Central API Access to LLMs at KIT via the AI-Toolbox

Version 1; 19.11.2025

Use modern AI language models easily and securely via a central interface (API). This solution is aimed at all employees and facilitates the integration of Large Language Models (LLMs) into your own applications and workflows.

This service is provided by the Scientific Computing Center (SCC) as part of the "GenAI@KIT" strategy project.

Your advantages at a glance:

- **No need for a separate account with providers:** Log in with your KIT account and use a self-generated API key. Separate registrations, e.g. with OpenAI or Microsoft, are not required.
- **Central token procurement:** KIT purchases the necessary usage quotas and makes them available to everyone. In this way, you benefit from better conditions.
- **Improved data protection and security standards:** Thanks to framework agreements, e.g. with Microsoft for the Azure Cloud, KIT has stricter requirements for handling your data than for individual use.

Billing and usage information (fair use principle):
At this time, you do not incur any personal costs for using the Service. Billing is carried out centrally by KIT. The service is designed to support your daily work and research in an uncomplicated way. You are welcome to use it to the usual extent.

However, if you are planning larger, automated projects – e.g. processing complete research datasets with many thousands of API calls – please contact the responsible team in advance via the service address ki-toolbox@scc.kit.edu . This allows us to plan resources together and ensure a smooth process for all users.

Special case: Service development and future service accounts
If you plan to develop an automated service, bot, or application that uses the API on a permanent basis, you will need to set up a dedicated service account for it. Please have this service account created by your ITB (IT representative) for each service. The service account must then be activated for use by sending an e-mail request (ki-toolbox@scc.kit.edu) to the AI-Toolbox team.

This procedure ensures a clear separation between personal use and business use.

The current resources are intended for individual, official use. Therefore, an agreement is mandatory. Conformity with the EU AI Act and other legal requirements must also be clarified on their own responsibility, e.g. obligations arising from the provider and operator role.

System overview: How does the service work?

The entire AI-Toolbox, both the web interface and the API, is initially only accessible within the KIT network. For access from outside, a connection via the KIT VPN is mandatory.

The system consists of two main components:

1. The web interface:
   - Available at: https://ki-toolbox.scc.kit.edu/
   - Provides an interactive chat environment comparable to that of ChatGPT.
   - You can also generate your personal API key here.

2. The API:
   - Provides an OpenAI-compatible endpoint so you can continue to use existing applications and scripts developed for the OpenAI API with minimal customization.
   - The API endpoint is: https://ki-toolbox.scc.kit.edu/api

Through this service, you have access to a selection of different models:

- **Local models**: These are operated entirely on the infrastructure of the SCC at KIT.
- **Cloud models**: Models from external providers such as OpenAI, e.g. provided via the Microsoft Azure cloud at a German location.
- Please do not use the two models standard-external and standard-local via the API. The two models have a customized system prompt that can change your own system prompt when used via the API and thus lead to unwanted results.

Information Security, Data Protection and Copyright

The responsible handling of data is of paramount importance when using AI models.

Your responsibility: Copyright and licenses: If you upload documents to the system (e.g. for RAG applications), it is your personal responsibility to ensure that you have the necessary rights of use. Check the license of the document in question and make sure that processing in this context is permitted.

The data flow: Where is my data and when?
The system has a vector database for retrieval-augmented generation (RAG) hosted locally at the SCC.

- **Data storage (RAG):** When you upload documents, they are processed and stored in the local vector database. In this state, your data does not leave KIT.
- **Data processing (request):** When you make a request, your prompt and the relevant data snippets found by the RAG system are sent to the selected model.
- **IMPORTANT**: If you are using a cloud model (e.g. *azure.gpt-4.1-mini*), this combined data is transferred to the external provider (Microsoft Azure) at the time of the query. In the case of local models, the data remains within KIT.

