

XML to PubMed format transformation

Information about how to run code

Quick overview

This project represents application that provides uploading, downloading and transforming one type of input file in other. Users that want to use given API should be authorized previously through JWT authentication mechanism.

Technologies required

Java 17, Maven 3.8.5, Java Spring Boot 2.6.4

Tools used

IntelliJ Ultimate IDEA 2021.3.3, Postman 7.36.7

How to run code

In order to run application, you would need initialized Java Spring Boot project where you can import written files. Project could be found on GitHub repository: <https://github.com/hkajevic/XMLtoPubmedConverter>

You should follow one of these ways:

First way (recommended):

- Open IntelliJ IDEA and under tab File choose Open and select directory where project is unzipped
- Select OK and project should be imported correctly

Second way:

- Open the Spring Initializr <https://start.spring.io/> and select mentioned Java Spring Boot version.
- Provide the group name net.cyrillicsoftware and artefact id xml-to-pubmed-converter
- Generate project and make sure to find downloaded file on your local computer
- Unzip downloaded file and lunch IntelliJ IDEA
- Import file in IntelliJ IDEA through File -> Import -> Existing Maven Projects -> Next -> Browse -> Select the folder xml-to-pubmed-converter-> Select Folder -> Finish
- Follow \xml-to-pubmed-converter\xml-to-pubmed-converter path and copy all project files and replace existing if needed

After importing project navigate to pom.xml file and if not loaded automatically then loaded existing dependencies into your project. Additionally navigate to the Paths class and change paths to the provided files (right click on package and select Copy Path/Reference where you can get absolute path to the files).

Run your application and it should start on Tomcat server on port 8080. Eventually if you want to change server port you can do it by adding server-port: PORT_NUMBER command in application.properties file (PORT_NUMBER represents specific port that you want to use and in future API calls instead :8080 that port number should be used).

You can find remaining information on your browser URL <http://localhost:8080/swagger-ui/> and request could be sent through that API or using Postmen.

Make sure you are firstly authorized through <http://localhost:8080/authenticate> URL before sending other requests.

API provides several endpoints:

- <http://localhost:8080/authenticate> - returns token required for other API calls
- <http://localhost:8080/req> - checks if application server is available
- <http://localhost:8080/upload> - provides xml file upload (file should be given as parameter "file")
- <http://localhost:8080/download/{fileId}> – allows you to download file on your local machine (file id should be provided)
- <http://localhost:8080/transform/{fileId}> – transforms xml file in proper pubmed format (file id should be provided)
- http://localhost:8080/upload_transform_download - merges previous functionalities into one and provides uploading file, transforming it and downloading transformation result

Make sure that while sending all requests that application is running.

For additional specification please make sure you checked on swagger documentation provided through above mentioned link and feel free to contact creator if something is not working as specified.