

```
C:\Users\HC>C:\Users\HC\AppData\Local\Programs\Python\Python311\  
python.exe --version  
Python 3.11.1  
  
C:\Users\HC>E:  
  
E:>cd "all vs tasks\Dev town project\study-bot"  
  
E:\all vs tasks\Dev town project\study-bot>C:\Users\HC\AppData\Local\Programs\Python\Python311\python.exe -m venv venv  
  
E:\all vs tasks\Dev town project\study-bot>venv\Scripts\activate  
  
(venv) E:\all vs tasks\Dev town project\study-bot>(venv) E:\all  
vs tasks\Dev town project\study-bot>  
E:\all was unexpected at this time.  
  
(venv) E:\all vs tasks\Dev town project\study-bot>python --versi  
on  
Python 3.11.1  
  
(venv) E:\all vs tasks\Dev town project\study-bot>
```

```
(venv) E:\all vs tasks\Dev town project\study-bot>python chatbot  
.py  
=====  
🤖 STUDY BOT - Google Gemini Version  
=====  
Type 'quit' to exit  
-----  
  
👤 You: hi  
🤖 Bot: Thinking...  
  
✖ Error: Error calling model 'gemini-1.5-flash' (NOT_FOUND): 40  
4 NOT_FOUND. {'error': {'code': 404, 'message': 'models/gemini-1  
.5-flash is not found for API version v1beta, or is not supporte  
d for generateContent. Call ListModels to see the list of availa  
ble models and their supported methods.', 'status': 'NOT_FOUND'  
}  
Please check your API key and internet connection.  
  
👤 You: bye  
🤖 Bot: Goodbye! Happy studying! 🌟  
  
(venv) E:\all vs tasks\Dev town project\study-bot>
```

Pretty print ✓

```
{  
    "message": "Welcome to Study Bot API",  
    "version": "1.0.0 (Gemini)",  
    "author": "Study Bot Project",  
    "endpoints": {  
        "/": "GET - This welcome message",  
        "/health": "GET - Check API and database status",  
        "/docs": "GET - Interactive Swagger UI documentation",  
        "/redoc": "GET - Alternative ReDoc documentation",  
        "/openapi.json": "GET - OpenAPI JSON schema",  
        "/chat": "POST - Send message to chatbot",  
        "/history": "POST - Get chat history for a user"  
    },  
    "how_to_use": {  
        "chat": "Send POST request to /chat with {'user_id': 'name', 'message': 'question'}",  
        "history": "Send POST request to /history with {'user_id': 'name', 'limit': 5}"  
    }  
}
```

The screenshot shows a web browser displaying the API documentation at 127.0.0.1:8000/redoc. The page title is "Study Bot API (1.0.0)". On the left, there's a sidebar with links: Root, Health Check, Chat, Get History, Info, Models, and Test. The "Root" link is currently selected. The main content area displays the API's welcome message and information about using Google Gemini. Below this, there's a "Responses" section with a button for "200 Successful Response". To the right, there's a "Response samples" panel showing a single response entry for a GET request to the root endpoint, which includes a Content type of application/json and a null value. A "Copy" button is also present in this panel.

The screenshot displays the API documentation for a service using the Redocly interface, showing two main endpoints: `/health` and `/chat`.

Health Check

- Description:** Check if API and database are working.
- Responses:**
 - 200 Successful Response:** Content type: application/json, body: null.

Chat

- Description:** Send message to chatbot.
- Example request:** `{"user_id": "zzz", "message": "What is photosynthesis?"}`
- REQUEST BODY SCHEMA:** application/json, required:
 - `user_id`: string (User ID)
 - `message`: string (Message)
- Responses:**
 - 200 Successful Response:** Content type: application/json, body: null.
 - 422 Validation Error:** Content type: application/json, body: null.

The screenshot shows a dark-themed API documentation interface. On the left, a sidebar lists endpoints: Root, Health Check, Chat, Get History, Info (which is selected), Models, and Test. The main content area is for the 'Get History' endpoint.

HTTP `GET /history` Get detailed API information

Responses

`> 200 Successful Response`

Models

Get available AI models

Responses

`> 200 Successful Response`

Get History

Get chat history for a user

Example request: `{"user_id": "jizzo", "limit": 5}`

REQUEST BODY SCHEMA: `application/json`

required

`user_id` string (User id)

`limit` Limit (integer) or Limit (null) (Limit)
Default: 5

Responses

`> 200 Successful Response`

`> 422 Validation Error`

API docs by Redocly

Response samples

`200`

Content type: application/json

null

`Copy`

GET /models

Responses

`> 200 Successful Response`

Request samples

Payload

Content type: application/json

`{ "user_id": "string", "limit": 5 }`

`Copy`

Response samples

`200` `422`

Content type: application/json

API docs by Redocly

The screenshot shows a dark-themed API documentation interface. On the left, a sidebar lists endpoints: Root, Health Check, Chat, Get History, Info (which is selected), Models, and Test. The main content area is for the 'Test' endpoint.

HTTP `GET /test` Simple test endpoint to check if API is working

Responses

`> 200 Successful Response`

Test

Simple test endpoint to check if API is working

Responses

`> 200 Successful Response`

Response samples

`200`

Content type: application/json

null

`Copy`

```
{  
    "api_status": "healthy",  
    "database": "connected",  
    "api_key": "available",  
    "model": "models/gemini-2.5-flash",  
    "timestamp": "2026-02-22 09:52:10.475566",  
    "server": "running"  
}
```