [
    [
      {
        "node": "AI Agent",
        "type": "main",
        "index": 0
      }
    ]
  ]
},
"Wait1": {
  "main": [
    [
      {
        "node": "AI Agent1",
        "type": "main",
        "index": 0
      }
    ]
  ]
},
"lf2": {
```

```
"main": [  
  [  
    {  
      "node": "Send a text message1",  
      "type": "main",  
      "index": 0  
    }  
  ],  
  [  
    {  
      "node": "AI Agent3",  
      "type": "main",  
      "index": 0  
    }  
  ]  
],  
},  
"Google Gemini Chat Model3": {  
  "ai_languageModel": [  
    [  
      {  
        "node": "AI Agent3",  
        "type": "ai_languageModel",  
        "index": 0  
      }  
    ]  
  ]  
},  
"AI Agent3": {  
  "main": [  
    [  
      {  
        "node": "Send a text message1",  
        "type": "main",  
        "index": 0  
      }  
    ],  
    [  
      {  
        "node": "AI Agent3",  
        "type": "main",  
        "index": 0  
      }  
    ]  
  ]  
}
```

```
{
  "node": "saludarUser",
  "type": "main",
  "index": 0
}
]
],
},
"Simple Memory2": {
  "ai_memory": [
    [
      {
        "node": "AI Agent2",
        "type": "ai_memory",
        "index": 0
      }
    ]
  ]
},
"obtener_Producto": {
  "ai_tool": [
    [
      {
        "node": "AI Agent1",
        "type": "ai_tool",
        "index": 0
      }
    ]
  ]
},
"obtener_Producto1": {
```

```
"ai_tool": [
  [
    {
      "node": "AI Agent3",
      "type": "ai_tool",
      "index": 0
    }
  ]
],
"HTTP Request": {
  "main": [
    [
      {
        "node": "Transcribe a recording",
        "type": "main",
        "index": 0
      }
    ]
  ]
},
"active": false,
"settings": {
  "executionOrder": "v1",
  "binaryMode": "separate",
  "timeSavedMode": "fixed",
  "callerPolicy": "workflowsFromSameOwner",
  "availableInMCP": false
},
"versionId": "2b59d31a-9fd3-41dd-9788-d6f658ea4ed2",
```

```
"meta": {  
  "templateCredsSetupCompleted": true,  
  "instanceId":  
"3dfd20fe276e3f45081251a36d7a6347ce8c3af2b53cfb15f044568e4f72bf6c"  
},  
"id": "SxsTrTPODfAi8vnh",  
"tags": []  
}
```

Descripción para las imágenes:

Figura X. Ejemplo de interacción entre el usuario y el agente automatizado a través de Telegram, donde se evidencia la interpretación del mensaje y la generación de una respuesta estructurada.

Figura X. Registro automático de la solicitud en Google Sheets, mostrando la estructuración de los datos capturados por el sistema.

Conclusión

El desarrollo e implementación del flujo “Sistema de clasificación y asignación inteligente de tickets de soporte” permitió demostrar el potencial de la automatización inteligente en la gestión de incidencias técnicas. A través de la plataforma n8n, se integraron de manera eficiente herramientas como Telegram para la comunicación con el usuario, Google Gemini para el análisis de lenguaje natural y Google Sheets como base de datos de registro, conformando un sistema funcional, escalable y de fácil mantenimiento.

La arquitectura del flujo logró cumplir con los objetivos propuestos: recibir solicitudes, interpretar su contenido mediante inteligencia artificial, clasificarlas automáticamente y registrarlas en tiempo real en una hoja de cálculo estructurada. Este proceso permitió reducir la intervención manual del personal de soporte, optimizar los tiempos de respuesta y garantizar una atención más ágil y precisa para el usuario final, Teniendo presente también que es perfectamente adaptable a ventas no solo a soporte con alto grado de escalabilidad empresarial.

Desde una perspectiva técnica, el sistema demostró una alta estabilidad operativa y un manejo eficiente de los datos. La combinación del procesamiento de mensajes de texto y voz, junto con la memoria conversacional del agente, otorgó al flujo una capacidad adaptable ante distintos escenarios de comunicación, incrementando su usabilidad y alcance. Asimismo, la estandarización de la información recopilada posibilitó una mejor organización de los reportes y una trazabilidad completa de cada caso.

En términos de aprendizaje, el proyecto fortaleció las competencias en automatización de procesos, integración de APIs, diseño de flujos lógicos, procesamiento de lenguaje natural y uso de modelos de IA aplicados a contextos reales. Además, evidenció la importancia de aplicar metodologías de desarrollo estructurado y documentado en entornos de automatización sin código.

Referencias bibliográficas

Truss, M., & Böhm, S. (2024). *AI-based classification of customer support tickets: State of the art and implementation with AutoML*. Proceedings of the IWEMB 2022.

<https://doi.org/10.48550/arXiv.2406.01789>

Imran, M., & Almusharraf, N. (2024). Google Gemini as a next-generation AI educational tool: A review. *Smart Learning Environments*, 11(22).

<https://doi.org/10.1186/s40561-024-00310-z>

Bonechi, S. (2024). Leveraging AI for efficient ticket classification. *Procedia Computer Science*, 235, 1234–1243.

Fettke, P. (2025). Business Process Management and Artificial Intelligence. *KI – Künstliche Intelligenz*.

Proser, Z. (2025, March 24). *n8n: The workflow automation tool for the AI age*. WorkOS Blog.

Anexos

Anexo – a código fuente n8n

