

## # Transcripción y Análisis de Videos de YouTube 🎬📝

![Python](https://img.shields.io/badge/Python-3.8+-blue.svg)

![Flask](https://img.shields.io/badge/Flask-2.0+-green.svg)

![Licencia MIT](https://img.shields.io/badge/Licencia-MIT-yellow.svg))(LICENSE)

Una aplicación web moderna para transcribir, analizar y extraer insights de videos de YouTube utilizando IA.

## ## ✨ Características Principales

- **Transcripción Automática** de videos con alta precisión
- **Análisis con Gemini** para resúmenes y extracción de insights clave
- **Formatos de Exportación**:
  - 📄 Markdown (para documentación)
  - 📊 JSON (para integración con otras apps)
  - 📄 TXT (para uso rápido)
- **Interfaz Intuitiva** con diseño responsive
- **Procesamiento por Lotes** (varios videos simultáneamente)

## ## 🚀 Requisitos del Sistema

Componente	Versión	Notas
Python	3.8+	Probado en 3.10
Flask	2.0+	Framework web
Google API Client	1.12+	Para YouTube API
Gemini API	-	Clave requerida

## ## 🛠️ Instalación Rápida

1. Clonar repositorio:

```
``bash
git clone https://github.com/tuusuario/TranscripcionYoutube.git
cd TranscripcionYoutube
``
```

2. Configurar entorno virtual:

```
``bash
python -m venv venv
source venv/bin/activate # Linux/Mac
venv\Scripts\activate   # Windows
``
```

3. Instalar dependencias:

```
```bash
pip install -r requirements.txt
```
```

## ## Configuración

1. Obtener claves API:

- [YouTube Data API v3](https://console.cloud.google.com/)
- [Gemini API](https://ai.google.dev/)

2. Configurar claves:

```
```python
# config.py
YOUTUBE_API_KEY = "tu_clave_youtube"
GEMINI_API_KEY = "tu_clave_gemini"
```
```

## ## Uso Básico

Iniciar servidor:

```
```bash
flask run --debug
```
```

Acceder en tu navegador:

```
...
http://localhost:5000
...
```

#### Ejemplo de análisis:

1. Pegar URL de YouTube
2. Seleccionar opciones:
  - Idioma preferido
  - Tipo de análisis (básico/avanzado)
3. Descargar resultados

## ## Estructura del Proyecto

...

TranscripcionYoutube/

```
|— app/           # Módulo principal
| |— __init__.py
| |— routes.py    # Endpoints Flask
```

```

|   |   | services/      # Lógica de negocio
|   |   | |   |
|   |   | |   | youtube.py # Integración YouTube API
|   |   | |   | gemini.py  # Procesamiento con IA
|   |   | |   |
|   |   | tests/        # Pruebas unitarias
|   |   | static/       # Assets frontend
|   |   | |   |
|   |   | |   | css/
|   |   | |   | |   |
|   |   | |   | |   | js/
|   |   | |   | |   |
|   |   | templates/   # Vistas HTML
|   |   | |   |
|   |   | |   | base.html
|   |   | |   | index.html
|   |   |
|   |   ...

```

## ## 🌐 API Reference

### ### Endpoints disponibles

| Método | Ruta                   | Descripción          |
|--------|------------------------|----------------------|
| GET    | /                      | Interfaz principal   |
| POST   | /api/analyze           | Procesar video       |
| GET    | /api/download/<format> | Descargar resultados |

## ## 🐛 Solución de Problemas

**\*\*Error común\*\*:** "Invalid API Key"

```
```bash
```

# Verificar que las claves estén en:

1. Variables de entorno O
2. Archivo config.py
3. Que no tengan espacios

```
```
```

**\*\*Error\*\*:** "Video unavailable"

```
```bash
```

- Verificar que el video no sea privado
- Comprobar restricciones regionales

```
```
```