

# Guía para Crear un Chatbot de Ausencias con Telegram, Google Sheets y Vision API

Este documento sirve como recordatorio y tutorial paso a paso para configurar un chatbot en Telegram que registre ausencias, procese certificados médicos mediante OCR y guarde la información en Google Sheets y Drive. Además, explica cómo habilitar APIs, generar credenciales y resolver problemas comunes que surgieron durante el desarrollo.

## 1. Crear un Bot de Prueba en Telegram

- Abrir Telegram y buscar el bot oficial @BotFather.
- Ejecutar el comando /newbot para crear un nuevo bot.
- Asignar un nombre y un 'username' único terminado en 'bot'.
- BotFather entregará un TOKEN. Guardar este token en el archivo .env como TELEGRAM\_TOKEN.
- Este token permitirá que tu código Python se conecte con tu bot.

## 2. Habilitar y Configurar APIs de Google

El proyecto requiere las siguientes APIs de Google Cloud: - Google Sheets API → Para leer y escribir registros de ausencias. - Google Drive API → Para subir certificados médicos escaneados. - Google Vision API → Para realizar OCR y extraer texto de imágenes o PDFs.

- Ingresar a <https://console.cloud.google.com/>
- Crear un proyecto nuevo o usar uno existente.
- Ir a 'API & Services > Library' y habilitar Google Sheets API, Google Drive API y Google Vision API.
- Ir a 'API & Services > Credentials' y crear una credencial de tipo 'Service Account'.
- Generar una clave en formato JSON y descargarla. Este archivo se usará en el bot.
- Compartir la hoja de Google Sheets con el correo de la cuenta de servicio para que tenga acceso.

## 3. Habilitación de Facturación

Para que la API de Google Vision funcione, es necesario habilitar la facturación en Google Cloud. Google ofrece un crédito gratuito inicial. Pasos: - Ir a 'Billing' en Google Cloud Console. - Asociar un método de pago válido (tarjeta de crédito o débito). - Vincular el proyecto a la cuenta de facturación.

## 4. Archivo .env

El archivo .env debe contener variables sensibles como tokens y credenciales. Ejemplo:  
TELEGRAM\_TOKEN=tu\_token\_aqui GS\_SERVICE\_ACCOUNT\_FILE=credenciales.json  
GS\_SHEET\_ID=ID\_de\_tu\_Google\_Sheet GS\_SHEET\_TAB\_NAME=Registros

## 5. Ejecutar el Bot

- Instalar dependencias: `pip install -r requirements.txt`
- Ejecutar el script principal: `python main_ocr_unificado.py`
- Enviar un mensaje al bot en Telegram con DNI o legajo para registrar ausencia.
- Subir la foto del certificado cuando se solicite.

## 6. Problemas y Soluciones Frecuentes

- Error con credenciales JSON → Revisar que el archivo exista y tenga permisos.
- El bot no escribe en Google Sheets → Compartir la hoja con el email de la cuenta de servicio.
- Fecha mal reconocida en OCR → Se solucionó formateando con `datetime` y `regex`.
- Avisos fuera de horario → Ahora se registra en 'Observaciones' si es después de las 10hs.
- Certificados subidos fuera de las 24hs → Se registra en 'Observaciones' con advertencia.
- DNI no coincide → Se agrega en 'Observaciones' que no coincide con el aviso.

## 7. Tips para Mejorar y Escalar

- Usar un entorno virtual (`venv`) para aislar dependencias.
- Mantener actualizado `requirements.txt` y `README.md`.
- Controlar logs del bot para detectar errores tempranos.
- Probar con diferentes certificados (PDF, fotos) para validar robustez.
- Si el OCR falla, usar preprocesamiento (contraste, escala de grises).
- Separar la lógica de negocio en módulos para mantenimiento futuro.