Straico - API

The **Straico API** utilizes **API keys** for authentication. Please visit API Key page in My profile section to obtain the **API key** required for your requests.

Remember, it is essential to keep your API key confidential. Avoid sharing it with others or embedding it in any client-side code like browsers or apps. For secure handling, all production requests should be directed through your backend server, where the API key can be securely fetched from an environment variable or key management service.

Every API request must include your API key in an Authorization HTTP header in the following manner:

```bash
Authorization: Bearer $STRAICO_API_KEY
```

User

GET User information

```
https://api.straico.com/v0/user
```

Overview

This endpoint allows users to fetch details of a specific user from the Straico platform. Users can access information such as the user's first name, last name, the number of coins associated with the account, and the plan they are subscribed to.

Authentication

All requests must include an `Authorization` header with a valid API key.

Request Headers

- Authorization: `Bearer $STRAICO_API_KEY`

Request Body

This endpoint does not require any parameters.

Response

The API responds with a JSON object containing the user's data under the "data" key.

Field	Type	Description
`first_name`	string	The user's first name
`last_name`	string	The user's last name
`coins`	number	The number of coins associated with the account
`plan`	string	The current subscription plan of the user

Model

GET Models information (v.0)

```
https://api.straico.com/v0/models
```

Overview

This endpoint provides users with a comprehensive list of available language models and their associated details within the Straico API ecosystem. By querying this endpoint, users can access up-to-date information about various models and their associated pricing

Authentication

All requests must include an `Authorization` header with a valid API key.

Request Headers

- Authorization: `Bearer $STRAICO_API_KEY`

Request Body

This endpoint does not require any parameters.

Response

The API responds with a JSON object containing an array of model objects. Each model object includes the following information:

Field	Type	Description
`name`	string	Model display name
`model`	string	Unique model identifier for API usage
`pricing`	object	Pricing information, detailing the cost in coins per specified number of words
`max_output`	number	Limit number of tokens (roughly words or word pieces) that can be generated in a single response

GET Models information (v.1)

https://api.straico.com/v1/models

The models information endpoint (v.1) allows users to fetch a list of available models, categorized by type, along with their corresponding details and from the Straico API. This version now includes both chat and image generation models, offering detailed specifications and pricing for each category.

Authentication

All requests must include an `Authorization` header with a valid API key.

Request Headers

- Authorization: `Bearer $STRAICO_API_KEY`

Request Body

This endpoint does not require any parameters.

Response

Upon a successful request, the API responds with a JSON object containing categorized arrays of model objects. The response includes separate arrays for chat models and image generation models.

Response Fields

Field	Type	Description
`chat`	array	Array of chat model objects
`image`	array	Array of image generation model objects

Model Object Fields

Field	Type	Description
`name`	string	Model display name
`model`	string	Unique model identifier for API usage
`word_limit`	number	Maximum number of words the model can process
`pricing`	object	For chat models, includes a `"coins"` value representing the cost per 100 words. For image models, includes aspect ratios (`"square"` , `"landscape"` , and `"portrait"`), each represented as an object containing `"coins"` (the price per image) and `"size"` (the resolution).
`max_output`	number	Limit number of tokens (roughly words or word pieces) that can be

Metadata Object Fields

Field	Type	Description
`editors_choice_level`	number	General valuation given from straico: a value between -1 and 3, higher for better models.
`pros`	array	Array of strings with model advantages.
`cons`	array	Array of strings with model disadvantages.
`applications`	array	Array of strings with areas or fields of knowledge where the model excels.
`features`	array	Array of strings with model additional features, for example, `"Web search"` or `"Image input"`
`icon`	string	Image url of the model icon
`other`	array	Array of strings with other characteristics, like `"Uncensored"` or `"Open`

