

Heratropic API

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

URL: <https://heratropic-main-c6ba0ae.d2.zuplo.dev>

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-a1a8-8e7
1d9095bda\",\"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_va
rs\":{\"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-8dd15094b751",
    "result_url": "/v1/result/33e2c92f-22b9-4957-ac0a-8dd15094b751",
    "upload_url": "https://s3.eu-central-1.amazonaws.com/dev.datum.eu-central-1.stresstech/zpka_0f68e5569e324ee59d1e192c609ade0a_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-Token=IQoJb3JpZ2luX2VjEi%2F%2F%2F%2F%2F%2F%2F%2F%2F%2F%2FwEaDGV1LWNlbnRyYWwtMSJGMEQCIC4LRsVaCJZDDgTuH6%2BfmYYiOcw%2F2OWHXc4VK4DSWXNAIBQro9r8a1skKUALiUF3RQQXppMAH8mMdOty5XijCopdCqIawjY%2F%2F%2F%2F%2F%2F%2F%2F%2F8BEAEaDDMxMzc1Mjl2Nzc3OSIMsnazbSacC884Qf%2F0KtwC73FQOG1oQriaWR3WsDX9LI%2F%2FJ0Bk8ywERVkZIEuIVrk11MmJAqhZuz7AHgaJ8uA7DKGYDcn2yMAqe1aoTknPGTxBNJo7OKrxBNC58P3%2B4KKt3%2FLeqBDuUdnWMhJr%2FTUhGzPxGVPIedt8ExuKzWgB8mG9JV6RYxHVNz74v06%2Ffi69wgFaF8fnfxBtuCWXYUmvcSnLPeuMiqnG67%2F%2F19Ceh%2FHd8hEre2ZMRvhIIvxpkPM3CzoyCYnF6rVrVw4TMgnwTYukOUtW7knBHr0z4EqJZQp1O6MhUL0MVpTr8uqYOOp%2FrmHTm9xy9sztb2h5gmgQBZ7v1KEyuLLLeUle9SZE3oJ9txkUgi8dk7I7Sto7H%2B3T5cQU6l36hECFWUCLMPETf5BjVgQAWgwftGMo2ExggDOaarF49dwXi7LTsNH%2BwFmuo0bN%2B9xilDc5ic1YpMJiQiArFC9YawlJeueE%2BrMO%2Fn74GOp8BEjUjClco%2B9okTcu%2FLU4IPkfZpqx0PB661OO3zeO1DJnBGt6ti3YNZNAjoxzkf1f81IL55FVGw3a81vj%2Ban4M79N8Orscn4AL6Q7C5XY4sl%2F52xFmqjrtlmDIUYyx6HmCSIXnnt73cNohFuLAG1FYW03jJI3ou25Din4HjogWmFVNxhyQAeyTTektalPyTc5LggKKZzwjDQfQfQsmfr3D&X-Amz-Signature=82d7ff3b1ddc2b5e5fc77f80cd84bb4d4b1456f5a4ef615b3ea9f09c26628dca",
    "original_file": {
      "extension": ".mp4",
      "format": "video"
    },
    "status": {
      "FILE_STORED": false,
      "FACIAL_ANALYSED": false,
      "VOICE_ANALYSED": false,
      "VOICE_TRANSCRIBED": false,
      "BIOMETRICS_EXTRACTED": false,
      "SPEECH_ANALYSED": false,
      "PERSONALITY_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-df07-4122-b97f-585c911caac8/video.mp4?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAUSDIXVBQIJTXRP6%2F20250221%2Feu-central-1%2Fs3%2Faws4_request&X-Amz-Date=20250221T093529Z&X-Amz-Expires=3600&X-Amz-SignedHeaders=content-type%3Bhost&X-Amz-Signature=e159e0f447c7e6cd754349e497e0e37bb8518235ffa213a629f14bcdb9cd845f" \

```
-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
```