

CenterMind — Instructivo: Alta de Nueva Distribuidora

Seguí este documento de principio a fin cada vez que se incorpore una distribuidora nueva al sistema.
Tiempo estimado: 20-30 minutos la primera vez, 10 minutos las siguientes.

Resumen de pasos

#	Tarea	Dónde	Tiempo
1	Crear bot de Telegram	@BotFather	3 min
2	Habilitar API de Google Drive	Google Cloud Console	5 min
3	Crear credencial OAuth2	Google Cloud Console	5 min
4	Autorizar acceso a Drive	Terminal del servidor	2 min
5	Crear carpeta en Google Drive	Google Drive	2 min
6	Registrar distribuidora en la DB	DB Browser	2 min
7	Verificar que el bot levanta	Terminal	2 min

PASO 1 — Crear bot de Telegram

1. Abrí Telegram y buscá **@BotFather**
2. Mandá /newbot
3. Ingresá el **nombre visible** del bot (ej: Distribuidora Norte Exhibiciones)
4. Ingresá el **username** del bot — debe terminar en `bot`
5. BotFather te devuelve el **token** — se ve así:

7123456789 : AAFxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

6. **Guardá ese token**, lo vas a necesitar en el Paso 6.

■■■ El token es como una contraseña. No lo compartás ni lo subas a GitHub.

PASO 2 — Habilitar API de Google Drive

1. Ir a console.cloud.google.com
2. Seleccioná (o creá) el proyecto **Bot-SQL**
3. En el menú: **APIs y servicios** → **Biblioteca**
4. Buscá **Google Drive API** → Habilitar

PASO 3 — Crear credencial OAuth2

■ ACTUALIZACIÓN: El tipo correcto es **APLICACIÓN DE ESCRITORIO**, no aplicación web. Además, debés agregar tu email como usuario de prueba antes de poder autorizar.

1. En Google Cloud Console: **APIs y servicios** → **Credenciales**

2. Clic en **+ Crear credencial** → **ID de cliente OAuth 2.0**
3. Tipo de aplicación: **Aplicación de escritorio** (Desktop app) — NO elegir 'Aplicación web'
4. Nombre: cualquiera (ej: CenterMind-Desktop)
5. Clic en **Crear** → **Descargar JSON**
6. Renombrarlo a `credencial_oauth.json` y copiarlo a `CenterMind/`

Agregar usuario de prueba (obligatorio)

■■ Sin este paso, Google bloquea el acceso con Error 403: access_denied.

1. En Google Cloud Console: **APIs y servicios** → **Pantalla de consentimiento de OAuth**
2. Bajá hasta la sección **Usuarios de prueba**
3. Clic en **+ Add users**
4. Ingresá el email de la cuenta Google dueña del Drive
5. Guardar

Este archivo contiene el `client_id` y `client_secret`. No subir a GitHub.

PASO 4 — Autorizar acceso a Drive (una vez por servidor)

Genera el `token_drive.json` que el bot usa para siempre. Solo necesitás hacerlo una vez. El token se renueva solo automáticamente.

■ FIX SSL — Windows con PostgreSQL instalado: PostgreSQL sobreescribe la variable `SSL_CERT_FILE` del sistema. El script `setup_drive_oauth.py` ya tiene el fix incorporado. Si en un equipo nuevo falla con 'Could not find a suitable TLS CA certificate bundle', abrir PowerShell como Administrador y ejecutar: `[System.Environment]::SetEnvironmentVariable("SSL_CERT_FILE", $null, "Machine")` Luego abrir una consola nueva y correr el setup.

1. Abrí PowerShell en la carpeta `CenterMind/`
2. Ejecutá:

```
cd C:\Users\cigar\OneDrive\Desktop\BOT-SQL\CenterMind
python setup_drive_oauth.py
```
3. Se abrirá el navegador con una pantalla de Google
4. Iniciá sesión con la cuenta de Google dueña del Drive
5. Hacé clic en **Permitir** (si muestra 'app no verificada' → 'Avanzado' → 'Ir a CenterMind')
6. El script mostrará confirmación con nombre, email y espacio disponible
7. Verificá que se creó el archivo `CenterMind/token_drive.json`

¿Qué pasa si el token expira? Los tokens OAuth2 no tienen vencimiento si se usan cada 6 meses. El bot los renueva automáticamente. Si se invalida, volvé a correr `setup_drive_oauth.py`.

PASO 5 — Crear carpeta en Google Drive

1. Abrí Google Drive con la misma cuenta que usaste en el Paso 4

2. Navegá a Mi unidad > CenterMind_Fotos
3. Creá una subcarpeta con el nombre de la distribuidora
4. Entrá a esa carpeta → copiá el **ID de la carpeta** desde la URL (la parte larga al final)
5. Guardá este ID, lo vas a necesitar en el Paso 6.

PASO 6 — Registrar distribuidora en la DB

1. Abrí **DB Browser for SQLite**
2. Abrí base_datos/centermind.db
3. Ir a la pestaña **Ejecutar SQL**
4. Ejecutá el siguiente INSERT reemplazando los valores:

```
INSERT INTO distribuidores  
  
(nombre_empresa, token_bot, id_carpetas_drive, estado)  
  
VALUES  
  
( 'Nombre Empresa' ,  
  
'1234567890:AAFxxxxxxxxxxxxxxxxxxxxxx' ,  
  
'1BxiMVs0XRA5nFMdKvBdBZjgmUUqptlbs' ,  
  
'activo' );
```

5. Clic en **Ejecutar** → **Guardar cambios** → **Cerrar DB Browser**

■■ Cerrá DB Browser antes de levantar el bot, de lo contrario obtendrás 'database is locked'.

PASO 7 — Verificar que el bot levanta

1. **Cerrá DB Browser** (no dejar abierto mientras corre el bot)
2. Abrí PowerShell:

```
cd C:\Users\cigar\OneDrive\Desktop\BOT-SQL\CenterMind  
  
# Obtener el id_distribuidor asignado:  
  
python -c "import sqlite3; c=sqlite3.connect('base_datos/centermind.db');  
print(list(c.execute('SELECT id_distribuidor, nombre_empresa FROM  
distribuidores')))"  
  
python bot_worker.py --distribuidor-id [ID_ASIGNADO]
```

3. Deberías ver en la consola:

- DriveUploader conectado con OAuth2
- BotWorker listo para: Distribuidora Norte
- Iniciando polling: Distribuidora Norte

4. Si el bot levanta correctamente, pará con `Ctrl+C`
 5. En producción, el orquestador lo levantará automáticamente con `python centermind_core.py`
-

PASO 8 — Configurar el bot en Telegram (una vez)

1. Buscá el bot por su username en Telegram
 2. Mandá `/start` — debería responder
 3. Agregá el bot al grupo de Telegram de esa distribuidora
 4. **Deshabilitar privacy mode** para que el bot reciba fotos:
 - Abrí `@BotFather` → `/mybots` → seleccioná el bot
 - Bot Settings → Group Privacy → **Turn off**
 - Sacá el bot del grupo y volvelo a agregar
-

Checklist rápido (para las siguientes distribuidoras)

Una vez que ya tenés el proyecto configurado (Pasos 2, 3, 4 ya hechos), cada nueva distribuidora solo requiere:

- Crear bot en `@BotFather` → guardar token
- Crear carpeta en Google Drive → copiar ID
- INSERT en distribuidores con token + folder ID
- Reiniciar `centermind_core.py` (o esperar 60s)
- Agregar el bot al grupo → desactivar privacy mode → probar con `/help`

Tiempo total: ~10 minutos.