Prompt

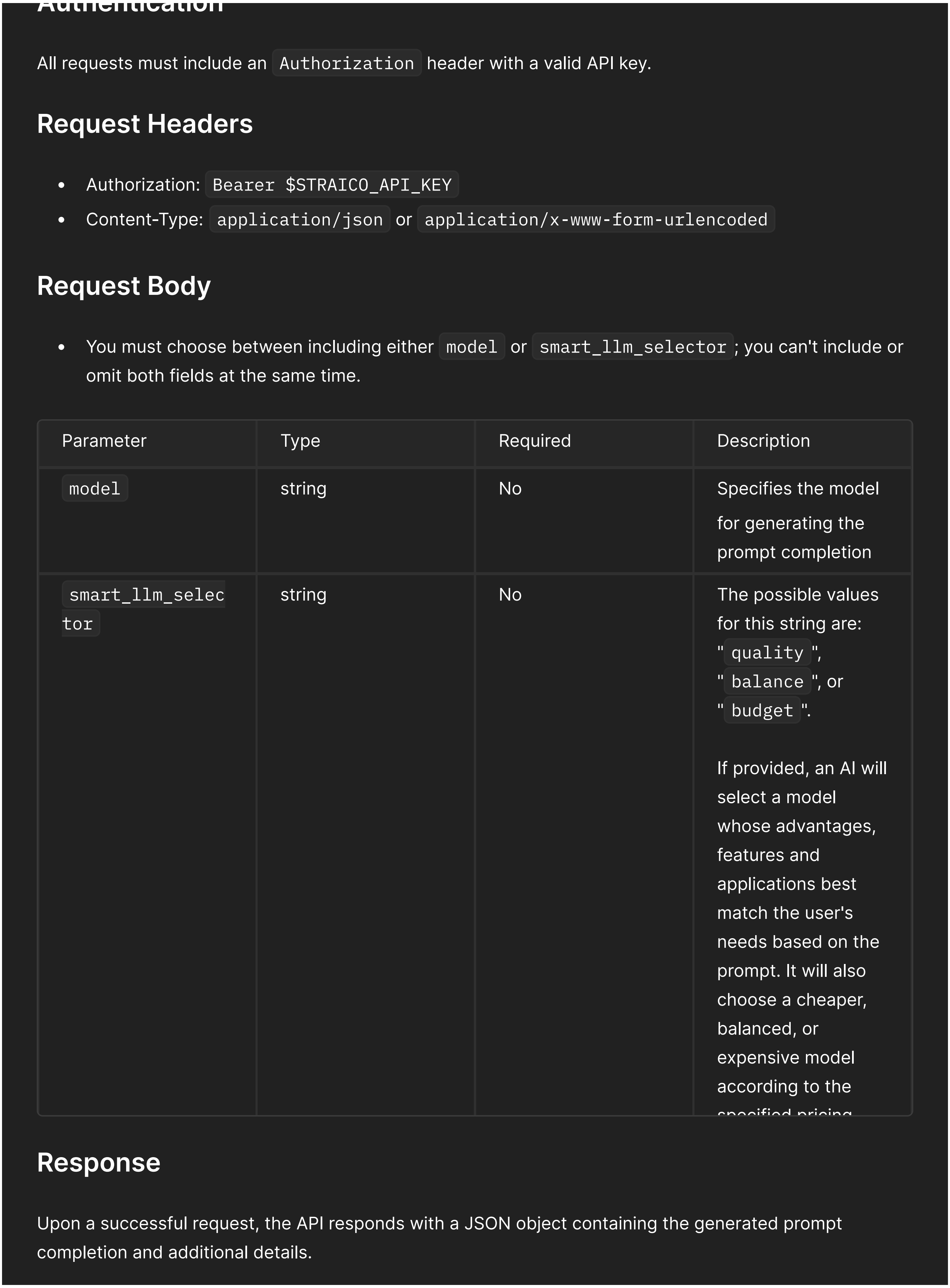
POST Prompt completion (v.0)

https://api.straico.com/v0/prompt/completion

Overview

This endpoint enables users to generate prompt completion based on a specified model using the Straico API. Users can make a POST request to the provided URL, including the necessary headers and parameters, to receive a prompt completion for a given message.

Authentication



Authentication

All requests must include an `Authorization` header with a valid API key.

Request Headers

- Authorization: `Bearer $STRAICO_API_KEY`
- Content-Type: `application/json` or `application/x-www-form-urlencoded`

Request Body

- You must choose between including either `model` or `smart_llm_selector`; you can't include or omit both fields at the same time.

Parameter	Type	Required	Description
`model`	string	No	Specifies the model for generating the prompt completion
`smart_llm_selector`	string	No	The possible values for this string are: "`quality`", "`balance`", or "`budget`". If provided, an AI will select a model whose advantages, features and applications best match the user's needs based on the prompt. It will also choose a cheaper, balanced, or expensive model according to the specified pricing

Response

Upon a successful request, the API responds with a JSON object containing the generated prompt completion and additional details.

Field	Type	Description
`price`	object	Cost breakdown. Contains `input`, `output`, and `total` coins
`words`	object	Word count details. Contains `input`, `output`, and `total` words
`completion`	object	Detailed result for the prompt completion generated
`model_selector_justification`	string	A detailed justification of the smart_llm_selector decision, if it was used.

POST Prompt completion (v.1)

https://api.straico.com/v1/prompt/completion

Overview

The Prompt Completions endpoint (v.1) enables users to generate prompt completions using multiple language models (LLMs) simultaneously. Users can request completions from up to 4 different models in a single request. This version supports the inclusion of YouTube video files and various attachments.