```
{
  "name": "Telegram Soporte",
  "nodes": [
    {
      "parameters": {
        "options": {}
      },
      "type": "@n8n/n8n-nodes-langchain.ImChatGoogleGemini",
      "typeVersion": 1,
      "position": [
        -944,
        240
      ],
      "id": "575b761f-a620-42b7-89c3-e53f3249d923",
      "name": "Google Gemini Chat Model",
      "credentials": {
        "googlePalmApi": {
          "id": "clrh3uFEmid1kX3F",
          "name": "api key orregol343 - grupo 1"
        }
      }
    }
  ]
},
```

```

{
  "parameters": {
    "conditions": {
      "options": {
        "caseSensitive": true,
        "leftValue": "",
        "typeValidation": "strict",
        "version": 2
      },
      "conditions": [
        {
          "id": "a71dc7ac-277b-43ee-bbab-41ed987b7ff9",
          "leftValue": "={{ $('Bot Orrares').item.json.output.clasificacion }}",
          "rightValue": "red",
          "operator": {
            "type": "string",
            "operation": "contains"
          }
        },
        {
          "id": "64f3d68c-68d4-4458-9ab1-0e59bc7ec535",
          "leftValue": "={{ $('Bot Orrares').item.json.output.clasificacion }}",
          "rightValue": "software",
          "operator": {
            "type": "string",
            "operation": "equals",
            "name": "filter.operator.equals"
          }
        },
        {
          "id": "9ec106e9-ce34-4efb-85f7-a92b630a010b",
          "leftValue": "={{ $('Bot Orrares').item.json.output.clasificacion }}",
          "rightValue": "hardware",
          "operator": {
            "type": "string",
            "operation": "equals",
            "name": "filter.operator.equals"
          }
        }
      ],
      "combinator": "or"
    },
    "options": {}
  },
  "type": "n8n-nodes-base.if",
  "typeVersion": 2.2,

```

```

"position": [
  48,
  -64
],
"id": "e41a125c-8ae5-4374-9950-7156678387e9",
"name": "If"
},
{
  "parameters": {},
  "type": "n8n-nodes-base.noOp",
  "typeVersion": 1,
  "position": [
    224,
    192
  ],
  "id": "ff368acd-4761-47a4-968a-cdd59dba3b19",
  "name": "Cerrar"
},
{
  "parameters": {
    "chatId": "={{ $('Mensaje a recibir').item.json.message.chat.id }}",
    "text": "={{ $json.output.mensaje }}",
    "additionalFields": {
      "appendAttribution": false
    }
  },
  "type": "n8n-nodes-base.telegram",
  "typeVersion": 1.2,
  "position": [
    -144,
    -64
  ],
  "id": "04909c4b-9271-45ab-a99e-f5144f877888",
  "name": "Respuesta al usuario",
  "webhookId": "37aa80d9-778c-4dd7-998d-059b60eaa6d3",
  "credentials": {
    "telegramApi": {
      "id": "7GfwHM2jl21XOFas",
      "name": "Soporte_S.A.S"
    }
  }
},
{
  "parameters": {
    "sessionIdType": "customKey",
    "sessionKey": "={{ $('Mensaje a recibir').item.json.message.chat.id }}"
  }
}

```



