



# Desafío Técnico · AI Engineer

## para Laburen.com

---



### Descripción General

Diseña y demuestra, de punta a punta, cómo un agente de IA puede vender productos mediante un **MCP (Model Context Protocol)** propio y una base de datos. El reto se divide en una **fase conceptual** (soft) y una **fase práctica** (técnica). Todo el material debe ser 100% ejecutable.

**⚠ IMPORTANTE:** No se busca un BOT (serie de menús en un chat), se espera un agente de IA capaz de ejecutar solicitudes HTTP que mantenga una interacción con el usuario coherente.



### Paso 1: Configuración Inicial

1. Crear una cuenta en <https://dashboard.laburen.com/>
2. Usar el código **challengexagents** para habilitar la cuenta de \$20 USD ingresando a la pasarela de pago
3. Recibirás por email las credenciales de acceso a una instancia de **Chatwoot** (usuario y contraseña)
4. Configura la conexión de Chatwoot para tu agente en Laburen con las siguientes credenciales:

#### Credenciales de Chatwoot

*Copia y pega estos datos en el formulario de configuración de Chatwoot en Laburen:*

**Integration Name\*:**

[Tu nombre]

Clic para copiar

**Chatwoot Server URL\*:**

<https://chatwootchallenge.laburen.com/>

Clic para copiar

**Platform App Token\*:**

7YRVVb8JzvDG8yfaVkJAQ48j

Clic para copiar

**New Chatwoot Integration**

**Step 1**  
Configure administrative access

**Step 2**  
Configure user token and select account

**Step 3**  
Create intelligent bot

**Connect Platform App (Administrator)**

**Integration Name \***  
tu nombre

How the agent's name will appear in the chat

**Chatwoot Server URL \***  
<https://chatwootchallenge.laburen.com/>

**Platform App Token \***  
7YRVVb8JzvDG8yfaVkJAQ48j

Solo para instancias self-hosted con acceso de Super Admin

**Test connection**

**Continue**

Vista del formulario de configuración de Chatwoot en Laburen

 **Fase Conceptual · Diseño del Agente de IA**

**Mapa de flujo**

Ilustra (diagrama de flujo o secuencia) cómo el agente atiende a un cliente que:

1. Explora productos
2. Crea un carrito
3. (Extra) Edita el carrito si el usuario lo pide

**Formato de entrega:** PDF o Markdown de máx. 2 páginas con los endpoints + diagrama de flujo de interacción del agente.



## Fase Práctica • MCP & Base de Datos

### 2.1 Fuente de datos

Se proveerá un archivo `products.xlsx` con N filas. Cada fila representa un producto.



[Descargar products.xlsx](#)

### 2.2 Base de datos

Crea el esquema mínimo siguiente (puedes ampliarlo):

Tabla	Campos clave	Notas
products	<code>id</code> (PK), <code>name</code> , <code>description</code> , <code>price</code> , <code>stock</code>	-
carts	<code>id</code> (PK), <code>created_at</code> , <code>updated_at</code>	Un carrito por conversación.
cart_items	<code>id</code> (PK), <code>cart_id</code> (FK), <code>product_id</code> (FK), <code>qty</code>	-

### 2.3 Capacidades del Agente de IA

Capacidad	Descripción
<b>Explorar productos</b>	El agente puede buscar y mostrar productos al usuario, utilizando el MCP para obtener información de productos con filtros opcionales por nombre o descripción.
<b>Mostrar detalles de productos</b>	El agente puede mostrar información detallada de un producto específico al usuario.
<b>Crear carrito de compras</b>	El agente puede crear un carrito y añadir productos cuando el usuario expresa intención de compra. El agente tiene que poder ver todos los ítems agregados al carrito. El agente debe agregar etiquetas en el CRM para identificar los productos que el usuario ha agregado al carrito.
<b>Editar carrito (Extra)</b>	El agente puede actualizar cantidades o eliminar ítems del carrito si el usuario lo solicita.
<b>Derivar conversación a humano</b>	El agente puede abrir una conversación en el CRM (Chatwoot) para derivar al usuario a un agente humano. El agente debe agregar etiquetas apropiadas cuando realiza la derivación, indicando el contexto de la conversación y el motivo de la derivación.

## 2.4 Requisitos de Despliegue

- El **MCP debe estar desplegado en Cloudflare** (<https://dash.cloudflare.com/>)
- Puedes usar la instancia **gratuita de Cloudflare**
- La base de datos puede estar desplegada en Cloudflare o en otro servicio

 Documentación de Cloudflare: <https://developers.cloudflare.com/>



## Fase Práctica · Integración del Agente

1. **Desarrollo en la plataforma de Laburen:** El agente se desarrollará en la plataforma de Laburen (<https://dashboard.laburen.com/>), donde podrás:

- Conectar tu MCP
- Probar diferentes modelos de LLM
- Conectarlo directamente a la instancia de Chatwoot de Laburen (CRM Laburen)

2. El agente debe:

- Mostrar productos (consume función MCP `list_products`)
- Crear un carrito (consume función MCP `create_cart`) al recibir intención de compra
- (Extra) Editar un carrito (consume función MCP `update_cart`)

3. **Interface:** El agente debe estar desplegado en **WhatsApp** y conectado mediante la instancia de **Chatwoot** de Laburen (CRM Laburen).



## Entregables

Nº	Elemento	Forma
1	Agente desplegado y listo para responder	Chatwoot (número común)
2	Repositorio GitHub	Código del MCP
3	Diagrama(s) & documento conceptual	Carpeta <code>/docs</code>



## Criterios de Evaluación

Peso	Competencia	Detalle
55%	Integracion del Agente	Se busca mantener una charla continua y COHERENTE, consumo del MCP e integración con el CRM
40%	Backend & MCP	Calidad del modelo de datos, manejo de errores
5%	Presentación & docu.	Orden, brevedad, facilidad para correr la prueba



## Entrega

A partir del día siguiente a la entrega del challenge comienza la cuenta de **5 días** para la entrega del challenge.

Se espera poder hacer el test del agente directamente por Chatwoot en vivo y ver el consumo del MCP.

---

**¡Éxitos!**