Authentication

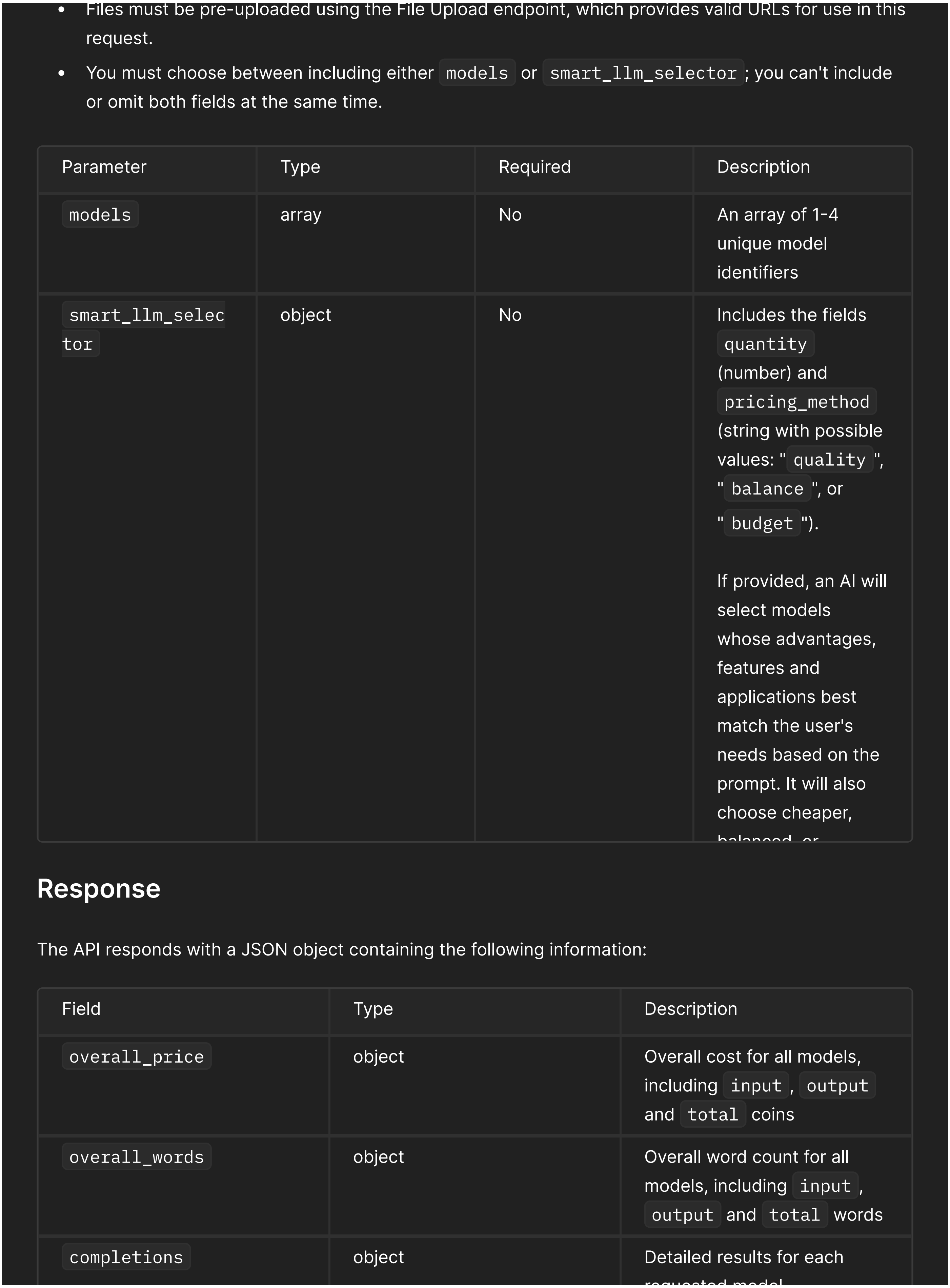
All requests must include an `Authorization` header with a valid API key.

Request Headers

- Authorization: `Bearer $STRAICO_API_KEY`
- Content-Type: `application/json`

Request Body

The request body must include model identifiers, prompt text, and optional file and YouTube URLs, all formatted as JSON.



- Files must be pre-uploaded using the File Upload endpoint, which provides valid URLs for use in this request.
- You must choose between including either `models` or `smart_llm_selector`; you can't include or omit both fields at the same time.

Parameter	Type	Required	Description
`models`	array	No	An array of 1-4 unique model identifiers
`smart_llm_selector`	object	No	Includes the fields `quantity` (number) and `pricing_method` (string with possible values: "`quality`", "`balance`", or "`budget`"). If provided, an AI will select models whose advantages, features and applications best match the user's needs based on the prompt. It will also choose cheaper, balanced, or

Response

The API responds with a JSON object containing the following information:

Field	Type	Description
`overall_price`	object	Overall cost for all models, including `input`, `output` and `total` coins
`overall_words`	object	Overall word count for all models, including `input`, `output` and `total` words
`completions`	object	Detailed results for each requested model

		requested model
`transcripts`	array	If the parameter `display_transcripts` is given as `true`, returns an array of transcripts, each containg a `name` and `text`
`model_selector_justification`	string	A detailed justification of the smart_llm_selector decision, if it was used.

File

POST File upload

https://api.straico.com/v0/file/upload

Overview

This endpoint allows users to upload files for use in the Prompt Completion (v.1) endpoint. Files are securely stored in our cloud storage system, ensuring data integrity and accessibility.

Authentication

All requests must include an `Authorization` header with a valid API key.

Request Headers

- Authorization: `Bearer $STRAICO_API_KEY`
- Content-Type: `multipart/form-data`

Request Body

The request should be sent as `multipart/form-data` with a single file.

- Maximum file size: 25 MB.
- Supported file types: pdf, docx, pptx, txt, xlsx, mp3, mp4, html, csv, json, py, php, js, css, cs, swift

Supported file types: pdf, docx, pptx, txt, xlsx, mp3, mp4, html, csv, json, py, php, js, css, cs, swift, kt, xml, ts, png, jpg, jpeg, webp, gif.