```

    "id": "7GfwHM2jl21XOFas",
    "name": "Soporte_S.A.S"
  }
},
{
  "parameters": {
    "promptType": "define",
    "text": "={{ $json.mensaje }} ",
    "hasOutputParser": true,
    "options": {
      "systemMessage": "=Eres SoporteBot, un asistente de soporte técnico
automatizado.\n\nTu tarea es atender solicitudes de usuarios sobre problemas
informaticos, recopilar su informacion de contacto y registrar la incidencia con una
breve clasificacion.\n\n### Tus funciones principales:\n1. Saludar al usuario y pedirle
su nombre, correo electronico y numero de telefono antes de procesar la solicitud.\n2.
Comprender la descripcion del problema que el usuario indica.\n3. Clasificar el tipo de
incidencia en una de estas categorias:\n - hardware (equipos fisicos como pantallas,
mouse, teclado, impresoras, CPU, etc.)\n - software (programas, sistemas operativos,
aplicaciones)\n - red (internet, WiFi, conexiones)\n - otro (cuando no aplica ninguna
de las anteriores)\n4. Generar una respuesta amable, profesional y clara para el
usuario, confirmando que la informacion fue registrada correctamente.\n5. Mantener
siempre el formato de texto simple, sin emojis, tildes ni caracteres especiales.\n6. Si el
usuario no proporciona su informacion de contacto completa, solicitala de forma
educada antes de continuar.\n\n### Estructura obligatoria de salida (en texto plano, no
JSON):\nRespuesta para el usuario:\n[Mensaje breve y amable con confirmacion o
solicitud de informacion]\n\nClasificacion del problema: [hardware | software | red |
otro]\n\nResumen breve: [descripcion corta del problema detectado]\n\n---\n\n###
Comportamiento adicional:\n- Si el usuario solo dice \"hola\" o inicia la conversacion,
responde con un saludo y pide su nombre, correo y numero de telefono.\n- Si ya tiene
los datos del usuario, procede a clasificar el problema.\n- Evita lenguaje tecnico
complicado. Usa terminos simples y comprensibles.\n- Si no puedes determinar la
categoria exacta, usa \"otro\" y pide mas detalles.\n"
    }
  },
  "type": "@n8n/n8n-nodes-langchain.agent",
  "typeVersion": 2.2,
  "position": [
    -672,
    -80
  ],
  "id": "f95c248f-1134-4af0-a000-810f5cc3da7f",
  "name": "Bot Orrares"
},
{
  "parameters": {

```

```

"operation": "append",
"documentId": {
  "__rl": true,
  "value": "1rWe7ALmsQJwwrtVcgVY6DtvWYLZZtmscL9fDpYFJLQo",
  "mode": "list",
  "cachedResultName": "Soporte Orrares",
  "cachedResultUri":
"https://docs.google.com/spreadsheets/d/1rWe7ALmsQJwwrtVcgVY6DtvWYLZZtmscL9fDpYFJLQo/edit?usp=drivesdk"
},
"sheetName": {
  "__rl": true,
  "value": "gid=0",
  "mode": "list",
  "cachedResultName": "Soporte",
  "cachedResultUri":
"https://docs.google.com/spreadsheets/d/1rWe7ALmsQJwwrtVcgVY6DtvWYLZZtmscL9fDpYFJLQo/edit#gid=0"
},
"columns": {
  "mappingMode": "defineBelow",
  "value": {
    "Fecha": "={{ $now }}",
    "Chat_ID": "={{ $('Respuesta al usuario').item.json.result.chat.id }}",
    "nombre ": "={{ $('Bot Orrares').item.json.output.nombre }}",
    "telefono": "={{ $('Bot Orrares').item.json.output.telefono }}",
    "clasificacion": "={{ $('Bot Orrares').item.json.output.clasificacion }}",
    "correo": "={{ $('Bot Orrares').item.json.output.correo }}",
    "resumen": "={{ $('Bot Orrares').item.json.output.resumen }}"
  },
  "matchingColumns": [
    "Chat_ID"
  ],
  "schema": [
    {
      "id": "Fecha",
      "displayName": "Fecha",
      "required": false,
      "defaultMatch": false,
      "display": true,
      "type": "string",
      "canBeUsedToMatch": true
    },
    {
      "id": "nombre ",
      "displayName": "nombre ",

```

