# Heratropic API

Procesamiento de video y audio para cuantificación y monitorización de estados emocionales.

URL: <a href="https://heratropic-main-c6ba0ae.d2.zuplo.dev">https://heratropic-main-c6ba0ae.d2.zuplo.dev</a>

API\_KEY: zpka\_0f68e5569e324ee59d1e192c609ade0a\_0a42c62e (Antonio)

API\_KEY: Zpka\_ed56a7576f47465095a2f3ee1d08c2ca\_56c05972 (Enrique)

## 1. Subir un archivo (≤ 5MB)

Para subir un archivo de hasta 5MB, usa el siguiente comando:

curl -H "Authorization: Bearer zpka\_0f68e5569e324ee59d1e192c609ade0a\_0a42c62e" -X POST https://heratropic-main-c6ba0ae.d2.zuplo.dev/v1/upload -F "file=@//Users/dmartincc/iridis/pythagoras/trump.mp4" -F external\_vars='{"id":"1"}'

{"status":"success","response":{"created\_at":1741878658,"aid":"2413354b-4d8b-4fee-a1a8-8e71d9095bda","result\_url":"/v1/result/2413354b-4d8b-4fee-a1a8-8e71d9095bda","upload\_url":null, "original\_file":{"extension":".mp4","format":"video"},"status":{"FILE\_STORED":false,"FACIAL\_AN ALYSED":false,"VOICE\_ANALYSED":false,"VOICE\_TRANSCRIBED":false,"BIOMETRICS\_EXT RACTED":false,"SPEECH\_ANALYSED":false,"PERSONALITY\_ANALYSED":false},"external\_vars":{"id":"1"}}}

#### Parámetros:

- file: El archivo que deseas subir.
- external\_vars: JSON con metadatos adicionales, como el ID y el canal de origen.
- Authorization: Token de autenticación requerido para acceder a la API.

## 2. Subir un archivo grande (> 5MB)

Para archivos mayores a 5MB, usa el siguiente comando:

curl -X POST "https://heratropic-main-c6ba0ae.d2.zuplo.dev/v1/upload/large" \

- -H "Content-Type: multipart/form-data" \
- -H "Authorization: Bearer zpka 0f68e5569e324ee59d1e192c609ade0a 0a42c62e" \
- -F "filename=recording.mp4" \

-F "external\_vars={}"

{"status":"success", "response": {"created at":1741878963, "aid": "33e2c92f-22b9-4957-ac0a-8dd1 5094b751","result\_url":"/v1/result/33e2c92f-22b9-4957-ac0a-8dd15094b751","upload\_url":"htt ps://s3.eu-central-1.amazonaws.com/dev.datum.eu-central-1.stresstech/zpka\_0f68e5569e 324ee59d1e192c609ade0a 0a42c62e/33e2c92f-22b9-4957-ac0a-8dd15094b751/video.mp4 ?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=ASIAUSDIXVQBQ2LILTJJ% 2F20250313%2Feu-central-1%2Fs3%2Faws4\_request&X-Amz-Date=20250313T151603Z&X -Amz-Expires=3600&X-Amz-SignedHeaders=content-type%3Bhost&X-Amz-Security-Toke n=IQoJb3JpZ2luX2VjEl%2F%2F%2F%2F%2F%2F%2F%2F%2F%2F%EaDGV1LWNlbn RyYWwtMSJGMEQCIC4LReSVaCJZDDgTuH6%2BfmYYiOcw%2F2OWHXc4VK4DSWXNAi BQro9r8a1skKUALiUF3RQQXppMAH8mMdOty5XijCopdCqlAwjY%2F%2F%2F%2F%2F%2 F%2F%2F%2F%2F8BEAEaDDMxMzc1Mjl2Nzc3OSIMsnazbSacC884Qf%2F0KtwC73FQOG 1oQriaWR3WsDX9LI%2F%2FJ0Bk8ywERVkZIEuIVrkl1MmJAqhZuz7AHqaJ8uA7DKGYDcn 2yMAqe1aoTknPGTxBNJo7OKrxBNC58P3%2B4KKt3%2FLeqBDuUdnWMhJr%2FTUhGzP xGVPiEdt8ExuKzWgB8mG9JV6RYxHVNz74v06%2Ffi69wgFaF8fNfxBtuCWXYUmvCsnLPe uMignG67%2F%2F19Ceh%2FHd8hEre2ZMRvhllVxpkpM3CzoyCYnF6rVrVw4TMgnwTYukO UtW7knBHr0z4EqJZQp1O6MhUL0MVpTr8uqYOp%2FrmHTm9xy9sztb2h5gmgQBZ7v1kEy uLLeUle9SZE3oJ9txkUgi8dk7I7Sto7H%2B3T5cQU6I36hECFWUCLMPETf5BjVqQAWqwfT GMo2ExgGDoaarF49dwXi7LTsNH%2BwFmuo0bN%2B9xiIDc5ic1YpMJiQiArFC9YawlJeue E%2BrMO%2Fny74GOp8BEjUjClco%2B9oKTcu%2FLU4lPkfZpqx0PB661OO3zeO1DJnBG T6ti3YNZNAjoxzkf1f81IL55FVGw3a81vj%2BAn4M79N8Orscn4AL6Q7C5XY4sl%2F52xFmqj rtlmDIUYyx6HmCSIXnnt73cNohFuLAG1FYW03jJl3ou25Din4HjogWmFVNXhyQAEyTTeKT alPyTc5LggKKZzwjDQfQfQsmfr3D&X-Amz-Signature=82d7ff3b1ddc2b5e5fc77f80cd84bb4 d4b1456f5a4ef615b3ea9f09c26628dca", "original file": {"extension": ".mp4", "format": "video"}, "st atus":{"FILE\_STORED":false,"FACIAL\_ANALYSED":false,"VOICE\_ANALYSED":false,"VOICE\_ TRANSCRIBED":false,"BIOMETRICS EXTRACTED":false,"SPEECH ANALYSED":false,"PER SONALITY\_ANALYSED":false\},"external\_vars":\{\}\}\%