Parameter	Type	Required	Description
`file`	File	Yes	The file to be uploaded

Response

The API responds with a JSON object containing the following information:

Field	Type	Description
`url`	string	The URL of the successfully uploaded file

Image

POST Image generation

https://api.straico.com/v0/image/generation

Overview

This endpoint enables users to generate high-quality images based on textual descriptions. Leveraging advanced AI models, it can create diverse visual content for various applications, from concept art to marketing materials. You can generate up to 100 images per day via the API.

Authentication

All requests must include an `Authorization` header with a valid API key.

Request Headers

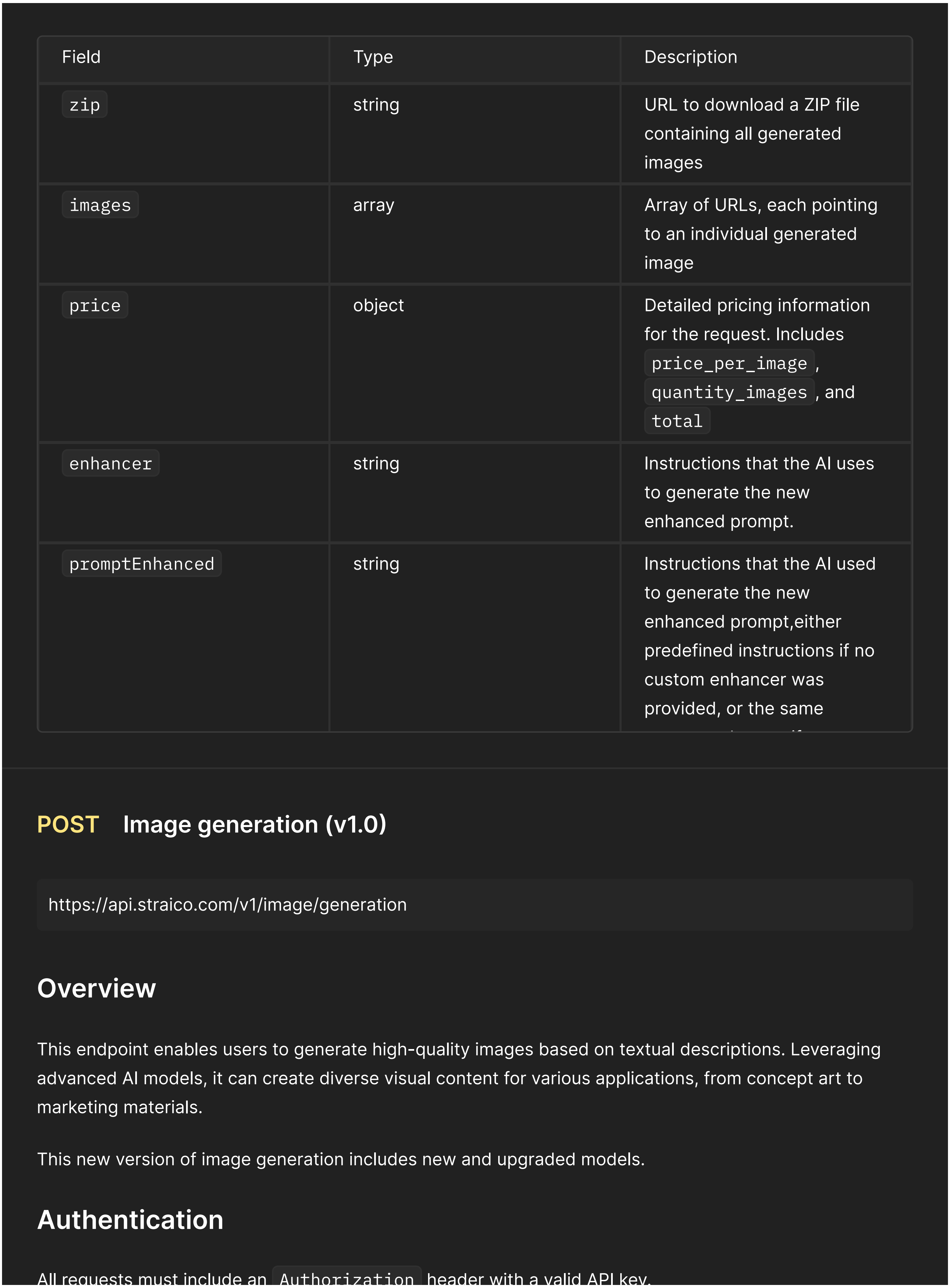
- Authorization: `Bearer $STRAICO_API_KEY`
- Content-Type: `application/json`

Request Body

Parameter	Type	Required	Description
`model`	string	Yes	The AI model to use for image generation, the current models available are: `openai/dall-e-3`, `flux/1.1`, `ideogram/V_2A`, `ideogram/V_2A_TURBO`, `ideogram/V_2` , `ideogram/V_2_TURBO`, `ideogram/V_1`, `ideogram/V_1_TURBO`
`description`	string	Yes	A detailed textual description of the image to be generated
`size`	string	Yes	The desired image

Response

The API responds with a JSON object containing the following information:



