DentalMammoth

DM API Overview

Introduction

DM Web API is a collection of web services that provides access to DentalMammoth clinical contents. This document describes DM Web API and how it is used to request data from DentalMammoth and integrate user information from the third party systems.

Technical solution

API resources are made accessible through an HTTPS-based RESTful API. DM API supports the HTTPS request method GET and POST.

The client creates new entries by issuing POST requests. The details of an individual entry or list of entries is retrieved using a GET request.

The supported response format is JSON. Set the HTTP request header Accept to media type application/json to specify the response format.

API base addresses

https://api.dentalmammoth.com/

Response codes

Status codes.

Response code	Description
200	The API call was ok
401	Unauthorized. The given credentials was not valid or the token was not valid or is expired
403	Forbidden resource. The caller was not able to access the resource with the given token.
404	The asked resource was not found.
500	Some other error occurred. In this case the API returns the error message which contains the error code and detailed error message.

Error codes and messages

Code	Message
1000	Unknown user
1001	User already exists
2000	Unknown database
2001	Unknown article
2002	Unknown browse item
2003	Search did not return any articles
9000	Unknown error: <error message>

Authentication and authorization

The DM API uses token based authentication. Each request must contain the valid token which is added as Bearer token in the Authorization request header.

If authentication fails the server will respond with HTTP status 401 “Unauthorized” or 403 “forbidden” if the token does not have an access to requested resource or it is expired.

How to get token?

To fetch the API token one should have the valid credentials (username and password). These credentials are sent to the login endpoint by using the proper POST message.

Endpoint address
<base address>/auth/login

Method
POST

Request message

{“username”: ”your-username, “password”: ”your-password”}

Response message
The response is either the valid token or status code 401 if credentials are not valid.

```
{
“token”:“eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJqYWhhIiwiZXhwIjoxNTUz
MTAyMjQxfQ.8tFxkt9XNAJuQgx56zhz5F7XiuHNRDbV8d2b3abLfr3v3TqfgtogCohkrzW6V6R
daNXxIm_AhznMNPK9WUtoa”
}
```

Content API

Provides access to the DentalMammoth Clinical database contents.

List of databases

Fetch the list of all Clinical Databases which the given token has access to.

Endpoint address
<base address>/databases

Method
GET

Response message
Response message contains the array of database objects which the given token has access to or error message if error occurs.

```
[
  {
      "id": "tba",
      "name": "Dental database",
      "shortName": "Dental db",
      "lang": "en",
      "browseId": 456,
      "authentication": null,
      "defaultArticle": "toebd_info",
      "linkName": "dental_database_therapia_odontologica",
      "sortType": "sortkey",
      "free": false
  },
  {
      "id": "hba",
      "name": "Dental Procedures",
      "shortName": null,
      "lang": "en",
      "browseId": 452,
      "authentication": null,
      "defaultArticle": "toebd_info",
      "linkName": "dental_procedures",
      "sortType": "sortkey",
      "free": false
  },
…
…
]
```

Database article search

Search the Clinical database content by given keyword and or database identifier.

Endpoint address
<base address>/databases/search/{page}

Method
POST

URL parameters

- page: controls the fetch result page number.

Request message
The request message can contain the keyword and/or database fields. Keyword is used to fetch the articles and database is used to filter the result to only specific database.

```
{
    “keyword”: search keyword,
    “database”: database identifier
}
```

Response message
Response message contains the array of article objects which fulfill the given search criteria. Response also contains the total number of hits and result page object.

```
{
    "totalHits": 184,
    "page": {
        "totalPages": 10,
        "currentPage": 4,
        "pageSize": 20,
        "startIndex": 60
    },
    "results": [
        {
            "databaseName": "Dental database",
            "companyName": null,
            "language": "en",
            "createdStr": "01.01.2012",
            "title": "Cystic fibrosis (fibrocystic disease; mucoviscidosis) (E84)",
            "url": "cystic_fibrosis",
            "objectType": "ARTICLE",
            "database": "tba",
            "companyId": null,
            "rate": null,
                        "letter": null,
                        "logo": null,
                        "id": "tba31043",
                        "objectId": "tba31043",
                        "status": 1
                },
                {
                        "databaseName": "Dental database",
                        "companyName": null,
                        "language": "en",
                        "createdStr": "01.01.2000",
                        "title": "Bleeding from a tooth",
                        "url": "bleeding_from_a_tooth",
                        "objectType": "ARTICLE",
                        "database": "tba",
                        "companyId": null,
                        "rate": null,
                        "letter": null,
                        "logo": null,
                        "id": "tba01180",
                        "objectId": "tba01180",
                        "status": 1
                },
                …
                …
        ]
}
```

Article information

Fetch the given article content and metadata. The proper article id’s can be found from the search response message. The article text is returned in html format.

Endpoint address
<base address>/databases/article/{articleId}

Method
GET

URL parameters

- articleId: The id of the article to fetch

Response message
Response message contains the article content in HTML format and article metadata.

```
{
        "id": "tbx05005",
        "title": "Dentine caries on approximal surfaces, occlusal incipient dentine caries ",
        "text": "<article><section><h1>Dentine caries on approximal surfaces, occlusal
incipient dentine caries </h1><figure id=\"F1\"><img id=\"I1\"
src=\"https://s3-eu-west-1.amazonaws.com/resources.dentalmammoth.cloud/content/tbx/5.2
7.jpg\"/><legend><p>In d. 85 dentine caries is found on both approximal surfaces. On the
occlusal surface of d. 46,  incipient dentine caries is displayed; mesial enamel caries is also
revealed.</p></legend></figure></section></article>",
        "database": "tbx",
        "language": "en",
        "sortkey": null,
        "created": 1311886800000,
        "createdBy": null,
        "updated": 1403816400000,
        "updatedBy": null,
        "published": 1403816400000,
        "authors": "",
        "description": null,
        "metaTitle": null,
        "linkName":
"dentine_caries_on_approximal_surfaces_occlusal_incipient_dentine_caries_",
        "updateStatus": 0,
        "navigation": null,
         "keywords": null,
        "articleDatabase": null,
        "publishedStr": "27.06.2014",
        "sites": null,
        "access": null,
        "copyright": null,
        "articleSource": null
}
```

Database browse

Fetch the hierarchical list of sections and articles for the selected database.