Devuelve un url de subida, a la que hay que hacer un post subiendo el fichero. Ejemplo en python más abajo.

Negrita url a la que hay que volver a hacer un post para subir el fichero, como esta

### curl -X PUT

"https://s3.eu-central-1.amazonaws.com/dev.datum.eu-central-1.stresstech/zpka\_a/b2e5f5d0-df 07-4122-b97f-585c911caac8/video.mp4?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Cre dential=AKIAUSDIXVQBQIJTXRP6%2F20250221%2Feu-central-1%2Fs3%2Faws4\_request&X-Amz-Date=20250221T093529Z&X-Amz-Expires=3600&X-Amz-SignedHeaders=content-type% 3Bhost&X-Amz-Signature=e159e0f447c7e6cd754349e497e0e37bb8518235ffa213a629f14bcdb 9cd845f" \

- -H "Content-Type: video/mp4" \
- --upload-file /Users/dmartincc/iridis/oraculum/test\_david.mp4

#### Diferencias con la subida estándar:

- Se utiliza una URL diferente: /file/upload/large.
- Importante: Verifica la ruta del archivo en tu sistema antes de ejecutar el comando.

# 3. Obtener resultados

Para consultar el estado o resultado de un archivo procesado, utiliza:

```
curl -X 'GET' \
  'https://your-api-url/v1/result/{id}' \
  -H 'Authorization: Bearer YOUR_API_KEY'
```

### Parámetros:

- id: Identificador del archivo previamente subido.
- Authorization: Token necesario para la autenticación.

# **Notas adicionales**

- **Seguridad**: Asegúrate de mantener seguro tu API\_KEY y no compartirlo públicamente.
- **Errores**: Revisa la respuesta de la API en caso de errores y ajusta los parámetros según sea necesario.

Para más información, consulta la documentación oficial o contacta al equipo de soporte.

Ejemplo de subida de fichero mayor de 5MB con python:

```
import requests
import json
API_KEY = "your_api_key"
URL = "https://your-api-url.com"
def send_video(video_path, set_analysis_states, set_analysis_data):
  try:
    # Step 1: Request a presigned URL from the API
    form data = {
       "filename": "recording.mp4",
       "external vars": json.dumps({})
    headers = {
       "Authorization": f"Bearer {API KEY}"
     presign response = requests.post(f"{URL}/v1/upload/large", data=form data,
headers=headers)
     if presign_response.status_code != 200:
       print("Failed to get presigned URL.")
       return None
     presigned_url = presign_response.json().get("response", {}).get("upload_url")
     result_url = presign_response.json().get("response", {}).get("result_url")
    if not presigned url or not result url:
       print("Invalid response from server.")
       return None
     print("Presigned URL received:", presigned_url)
    # Step 2: Upload the video file to S3 using the presigned URL
     with open(video_path, "rb") as video_file:
```

```
s3_upload_response = requests.put(presigned_url, data=video_file, headers={"Content-Type": "video/mp4"})

if s3_upload_response.status_code != 200:
    print("Failed to upload video to S3.")
    return None

print("Video uploaded successfully to S3.")

except Exception as error:
    print("Error during video upload:", error)
    return None
```