Field	Type	Description
`zip`	string	URL to download a ZIP file containing all generated images
`images`	array	Array of URLs, each pointing to an individual generated image
`price`	object	Detailed pricing information for the request. Includes `price_per_image` , `quantity_images` , and `total`
`enhancer`	string	Instructions that the AI uses to generate the new enhanced prompt.
`promptEnhanced`	string	Instructions that the AI used to generate the new enhanced prompt,either predefined instructions if no custom enhancer was provided, or the same

POST Image generation (v1.0)

https://api.straico.com/v1/image/generation

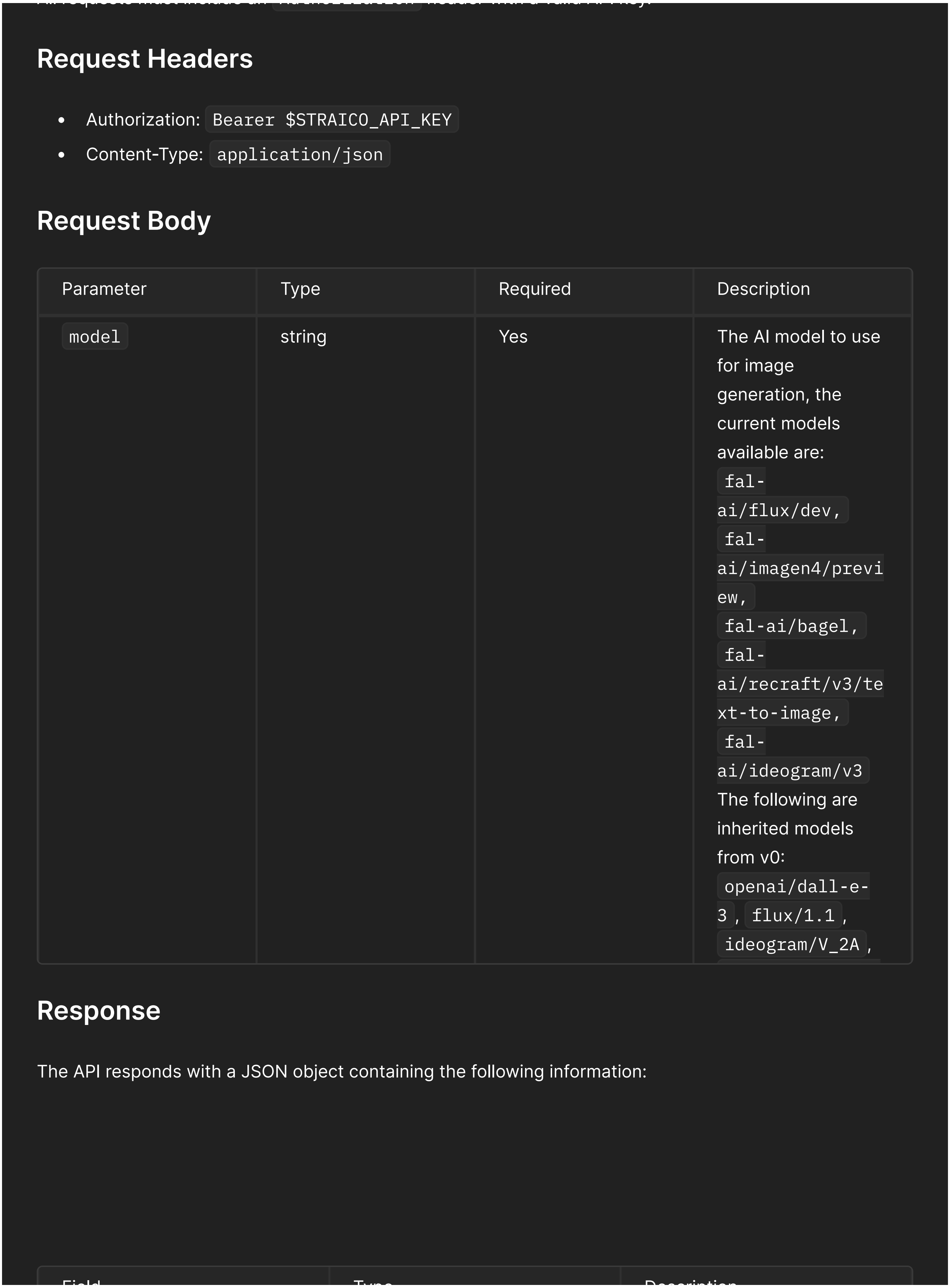
Overview

This endpoint enables users to generate high-quality images based on textual descriptions. Leveraging advanced AI models, it can create diverse visual content for various applications, from concept art to marketing materials.

This new version of image generation includes new and upgraded models.

Authentication

All requests must include an `Authorization` header with a valid API key.



All requests must include an `Authorization` header with a valid API key.

Request Headers

- Authorization: `Bearer $STRAICO_API_KEY`
- Content-Type: `application/json`

Request Body

Parameter	Type	Required	Description
`model`	string	Yes	The AI model to use for image generation, the current models available are: `fal-ai/flux/dev,` `fal-ai/imagen4/preview,` `fal-ai/bagel,` `fal-ai/recraft/v3/text-to-image,` `fal-ai/ideogram/v3` The following are inherited models from v0: `openai/dall-e-3` , `flux/1.1` , `ideogram/V_2A` ,

Response

The API responds with a JSON object containing the following information:

Field	Type	Description

Field	Type	Description
`zip`	string	URL to download a ZIP file containing all generated images
`images`	array	Array of URLs, each pointing to an individual generated image
`price`	object	Detailed pricing information for the request. Includes `price_per_image`, `quantity_images`, and `total`

Video

POST Image to video generation

```
https://api.straico.com/v1/image/tovideo
```

Overview

This endpoint enables users to generate high-quality videos based on attached images.

