**Tasks to perform-**

Create a service that searches a pre-configured directory and subdirectories for files containing a list of users provided words-

The service should:

• Present a simple rest API to allow perform a search

• Allow searching with multiple words

• Multiple word searches should return files that match all words

• Should perform whole word matches

• No fuzzy search required

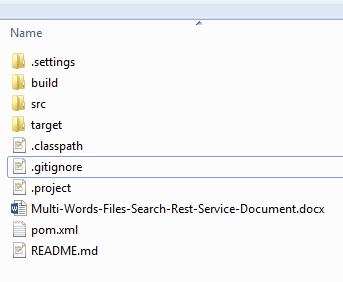
• Return results as quickly as possible

• Code should be production ready code to a professional standard

**Technologies used-**

* Swagger API
* Spring BOOT
* Java version: 1.8
* Build tool: Maven

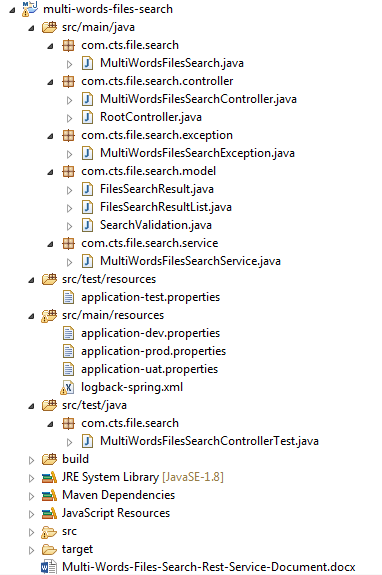
**Code Directory Structure-**



src code folder contains all the source code

pom.xml is the build file

**Main components-**

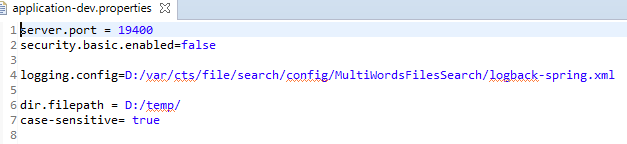


**Salient features of program**—

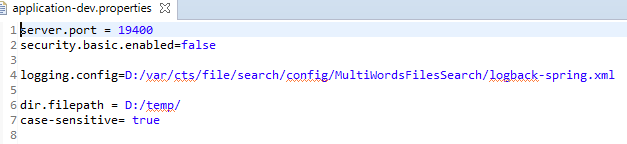
* MultiWordsFileseSearchService has been implemented using advance java features i.e. – File NIO, Java8 – lambda expression, parallel stream, Fork – Join Pool Framework etc.
* Rest Contract Client UI has been implemented using Swagger API

**Some Important Parameters-**

* Open the application-dev. properties file and change the following parameters as per your requirements.



* Directory search: By default application search all files present under root or sub directories.
* Setting case sensitive or insensitive search: Change the “case-sensitive” parameter to false to make the search case insensitive.



* Search logic is implemented only for txt file. Further implementation can be taken care of during enhancement.

**How to run-**

Pre-Requisite: JDK 1.8 and Maven needs to be installed.

Go to Path …multi-words-files-search\target\generated-jar

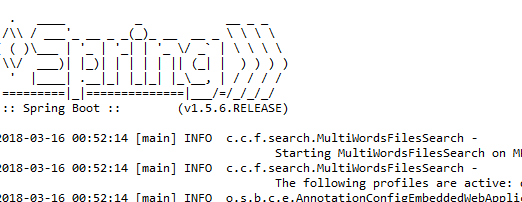
**Run below command –**

Java -jar -Dspring.profiles.active=dev -Dspring.config.location=${dir where application-dev.properties is placed} MultiWordsFilesSearch-1.0\_A0.jar

Alternatively, you can also run using – **mvn spring-boot:run** command from the project root directory.

This will start the application at given port in application-dev. properties (server. port), in above example, it’s 19400.