```
"required": false,
"defaultMatch": false,
"display": true,
"type": "string",
"canBeUsedToMatch": true,
"removed": false
},
{
  "id": "Chat_ID",
  "displayName": "Chat_ID",
  "required": false,
  "defaultMatch": false,
  "display": true,
  "type": "string",
  "canBeUsedToMatch": true,
  "removed": false
},
{
  "id": "correo",
  "displayName": "correo",
  "required": false,
  "defaultMatch": false,
  "display": true,
  "type": "string",
  "canBeUsedToMatch": true,
  "removed": false
},
{
  "id": "telefono",
  "displayName": "telefono",
  "required": false,
  "defaultMatch": false,
  "display": true,
  "type": "string",
  "canBeUsedToMatch": true,
  "removed": false
},
{
  "id": "clasificacion",
  "displayName": "clasificacion",
  "required": false,
  "defaultMatch": false,
  "display": true,
  "type": "string",
  "canBeUsedToMatch": true,
  "removed": false
}
```

```

    },
    {
      "id": "resumen",
      "displayName": "resumen",
      "required": false,
      "defaultMatch": false,
      "display": true,
      "type": "string",
      "canBeUsedToMatch": true,
      "removed": false
    }
  ],
  "attemptToConvertTypes": false,
  "convertFieldsToString": false
},
"options": {}
},
"type": "n8n-nodes-base.googleSheets",
"typeVersion": 4.7,
"position": [
  256,
  -176
],
"id": "21270cb0-d783-4897-a36e-cbacdfd0453e",
"name": "Agrega google sheets",
"credentials": {
  "googleSheetsOAuth2Api": {
    "id": "3A6vvkmJxOMny06U",
    "name": "Google Sheets - orregol343"
  }
}
},
{
  "parameters": {
    "resource": "file",
    "fileId": "={{ $json.message.voice.file_id }}",
    "additionalFields": {}
  },
  "type": "n8n-nodes-base.telegram",
  "typeVersion": 1.2,
  "position": [
    -1440,
    -288
  ],
  "id": "3c5a35c8-986e-4fdb-8a82-69f7ac9d5698",
  "name": "Get a file",

```

```

"webhookId": "cdf810ac-20ca-464b-aeb0-bceefa365026",
"credentials": {
  "telegramApi": {
    "id": "7GfwHM2jl21XOFas",
    "name": "Soporte_S.A.S"
  }
},
{
  "parameters": {
    "resource": "audio",
    "modelId": {
      "__rl": true,
      "value": "models/gemini-2.5-flash",
      "mode": "list",
      "cachedResultName": "models/gemini-2.5-flash"
    },
    "inputType": "binary",
    "options": {}
  },
  "type": "@n8n/n8n-nodes-langchain.googleGemini",
  "typeVersion": 1,
  "position": [
    -1200,
    -288
  ],
  "id": "30199a97-b3c9-4c87-8a4f-e858e6e07323",
  "name": "Transcribe a recording",
  "credentials": {
    "googlePalmApi": {
      "id": "clrh3uFEmid1kX3F",
      "name": "api key orregol343 - grupo 1"
    }
  },
  "parameters": {
    "rules": {
      "values": [
        {
          "conditions": {
            "options": {
              "caseSensitive": true,
              "leftValue": "",
              "typeValidation": "strict",
              "version": 2
            }
          }
        }
      ]
    }
  }
}

```