Authentication

All requests must include an `Authorization` header with a valid API key.

Request Headers

- Authorization: `Bearer $STRAICO_API_KEY`
- Content-Type: `application/json`

Request Body

Parameter	Type	Required	Description
`model`	string	Yes	The AI model to use for image-to-video generation, the current models available are: `fal-ai/kling-video/v2.1/master/image-to-video`, `fal-ai/veo2/image-to-video`, `fal-ai/vidu/q1/image-to-video`, `gen3a_turbo`, `gen4_turbo`
`description`	string	Yes	A detailed textual description of the video to be generated
`size`	string	Yes	The desired image dimensions.

Response

The API responds with a JSON object containing the following information:

Field	Type	Description
`zip`	string	URL to download a ZIP file containing all generated images
`video`	array	Array of URLs, each pointing to an individual generated image
`price`	object	Detailed pricing information for the request. Includes

		for the request. Includes `price_per_video` and `total`

TTS

GET Eleven Labs Voices

```
https://api.straico.com/v1/tts/elevenlabslist
```

Overview

This endpoint returns the list of available voices in Eleven Labs. You can retrieve the ids for the endpoint "Create Text To Speech".

Authentication

All requests must include an `Authorization` header with a valid API key.

Request Headers

- Authorization: `Bearer $STRAICO_API_KEY`

POST Create Text-to-Speech

```
https://api.straico.com/v1/tts/create
```

Create Text-to-Speech (TTS)

This endpoint allows users to create a text-to-speech (TTS) conversion by providing specific parameters related to the desired voice model and text content.

Authentication

All requests must include an `Authorization` header with a valid API key.

All requests must include an `Authorization` header with a valid API key.

Request Headers

- Authorization: `Bearer $STRAICO_API_KEY`
- Content-Type: `x-www-form-urlencoded`

Request Body

Parameter	Type	Required	Description
`model`	string	Yes	Specifies the of model to use for the TTS conversion. Currently, you can choose between `eleven_multilingual_v2` and `tts-1`.

`text`	string	Yes	The actual text content that you want to convert into speech. This parameter is essential for the TTS process. (Up to 4,000 characters)
`voice_id`	string	Yes	Identifies the specific voice to be used for the TTS conversion. If the selected model is

Agents

This is the first version of the documentation of the API endpoints for agents, allowing users to upload a file and converting it into a suitable format for RAG.

POST Create Agent

https://api.straico.com/v0/agent

Creates a new agent in the datebase for the user.

json

```json
{
    "success": true,
    "data": {
        "uuidv4": "73648dd8-79d9-4e97-8dd1-2ef2df2b8843",
        "user_id": "743ada93ff7a1d6822131d3f5",
        "default_llm": "anthropic/claude-3.5-sonnet",
        "custom_prompt": "Act as an expert in airline regulations, based on an att
        "name": "Expert in airline regulations",
        "description": "Agent acting as an expert in airline regulations",
        "status": "active",
        "tags": ["assistant","rag"],
```

AUTHORIZATION Bearer Token

Token	<token>

Body urlencoded

name	<A name for the agent>
custom_prompt	<A custom prompt for the agent behaviour>
default_llm	<A model that the agent will use for processing prompts>
description	<A brief description of what the model does>
tags	<An array of tags for the agent. Example: ["assistant","rag"]>

POST Add RAG to Agent

```
https://stapi.straico.com/v0/agent/<agent-id>/rag
```

Creates a new agent in the datebase for the user.

json

```json
{
    "success": true,
    "data": {
        "uuidv4": "73648dd8-79d9-4e97-8dd1-2ef2df2b8843",
        "user_id": "743ada93ff7a1d6822131d3f5",
        "default_llm": "anthropic/claude-3.5-sonnet",
        "custom_prompt": "Act as an expert in airline regulations, based on an att
        "name": "Expert in airline regulations",
        "description": "Agent acting as an expert in airline regulations",
        "status": "active",
        "tags": ["assistant","rag"],
```

AUTHORIZATION Bearer Token

Token	<token>

Body raw (json)

json

```json
{
  "rag": "670469b677rwtrwwr5b0903"
}
```

GET Agent details

https://api.straico.com/v0/agent/<agent-id>

Get Agent Details

This endpoint makes an HTTP GET request to retrieve details of a specific agent identified by the provided ID.

Request Parameters

- : the id of the required agent

Response

json

```json
{
    "success": true,
    "data": {
        "_id": "66df87d2bb560c6acabb686s",
        "uuidv4": "c85254c4-0f89-46d7-bf28-428eed764e16",
        "default_llm": "openai/gpt-4o",
        "custom_prompt": "Act a as expert in airline regulations",
        "name": "Expert in airline regulations",
        "description": "Agent expert in airline regulations",
        "status": "active",
        "tags": [],
```

AUTHORIZATION Bearer Token

Token	<token>

GET List of agents

```
https://api.straico.com/v0/agent/
```

The endpoint retrieves the list of agents created by and available to the user.

