Knowledge Service API

**Summary:**

Knowledge Search Service is an AI-powered service that enables you to build sophisticated knowledge bases that can be intelligently searched using APIs**.**

**API Documentation:**

Content for the documentation is available in **knowledge\_swagger.yaml** file.

**TIP:**  To get a better view of the documentation, copy the contents of the above yaml file in <https://editor.swagger.io/>. This will present the documentation in a neat UI.

NOTE: “Try it out” button in the swagger editor will not work as this document is just for information purpose.

**Usage of APIs:**

1. Create a knowledge base, which is basically a library for your FAQs.
2. Upload documents to the knowledge base.
3. Train a model on the content that will be indexed.
4. After that, all content in the knowledge base is indexed and can be searched for answers to end users' questions or keywords.
5. Search the knowledge base with your own questions. All search results returned to the user contain a confidence score.

## Knowledge bases

A knowledge base is basically a library of stored information (FAQs) that can be searched for answers to customers' questions. You first create a knowledge base so that you can add content to it later on.

Use the Knowledge APIs to accomplish any of the following:

* Get/view a single knowledge base
* Get/view a list of knowledge bases
* Create a knowledge base
* Update a knowledge base
* Delete a knowledge base

## Documents

Now that you have a knowledge base created, you can add content/documents to it. All FAQs must be constructed as fully-descriptive questions in order to effectively train a model and search the knowledge base.

Here are some examples:

* Effective: What is your favorite color?
* Ineffective: What is your favorite?
* Effective: How many coins do you have in your pocket?
* Ineffective: How many coins?

Use the Knowledge APIs to accomplish any of the following:

* Upload a single document
* Update and/or Upload multiple documents
* View all documents
* View a single document
* Delete a document

## Training

You train a model on the documents within a knowledge base. You can have multiple trained models - one for each locale and language (extracted from coreLanguage and language\_LOCALE defined in the knowledge base request) - within a knowledge base. Once a model is trained, the search queries return documents based on the latest trained model. Each answer matching the end user's query will be returned, and each will contain a confidence score.

Use the Knowledge APIs to accomplish any of the following:

* Train a model
* View a trained model
* View a list of trained models - Models are sorted by date in descending order, meaning the most recently trained model appears first.

## Knowledge base search

Use the Knowledge APIs to accomplish any of the following:

* Search the knowledge base

## Response codes

The following table defines all possible response codes.

|  |  |
| --- | --- |
| **Code** | **Description** |
| **200** | Success. |
| **400** | The request could not be understood by the server due to malformed syntax. |
| **401** | No authentication bearer token specified in authorization header. |
| **403** | You are not authorized to perform the requested action. |
| **404** | The requested resource was not found. |
| **415** | Unsupported media type - Unsupported or incorrect media type, such as incorrect Content type value in the header. |
| **429** | Rate limit exceeded the maximum [%s] requests within [%s] seconds. |
| **500** | The server encountered an unexpected condition that prevented it from fulfilling the request. |
| **503** | Service unavailable - the server is currently unavailable (because it is overloaded or down for maintenance). |
| **504** | The request timed out. |

**Important notes about Knowledge Base APIs:**

* Supported data format: JSON.
* Supported content: Text only. Rich media (images and video, for example) is not supported.
* Maximum number of allowable FAQs in the knowledge base: 500. Note: There is no minimum, but you need at least 5 documents in the knowledge base in order to effectively train a model and search the knowledge base.
* Maximum number of allowable knowledge bases per PureCloud organization: 5.
* Maximum allowable document file size: 10 KB.
* Maximum payload size for any request: 3 MB.

**FAQs of Knowledge Service API:**

1. **How to authenticate for using this API?**

Authentication token as part of request header (will be provided at the time of hackathon)

1. **What is knowledge base and what it should consists of?**

A knowledge base is basically a library of stored information (FAQs) that can be searched for answers to customers' questions. It should consist of single/multiple documents with FAQ

1. **Is Document a file consisting of FAQs?**

No, Document is not a file here. It is a JSON which will be used in body of the POST/PUT request.