You will get similar message as below when it is running.



Now open the browser and give the following URL.

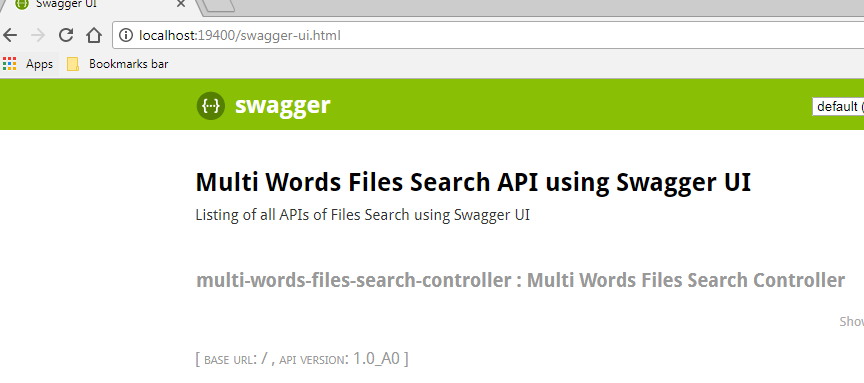
<http://localhost:19400/swagger-ui.html>

You will get a page of Swagger which gives a detailed specification of API with provision to test it out.

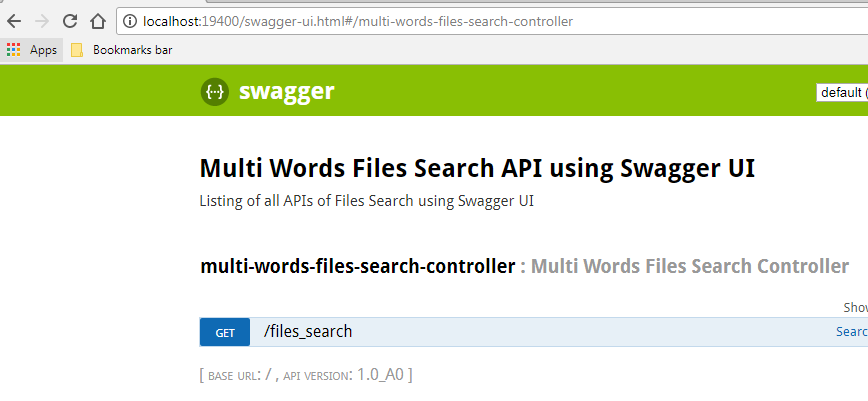
A sample run can be as follows.

**Run below URL –**

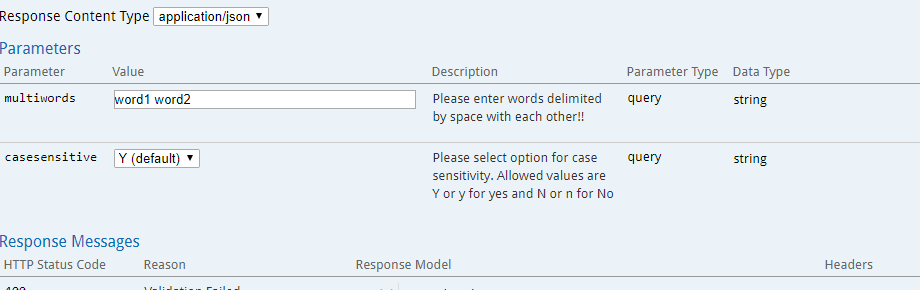
<http://localhost:19400/swagger-ui.html>



Click on multi-words-files-search-controller hyperlink-



Click on GET-



Enter words to search e.g. – word1 word2 , select casesensitive- Y and Response Content Type – json/xml and run it using “Try it Out” button



**Test Cases Run Status –**

I have also created a set of Junit test cases under MultiWordsFilesSearchControllerTest.

Sample test run is as shown below.