```

    },
    "conditions": [
      {
        "leftValue": "={{ $json.message.voice.mime_type }}",
        "rightValue": "audio/ogg",
        "operator": {
          "type": "string",
          "operation": "equals"
        }
      },
      {
        "id": "a7f7e25c-8e02-4cce-a962-2daacc9e27ee"
      }
    ],
    "combinator": "and"
  },
  "renameOutput": true,
  "outputKey": "Audio"
},
{
  "conditions": {
    "options": {
      "caseSensitive": true,
      "leftValue": "",
      "typeValidation": "strict",
      "version": 2
    }
  },
  "conditions": [
    {
      "id": "a2e0762b-b645-4092-9244-b35d4b6b49c1",
      "leftValue": "={{ $json.message.chat }}",
      "rightValue": "={{ $json.message.text }}",
      "operator": {
        "type": "object",
        "operation": "exists",
        "singleValue": true
      }
    }
  ],
  "combinator": "and"
},
"renameOutput": true,
"outputKey": "Texto"
}
]
},
"options": {}
},

```

```

    "type": "n8n-nodes-base.switch",
    "typeVersion": 3.3,
    "position": [
      -1712,
      -80
    ],
    "id": "a23c46c9-8008-42e8-96bd-0edfe779414e",
    "name": "Switch"
  },
  {
    "parameters": {
      "assignments": {
        "assignments": [
          {
            "id": "2a9e6885-a474-4fc0-96a0-ebea12eb295a",
            "name": "mensaje",
            "value": "={{ $json.content.parts[0].text }} {{ $json.message.text }}",
            "type": "string"
          }
        ]
      },
      "options": {}
    },
    "type": "n8n-nodes-base.set",
    "typeVersion": 3.4,
    "position": [
      -1008,
      -80
    ],
    "id": "fa993056-c131-48a9-9e88-6509adcfadbb",
    "name": "Edit Fields"
  }
],
"pinData": {},
"connections": {
  "Google Gemini Chat Model": {
    "ai_languageModel": [
      [
        {
          "node": "Bot Orrares",
          "type": "ai_languageModel",
          "index": 0
        }
      ]
    ]
  }
},

```

```

    "If": {
      "main": [
        [
          {
            "node": "Agrega google sheets",
            "type": "main",
            "index": 0
          }
        ],
        [
          {
            "node": "Cerrar",
            "type": "main",
            "index": 0
          }
        ]
      ]
    },
    "Respuesta al usuario": {
      "main": [
        [
          {
            "node": "If",
            "type": "main",
            "index": 0
          }
        ]
      ]
    },
    "Contexto de conversacion": {
      "ai_memory": [
        [
          {
            "node": "Bot Orrares",
            "type": "ai_memory",
            "index": 0
          }
        ]
      ]
    },
    "Estructura del JSON": {
      "ai_outputParser": [
        [
          {
            "node": "Bot Orrares",
            "type": "ai_outputParser",

```

```
    "index": 0
  }
]
],
},
"Mensaje a recibir": {
  "main": [
    [
      {
        "node": "Switch",
        "type": "main",
        "index": 0
      }
    ]
  ]
},
"Bot Orrares": {
  "main": [
    [
      {
        "node": "Respuesta al usuario",
        "type": "main",
        "index": 0
      }
    ]
  ]
},
"Get a file": {
  "main": [
    [
      {
        "node": "Transcribe a recording",
        "type": "main",
        "index": 0
      }
    ]
  ]
},
"Transcribe a recording": {
  "main": [
    [
      {
        "node": "Edit Fields",
        "type": "main",
        "index": 0
      }
    ]
  ]
}
```

```

    ]
  ],
  "Switch": {
    "main": [
      [
        {
          "node": "Get a file",
          "type": "main",
          "index": 0
        }
      ],
      [
        {
          "node": "Edit Fields",
          "type": "main",
          "index": 0
        }
      ]
    ]
  },
  "Edit Fields": {
    "main": [
      [
        {
          "node": "Bot Orrares",
          "type": "main",
          "index": 0
        }
      ]
    ]
  },
  "active": false,
  "settings": {
    "executionOrder": "v1"
  },
  "versionId": "8fbbe677-6513-4c60-9fc9-18ab187b5b91",
  "meta": {
    "templateCredsSetupCompleted": true,
    "instanceId":
"0d262c77ddf5e98146562ac998010e66121b3ae266f215c22f6bf79557362581"
  },
  "id": "9z2LFa84yF5shkhW",
  "tags": []
}

```