Three-step model for classifying data (guidance)Please strictly adhere to the following classification:

- **Stage 1: Public data**
  - **Description:** Data that is publicly available without restrictions.
  - **Classification:** For this type of data, all available models (on-premises and cloud) can usually be used.
- **Level 2: Confidential but semi-public data**
  - **Description**: Data that is not intended for the general public, but whose disclosure does not pose a major risk (e.g. KIT intranet content).
  - **Classification**: Depending on the case, the OpenAI models in the Azure Cloud can be used for this purpose. Contractual regulations often provide an adequate level of protection here. These are classified under Azure in Open WebUI and follow the naming scheme *azure.[...]*.
- **Level 3: Highly confidential or secret data**
  - **Description**: Data the disclosure of which could cause significant harm (sensitive research data, personal data with a high need for protection).
  - **Classification**: Only the local models may be used for this data. These are classified under KIT in Open WebUI and follow the kit naming scheme.*[...]*.

Responsible use of resources:
Use smaller, cheaper models (e.g. GPT-4.1-nano) first and only use the more powerful and expensive models when necessary.

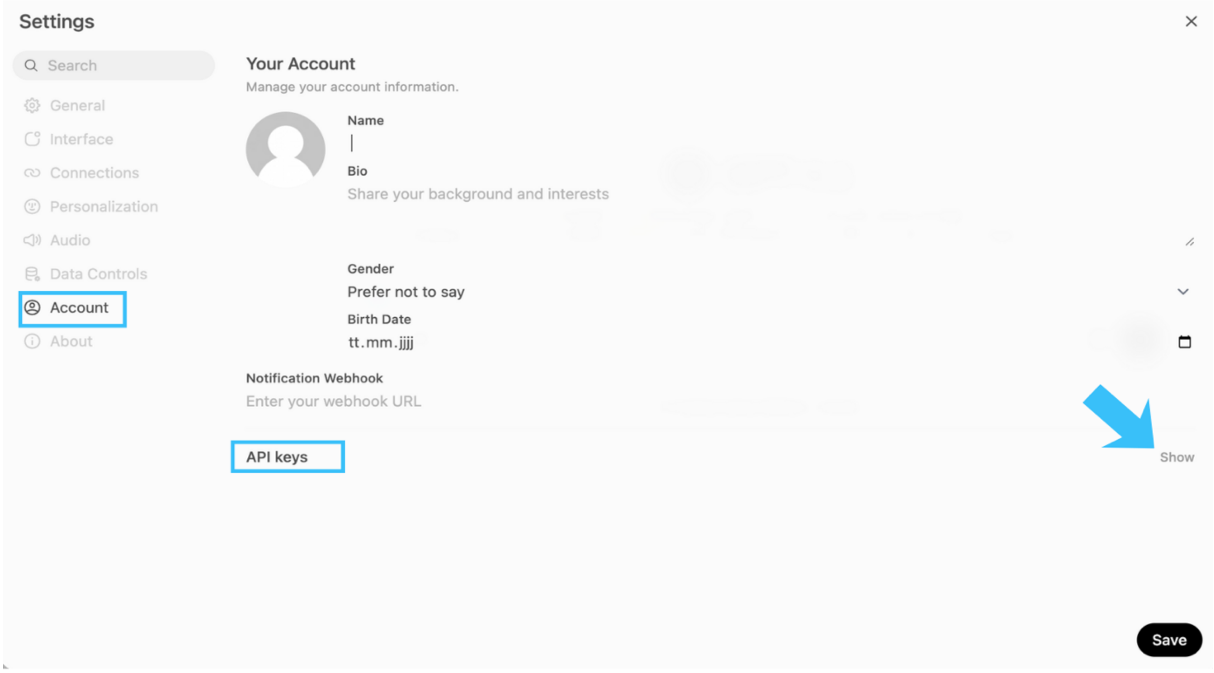
Instructions for API usage (Chat Completion)

Interactive API Documentation for ReferenceThe most comprehensive and always up-to-date reference for all endpoints and parameters can be found in the Interactive API Documentation (Swagger UI**).** There you can also test requests directly.

