

Qanouny Legal Assistant API Documentation

Server deployed at <http://52.143.145.178:8000/>

Important Note on Request Format:

All POST requests in this documentation (endpoints starting with /api/query/ and /api/init) must be sent using **Form Data**

1. Health Check

Endpoint: GET /

Description: Checks if the API is

Request Parameters:

- None

Response Explanation:

- status (string): Shows the current status of the API, e.g., "running".
- message (string): A simple description of the API, e.g., "Legal Assistant API".
- endpoints (object): Lists the main functional endpoints for reference:
 - text: <http://52.143.145.178:8000/api/query/text>
 - audio: <http://52.143.145.178:8000/api/query/audio>
 - image: <http://52.143.145.178:8000/api/query/file>

```
{
  "status": "running",
  "message": "Legal Assistant API",
  "endpoints": {
    "text": "/api/query/text",
    "audio": "/api/query/audio",
    "image": "/api/query/image"
  }
}
```

2. Text Query

Endpoint: POST <http://52.143.145.178:8000/api/query/text>

Description: Send a legal question as text.

Request Parameter:

- question (string, required): The legal question in Arabic.

Success Response Explanation:

- success (boolean): True.
- query (string): The exact question received by the API.
- answer (string): The assistant's legal answer based on the query.
- risk_level (string): The assessed risk level for the legal issue ("red", "amber", "green").
- sources (array): References or documents used to generate the answer.
- cited_sources (array of objects): Contains the specific, structured legal articles that the answer directly refers to. This includes the full text of the article.
- terms_summary (array of strings): Simplified explanation of complex legal terms found in the context.

```
{
  "success": True,
  "query": result['query'],
  "answer": result['answer'],
  "risk_level": result['risk_level'],
  "sources": result['sources'],
  "cited_sources": result['cited_sources'],
  "terms_summary": result.get("terms_summary", "")
}
```

Error Response:

- success (boolean): False.
- error_message (string): A user-friendly error description.

```
{
  "success": False,
  "error_message": "حدث خطأ داخلي أثناء معالجة استفسارك. نأسف لذلك."
}
```

Validation Rules:

- question must not be empty.
- Max length: 5000 characters

3. Audio Query

Endpoint: POST <http://52.143.145.178:8000/api/query/audio>

Description: Upload an audio file for speech-to-text query.

Request Parameter:

- audio_file (file, required): The audio file (.WAV).

Success Response Explanation:

- success (boolean): True.
- query (string): The exact question received by the API.
- answer (string): The assistant's legal answer based on the query.
- risk_level (string): The assessed risk level for the legal issue ("red", "amber", "green").
- sources (array): References or documents used to generate the answer.
- cited_sources (array of objects): Contains the specific, structured legal articles that the answer directly refers to. This includes the full text of the article.
- terms_summary (array of strings): Simplified explanation of complex legal terms found in the context.

```
{
  "success": True,
  "query": result['query'],
  "answer": result['answer'],
  "risk_level": result['risk_level'],
  "sources": result['sources'],
  "cited_sources": result['cited_sources'],
  "terms_summary": result.get("terms_summary", "")
}
```

Error Response:

- success (boolean): False.
- error_message (string): A user-friendly error description.

```
{
  "success": False,
  "error_message": "حدث خطأ داخلي أثناء معالجة استفسارك. نأسف لذلك."
}
```

Validation Rules:

- File must be in .WAV format.
- Only one audio file can be sent at a time

4. File Query (Image / PDF)

Endpoint: POST <http://52.143.145.178:8000/api/query/file>

Description: Upload an image or PDF for OCR + Q&A.

Request Parameters:

- file (file, required): Image (PNG, JPG, JPEG) or PDF.
- question (string, required): The legal question related to the file content.

Response Explanation:

- success (boolean): True if the file was processed successfully.
- full_text (string): Text extracted from the uploaded file.
- query (string): The user's question.
- answer (string): The assistant's legal answer.
- risk_level (string): Risk assessment.
- sources (array of strings): References used.
- cited_sources (array of objects): Contains the specific, structured legal articles that the answer directly refers to. This includes the full text of the article.
- terms_summary (array of strings): Simplified explanation of complex legal terms found in the context.

```
{
    "success": True,
    "full_text": result['full_text'],
    "query": result['query'],
    "answer": result['answer'],
    "risk_level": result['risk_level'],
    "sources": result['sources'],
    "cited_sources": result['cited_sources'],
    "terms_summary": result.get("terms_summary", "")
}
```

Error Response:

- success (boolean): False.
- error_message (string): A user-friendly error description.

```
{
    "success": False,
    "error_message": "حدث خطأ داخلي أثناء معالجة استفسارك. نأسف لذلك."
}
```

Validation Rules:

- File must be an accepted format.
- question must not be empty.
- Max size: 5mb

5. Chat Initialization (reset previously)

Endpoint: POST <http://52.143.145.178:8000/api/init>

Description: Clears the conversation history for the current session. Useful if the user wants to start a new query context without previous interactions affecting the answers.

Request Parameters:

- name (string, required): The name of the user.
- gender (string, required): The gender of the user.

Response Explanation:

- success (boolean): Indicates whether the conversation history was successfully reset.
- message (string): Confirmation message, e.g., "Conversation history reset successfully".

Validation Rules:

- Endpoint can be called anytime; it will reset the assistant's memory for that session.

Data Structure & Usage Guidance for Sources

The API provides two distinct fields related to the legal material used, each serving a different purpose:

1. sources (Internal Tracing/Metadata)

Field	Type	Sub-Fields	Usage
sources	array of objects	id, category, source_file	Internal Use Only. This array lists all the raw documents/articles retrieved from the knowledge base before the AI selects the final relevant citations. It is used for backend tracing and debugging.

Eample:

```
[
  {
    "source_file": "penal Law.csv",
    "id": "3011",
    "category": "قانون العقوبات"
  },
  {
    "source_file": "civil Law.csv",
    "category": "القانون المدني",
    "id": "155"
  },
  {
    "id": "3018",
    "category": "قانون العقوبات",
    "source_file": "penal Law.csv"
  }
]
```

2. cited_sources (User Content/UI Display)

Field	Type	Sub-Fields	Usage
cited_sources	array of objects	id, category, text	Primary UI Content. This array contains only the specific legal articles the AI actually referenced in the generated answer. Each object includes the full text of the article in the text field.

Example:

```
[
  {
    "id": "3011",
    "category": "قانون العقوبات",
    "text": "... المادة 366 كل نهب أو إتلاف شيء من البضائع أو الأمتعة أو المحصولات وقع من جماعة"
  },
  {
    "id": "3018",
    "category": "قانون العقوبات",
    "text": "... لمادة 361 كل من خرب أو أتلف عمداً أموالاً ثابتة أو منقول"
  }
]
```