Response

The response for this request follows the JSON schema below:

```json
{
    "success": true,
    "data": [
        {
            "_id": "66e8927659c3f8a86098c8da",
            "uuidv4": "73648d38-79d9-4e97-8031-2ef2df2b8843",
            "user_id": "64ada93ff7a1fr322131d3f5",
            "default_llm": "anthropic/claude-3.5-sonnet",
            "custom_prompt": "Act as an expert in airline regulations, based on an
            "name": "Expert in airline regulations",
            "description": "Agent acting as an expert in airline regulations",
```

AUTHORIZATION Bearer Token

Token	<token>

PUT Update agent

https://api.straico.com/v0/agent/<agent-id>

Update Agent Details

This endpoint allows updating the details of a specific agent.

Request

- Method: PUT
- Endpoint: `https://stapi.straico.com/v0/agent/`

Response

The response is in the form of a JSON schema.

json

```json
"success": true,
    "data": {
        "_id": "66df87d2bb560c6acabb6822",
        "uuidv4": "c85254c4-0f89-46d7-bf28-428eed764e16",
        "user_id": "64ada93ff7a1d6822131d3f5",
        "default_llm": "openai/gpt-4o",
        "custom_prompt": "Actúa como un buen abogado penalista, que responda pregu
        "name": "Experto en Código Penal Colombiano",
        "description": "Agente que actúa como abogado penalista experto en código
        "status": "active",
        "tags": [],
```

AUTHORIZATION Bearer Token

Token	<token>

Body urlencoded

name	<New value of name>
<any other attribute>	<New value of any other attribute>

POST Agent prompt completion

https://api.straico.com/v0/agent/<agent-id>/prompt

Agent Prompt Completion

This endpoint allows the user to make an HTTP POST request to prompt the completion of an agent message. It uses the previously uploaded RAG base and the agent's default model.

Request Body

- `prompt` (text): A text prompt for the RAG model.

Optional Parameters

RAG bases are build using FAISS format. To retrieve results, we use a VectorStoreRetriever. This entity has custom parameters that can modify the results used for prompt completion. Check this document for more information: https://api.python.langchain.com/en/latest/vectorstores/langchain_community.vectorstores.faiss.FAISS.html#langchain_community.vectorstores.faiss.FAISS.as_retriever

- `search_type` : one of the following: `similarity` , `mmr` or `similarity_score_threshold`
- `k` : number of documents to return
- `fetch_k` : ammount of documents to pass to MMR algorithm
- `lambda_mult` : diversity of results return by MMR, 1 for minimum and 0 for maximum
- `score_threshold` : Minimum relevance threshold for `similarity_score_threshold`

Notes on coins

- Total coins will be based on the regular fees for each LLM available in Straico.
- **WARNING:** coins will consider the number of words in the `references` .

Response

json

```json
{
    "success": true,
    "data": {
        "answer": "This appears to be a section from an academic regulations or st
        "references": [
            {
                "page_content": "diante tiene matrícula condicional como consecuen
                "page": 73
            },
            {
                "page_content": "institucional.",
```

IMPORTANT: Despite the response has now a `data` object, the response still keeps the object `response`, in order to keep backward compatibility with older versions of the API.

AUTHORIZATION Bearer Token

Token	<token>

Body urlencoded

prompt	What's the document about?

DELETE Delete agent

```
https://api.straico.com/v0/agent/<agent-id>
```

Delete Agent

This endpoint is used to delete a specific agent by providing the agent ID in the URL.

Request Body

This request does not require a request body.

Response

Example Response:

json

```
{
    "success": true,
    "message": "Agent deleted successfully"
}
```

AUTHORIZATION Bearer Token

Token	<token>

Rag

POST Create RAG

```
https://api.straico.com/v0/rag
```

Create a new RAG

This endpoint allows the user to create a new RAG base in the database. The request should be sent as an HTTP POST .Request Body

The request should have a form-data body type with the following parameters:

- `name` (text): Represents the name of the RAG base
- `description` (text): Represents the description of the agent.
- `files` (file): Represents the files to be attached (up to 4 files). Accepted file extensions are: pdf, docx, csv, txt, xlsx, py.
- `chunking_method` (text - optional): Represents the chunking method to be use for generating the RAG base. The default value is `fixed_size`. Depending on the value set, there are other optional parameters than can be set:

`chunking method`	Available optional parameters	Should perform quiet well with	Cons
`fixed_size`	`chunk_size` (default value: 1000) `chunk_overlap` (default value: 50) `separator` (default value: \n)	Not so elaborated text	Does not consider text structure and semantics
`recursive`	`chunk_size` (default value: 1000) `chunk_overlap` (default value: 50) `separators` (default value: ["\n\n", "\n", " ", ""])	Not so elaborated text	Does not consider text structure and semantics
`markdown`	`chunk_size` (default value: 1000) `chunk_overlap` (default value: 50)	Files with Markdown	Only suitable for files with Markdown
`python`	`chunk_size` (default value: 1000) `chunk_overlap`	Python files	Only suitable for Python Files