1. **How should we add more FAQs to the Knowledge base?**

You should use the PUT request with the specified document id and should update more FAQs through the body of the request (Same as Create Document but here you are updating with the created document)

**Sample Request:** {{Knowledge\_Base\_Host}}/organizations/{{orgId}}/knowledgebases/{{knowledge\_base\_Id}}/languages/{{languageCode}}/documents/{{documentId}}

1. **How would you train and search with Knowledge base?**

With the train API, you should initiate the training and you could check the training status through GET training API with specific training id

With the search API, you could POST your question and it will be indexed in the Knowledge Base trained FAQs and the appropriate answer will be returned.

**Note:** The search question need not to be exactly same as trained FAQs. But it should be contextually same.

**Sample Request:**

1. Create Document:

NOTE: For this hackathon, Knowledge Service supports only document type, which is “Faq”

POST /knowledgebases/{{knowledge\_base\_id}}/languages/en-US/documents HTTP/1.1

Host: https://api.inintca.com/api/v2/knowledge

organizationId: {{org\_id}}

Content-Type: application/json

Authorization: Bearer {{access\_token}}

[

{

"externalUrl": "null",

"type": "Faq",

"faq": {

"question": " When do I receive the hard copy of the offer letter?",

"answer": "The hard copy of the offer letter is issued on the day of joining with Genesys. You will be required to attend the HR Orientation post which the offer letter will be issued.",

"alternatives": [

"Can I receive the hard copy of the offer letter before joining Genesys ?"

]

}

},

{

"externalUrl": "null",

"type": "Faq",

"faq": {

"question": "When do I receive ID Cards and Access Cards?",

"answer": "This can be done once your Genesys Employee ID is active. This will be ready within a couple of days of your joining. Please visit https://unifiedportal-mem.epfindia.gov.in/memberinterface/ for more details. A request has to be raised & you need to furnish your existing PF account number to facilitate this. ",

"alternatives": [

" Will my access cards be immediately activated ? "

]

}

}

]

1. Train Model

POST /knowledgebases/{{knowledge\_base\_id}}/languages/en-US/trainings HTTP/1.1

Host: https://api.inintca.com/api/v2/knowledge

organizationId: {{org\_id}}

Authorization: Bearer {{access\_token}}

1. Search Question

POST /knowledgebases/{{knowledge\_base\_id}}/search HTTP/1.1

Host: https://api.inintca.com/api/v2/knowledge

Content-Type: application/json

Authorization: Bearer {{access\_token}}

:

{

"query": "When will I get ID card?",

"pageSize": 1,

"pageNumber": 1,

"languageCode":"en-US",

"documentType": "Faq",

"searchOnDraftDocuments": "True"

}

**Glossary**

1. [API](https://en.wikipedia.org/wiki/Application_programming_interface) – Application Program Interface –interface between various parts of the computer program.
2. [REST](https://en.wikipedia.org/wiki/Representational_state_transfer) – Representational State Transfer – a way of developing and exposing web services to use leverage HTTP methods like GET, POST, PUT and DELETE. It simplifies the structure of the web service than using SOAP (Simple Object Access Protocol)
3. Payload – The JSON/XML input parameter that shall be sent for the service. This usually are required for POST method calls
4. Model – Model or pretrained model - is the synonym of the trained system (comprises of the algorithm along with the stored knowledge obtained in the training)
5. Entity/ Tag – Entities or tags are the elements of any sentence that represents some specific attribute in the sentence. For example, the string “Ram wants to book a flight from Chennai to Mumbai on 5th March”, **Ram** shall be identified as **PERSON, Chennai** and **Mumbai** is tagged as **GEO** and **5th March** is automatically converted to the respective timestamp.