- Available at: https://ki-toolbox.scc.kit.edu/ docs

Step 1: Generate API Key

1. Log in to https://ki-toolbox.scc.kit.edu/ .
2. Use the gear icon to navigate to **Settings > account**.
3. Click Create API Key and copy the key. Treat it like a password!



Settings

Your Account
Manage your account information.

Name

Bio
Share your background and interests

Gender
Prefer not to say

Birth Date
tt.mm.jjjj

Notification Webhook
Enter your webhook URL

Step 2: Use the OpenAI-compatible API endpointThe base endpoint is *`https://ki-toolbox.scc.kit.edu/api/v1`*. The endpoint for chat requests is: `https://ki-toolbox.scc.kit.edu/api/v1/chat/completions`

Step 3: Choose the right modelIdentify the model's ID (e.g. *`azure.gpt-4.1-mini`* or *`GPT-OSS:120b`*

Step 4: Make an API request (examples)
curl example:

```
curl -X POST https://ki-toolbox.scc.kit.edu/api/v1/chat/completions \
     -H "Authorization: Bearer IHR_PERSÖNLICHER_API_SCHLÜSSEL" \
     -H "Content-Type: application/json" \
     -d '{
         "model": "azure.gpt-4.1-mini",
         "messages": [
             {
                 "role": "system",
                 "content": "You are a helpful assistant at KIT."
             },
             {
                 "role": "user",
                 "content": "Explain the principle of Rayleigh scattering
in three sentences."
             }
         ]
     }'
```

Python example with the openai library:
(Prerequisite: *pip install openai*)

```
from openai import OpenAI

# Configure API key and endpoint
api_key = "IHR_PERSÖNLICHER_API_SCHLÜSSEL"
base_url = "https://ki-toolbox.scc.kit.edu/api/v1"

# Initialize Client
client = OpenAI(api_key=api_key, base_url=base_url)

Try:
    # Send chat request
    chat_completion = client.chat.completions.create(
        model="azure.gpt-4.1-mini", # Select the appropriate model here
        messages=[
            {"role": "system", "content": "You are a helpful assistant at
KIT."},
            {"role": "user", "content": "Explain the principle of Rayleigh
scattering in three sentences."}
        ]
    )

    # Output answer
    print(chat_completion.choices[0].message.content)
```

```
except Exception as e:
    print(f"An error has occurred: {e}")
```

Use of the RAG system (knowledge base)

With Retrieval-Augmented Generation (RAG), you can provide the model with your own documents as a knowledge base for its answers.

Upload via the web interface (for easy starting)
For basic use and the upload of individual documents, you can use the web interface.

- **Where?** The upload area can be found under Tools > Knowledge.
- **What?** This method works well for typical file formats such as PDF, TXT, etc.

Upload via API and best practices (for advanced use)
For automation or the upload of larger volumes of documents, the use of the API is more convenient. The exact endpoints for this can be found in the interactive API documentation

- **API endpoints:** The specific endpoints for managing files and knowledge bases can be found in the interactive API documentation (https://ki-toolbox.scc.kit.edu/docs).
- **Data preparation recommendation:** To get the best results with RAG, it's a good idea to prepare your source documents in a clean, structured Markdown format before uploading them. This allows the model to interpret the content more reliably. Tools like Docling can be helpful in this conversion.

Info & Contact

License Notice

Imprint

Publisher: Karlsruhe Institute of Technology (KIT) Kaiserstraße 12 76131 Karlsruhe

Contact: InformatiKOM Adenauer Ring 12 76131 Karlsruhe Germany Tel.: +49 721 608-48200 E-mail: info@zml.kit.edu

Questions about the AI-Toolbox should be directed to: ki-toolbox@scc.kit.edu