

Leonel Gordon

Informe Agente de ventas

Flujo de interacción del Agente

1. Usuario inicia conversación.
2. El agente saluda y solicita qué producto busca.
3. El usuario describe su necesidad.
4. El agente detecta intención:

Si NO es compra

→ responde brevemente o deriva a humano (Chatwoot).

Si SÍ es compra:

5. El agente consulta productos (`list_products`) y muestra resultados (opcional: filtra por rango de precio). En caso de no encontrar productos, solicita más información y reintenta. Ante fallas persistentes o error MCP, notifica al usuario y deriva a humano.
6. El usuario selecciona un producto o pide detalles.
7. El agente muestra información del producto.
8. El usuario confirma producto y cantidad.
9. El agente crea carrito (`create_cart`) y agrega el ítem si se excede la cantidad del stock se notifica al usuario para cambiar cantidad y da opción de derivar a humano.
10. El agente pregunta si desea agregar más productos.

Loop:

- Si el usuario continúa → vuelve al paso 5
 - Si finaliza → muestra resumen del carrito
11. El usuario puede editar cantidades o eliminar productos (`update_cart`).
 12. El agente confirma carrito final.
 13. En cualquier momento, si el usuario lo solicita o hay reclamo → derivación humana en Chatwoot (`handoff_to_human`).

MCP Endpoints

1. **list_products**

Lista productos desde D1 con filtros opcionales.

Entradas: query (opcional), terms (opcional), min_price (opcional), max_price (opcional), limit (opcional).

Incluye normalización (sin tildes / mayúsculas) y expansión básica (plural/singular).

Devuelve: array de productos { id, name, description, price, stock }.

2. **create_cart**

Crea (o devuelve) un carrito único por conversation_id.

Entrada: conversation_id (requerido).

Devuelve: { id, conversation_id, created_at, updated_at }.

3. **update_cart**

Agrega/actualiza/elimina un ítem del carrito.

Entrada: conversation_id, product_id, qty (requeridos).

Regla: qty <= 0 elimina el ítem; qty > 0 hace upsert.

Devuelve: { cart_id, product_id, qty }.

4. **get_cart**

Devuelve el carrito actual con sus ítems.

Entrada: conversation_id (requerido).

Devuelve: { cart_id, items: [{ id, name, price, qty }] }.

5. **handoff_to_human**

Deriva a un humano vía Chatwoot API.

Entradas: conversation_id, reason, context (requeridos), labels (opcional).

Acciones:

- Parsea conversation_id (acepta numérico o “decorado” tipo chatwoot_..._118_87_59 extrayendo el ID real).
- Aplica label “handoff”.
- Crea nota privada con contexto.
- Cambia el estado de la conversación a “open”.

Observabilidad / logs

Cada tool registra: inicio, parámetros clave (sin exponer contenido sensible), resultado (status/contador) y errores.

Debug en tiempo real: **npx wrangler tail**