Response

- Status: 201
- Content-Type: application/json

The response will contain a JSON object with the following structure:

json

```json
{
    "success": true,
    "data": {
        "user_id": "64ada93f22131d3f5",
        "name": "Rag de prueba txt",
        "rag_url": "https://prompt-rack.s3.amazonaws.com/api/rag/64ada93ff7a1d6822
        "original_filename": "sample_txt.txt",
        "chunking_method": "fixed_size",
        "chunk_size": 1000,
        "chunk_overlap": 50,
        "_id": "670565d07e1234eb",
```

Notes on coins and processing time

- Current fee is 0.1 coins / 100 words of the processed files.
- Processing the file may take several minutes. It takes approximately 1 minute per 50.000 words of the original documents.

AUTHORIZATION Bearer Token

Token	<token>

Body formdata

name	Rag entity
description	A description of the RAG entity
files	

GET List of RAG's

```
https://api.straico.com/v0/rag/user
```

Get list of RAGs for user

This endpoint returns the list of RAG bases for the user.

Response

json

```json
{
    "success": true,
    "data": [
        {
            "_id": "670558840d774c63b439",
            "user_id": "64ada93ff7a131d3f5",
            "name": "Test",
            "rag_url": "https://prompt-rack.s3.amazonaws.com/api/rag/64ada93ff7a1d
            "original_filename": "sample_txt.txt, sample_excel.xlsx, 2000_codigope
            "chunking_method": "fixed_size",
            "chunk_size": 1000,
```

AUTHORIZATION Bearer Token

Token	<token>

GET RAG by ID

https://api.straico.com/v0/rag/<rag-id>

Get RAG base by ID

This endpoint makes an HTTP GET request to retrieve details of a specific RAG base identified by the provided ID.

Request Parameters

- id: the id of the required RAG base

Expected Response

Upon successful execution, the API returns a JSON object with the following format:

```json
{
    "success": true,
    "data": {
        "_id": "670565d02fc07e1234eb",
        "user_id": "64ada922131d3f5",
        "name": "Test",
        "rag_url": "https://prompt-rack.s3.amazonaws.com/api/rag/64ada93ff7a1d6822
        "original_filename": "sample_txt.txt",
        "chunking_method": "fixed_size",
        "chunk_size": 1000,
        "chunk_overlap": 50,
```

AUTHORIZATION Bearer Token

Token	<token>

PUT Update RAG

```
https://api.straico.com/v0/rag/<rag-id>
```

This endpoint is used to update a RAG given some files to enhance the existing RAG base.

Request Body

- form-data
  - `files` (text): The files to be processed and added to the existing RAG base

Response

json

```json
{
    "type": "object",
    "properties": {
        // Add properties based on the actual response
  {
    "success": true,
    "data": {
        "_id": "341234123412344s4ae",
        "user_id": "jeqjreqwez34123",
        "name": "A RAG to be updated",
        "description": "Testing a RAG base to be updated later",
```

AUTHORIZATION Bearer Token

Token	<token>

Body formdata

files

DELETE Delete RAG

https://api.straico.com/v0/rag/<rag-id>

Delete RAG

This endpoint is used to delete a specific agent by providing the rag ID in the URL.

Response

json

```json
{
    "success": true,
    "message": "RAG deleted successfully"
}
```

AUTHORIZATION Bearer Token

Token	<token>

POST RAG prompt completion

```
https://api.straico.com/v0/rag/<rag-id>/prompt
```

RAG prompt compleition

This endpoint is used to submit a prompt to a specific RAG (Retrieval-Augmented Generation) model.

Request Body

- `prompt` (text): A text prompt for the RAG model.
- `model` (text): The specific LLM to be used.

Optional Parameters

RAG bases are build using FAISS format. To retrieve results, we use a VectorStoreRetriever. This entity has custom parameters that can modify the results used for prompt completion. Check this document for more information: https://api.python.langchain.com/en/latest/vectorstores/langchain_community.vectorstores.faiss.FAISS.html#langchain_community.vectorstores.faiss.FAISS.as_retriever

- `search_type` : one of the following: `similarity` , `mmr` or `similarity_score_threshold`
- `k` : number of documents to return
- `fetch_k` : ammount of documents to pass to MMR algorithm

- `lambda_mult` : diversity of results return by MMR, 1 for minimum and 0 for maximum
- `score_threshold` : Minimum relevance threshold for `similarity_score_threshold`

Notes on coins

- Total coins will be based on the regular fees for each LLM available in Straico.
- **WARNING:** coins will consider the number of words in the `references` .

Response

The response is a JSON object with the following schema:

json

```json
{
    "success": true,
    "data": {
        "answer": "This appears to be a section from an academic regulations or st
        "references": [
            {
                "page_content": "diante tiene matrícula condicional como consecuen
                "page": 73
            },
            {
                "page_content": "institucional.",
```

IMPORTANT: Despite the response has now a `data` object, the response still keeps the object `response` , in order to keep backward compatibility with older versions of the API.

AUTHORIZATION Bearer Token

Token	<token>

Body urlencoded

prompt	What happens if the flight is delayed?
model	anthropic/claude-3.5-sonnet