Endpoint address
<base address>/databases/browse/{database}/{browseId}

Method
GET

URL parameters

- database: The id of the database to browse
- browseId (optional): The id of the database browse section

Response message

Response message contains the array of browse sections and/or articles.

```
{
        "id": "4",
        "parentId": "0",
        "title": "Dental Pictures",
        "linkName": null,
        "sortOrder": 0,
        "level": 0,
        "leaf": false,
        "children": [
                {
                        "id": "1555",
                        "parentId": null,
                        "title": "Oral pathology",
                        "linkName": null,
                        "sortOrder": 6,
                        "level": 0,
                        "leaf": false,
                        "children": null,
                        "articles": null,
                        "article": null
                },
                {
                        "id": "1560",
                        "parentId": null,
                        "title": "Aesthetic dentistry",
                        "linkName": null,
                        "sortOrder": 13,
                        "level": 0,
                        "leaf": false,
                        "children": null,
                        "articles": null,
                        "article": null
                },
                …
                …
        ]
}
```

DM Article view page

DM API offers also an easy way to show the properly formatted Clinical Database articles.

After the search request you can use the article ids from response message and direct users to address

- https://www.dentalmammoth.com/en-us/article/{articleId} (in production environment)
- http://beta.dentalmammoth.com/en-us/article/{articleID} (in test environment)

These urls offer the simple DentalMammoth view to the given article.

User API

Provides the simple API to integrate third party users to DentalMammoth and offers the simple single sign-on system to DentalMammoth site.

Create a new user

Creates a new user to DentalMammoth user database.

Endpoint address
<base address>/user

Method
POST

Request messages

```
{
	"remoteId": Third party system user id, (*)
	"name": User full name,
	"username": username,
	"email": user email address, (*)
	"lang": language code in ISO 639-1 format, (*)
	"city":   the name of the city,
	"country": country code in ISO Alpha 2 format
}
```

(*) mandatory fields
DentalMammoth creates a new user with the proper privileges.

Response message
Response message contains the given user information and generated, disposable, session token which can be used to login user to DentalMammoth site (see: **Login with session token)** or error message if error occurs.

```
{
	"remoteId": Third party system user id,
	"name": User full name,
	"username": username,
	"email": user email address,
	"lang": language code in ISO 639-1 format,
	"city":   the name of the city,
	"country": country code in ISO Alpha 2 format,
	“sessionToken”: the disposable session token
}
```

Update existing user information

Updates existing user information to DentalMammoth user database.

Endpoint address
<base address>/user

Method
PUT

Request messages

```
{
        "remoteId": Third party system user id,
        "name": User full name,
        "username": username,
        "email": user email address,
        "lang": language code in ISO 639-1 format,
        "city":   the name of the city,
        "country": country code in ISO Alpha 2 format
}
```

The remoteId field must be the id of the existing DentalMammoth user.

Response message
Response message contains the given user information and and generated, disposable, session token which can be used to login user to DentalMammoth site (see: **Login with session token)** or error message if error occurs.

```
{
        "remoteId": Third party system user id,
        "name": User full name,
        "username": username,
        "email": user email address,
        "lang": language code in ISO 639-1 format,
        "city":   the name of the city,
        "country": country code in ISO Alpha 2 format,
        “sessionToken”: the disposable session token
}
```

Fetch user information

Fetch the existing user information from DentalMammoth user database.

Endpoint address
<base address>/user/{remoteId}

Method
GET

URL parameters

- remoteId: The user id whose information is being retrieved

Response message
Response message contains the given user information and generated, disposable, session token which can be used to login user to DentalMammoth site, (see: **Login with session token)** or error message if error occurs.

```
{
        "remoteId": Third party system user id,
        "name": User full name,
        "username": username,
        "email": user email address,
        "lang": language code in ISO 639-1 format,
        "city":   the name of the city,
        "country": country code in ISO Alpha 2 format,
        “sessionToken”: the disposable session token
}
```

Delete user information

Removes the existing user information from DentalMammoth user database.

Endpoint address
<base address>/user/{remoteId}

Method
DELETE

URL parameters

- remoteId: The user id whose information will be deleted

Response message
Returns the status code 200 if delete success or error message if error occurs.

Fetch user specific session token

Returns the user specific disposable session token which can be used to login user to DentalMammoth site, see: **Login with session token**

Endpoint address
<base address>/user/session/{remoteId}

Method
GET

URL parameters

- remoteId: The user id whose session will be retrieved

Response message
Returns the session token or error message if error occurs.

```
{
        “sessionToken”: the disposable session token
}
```

Login with disposable session token

DentalMammoth API offers an easy way to log in end users to DentalMammoth site without entering the login credentials.

1. create a new user to DentalMammoth using the **/user** endpoint
2. fetch new disposable session token using the **/user/session/:userId** endpoint
3. redirect user to link https://www.dentalmammoth.com/en-us/sessionLogin?session=<user disposable session token>

Step one can be skipped if the user already exists.
Session token is disposable which means that it’s valid until the user logs in by using the link.