Task to do

=====================================================================================

Create a service that searches a configured directory and subdirectories for files containing a list of words that are provided to the service.

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

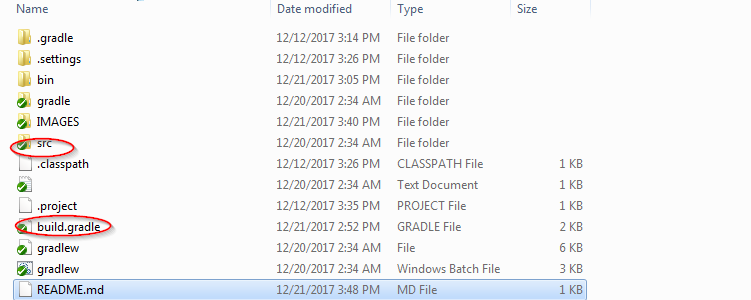
• Code should be production ready code to a professional standard

=====================================================================================

Technology used:

* Spring BOOT
* Java version: 1.8
* Build tool: Gradle

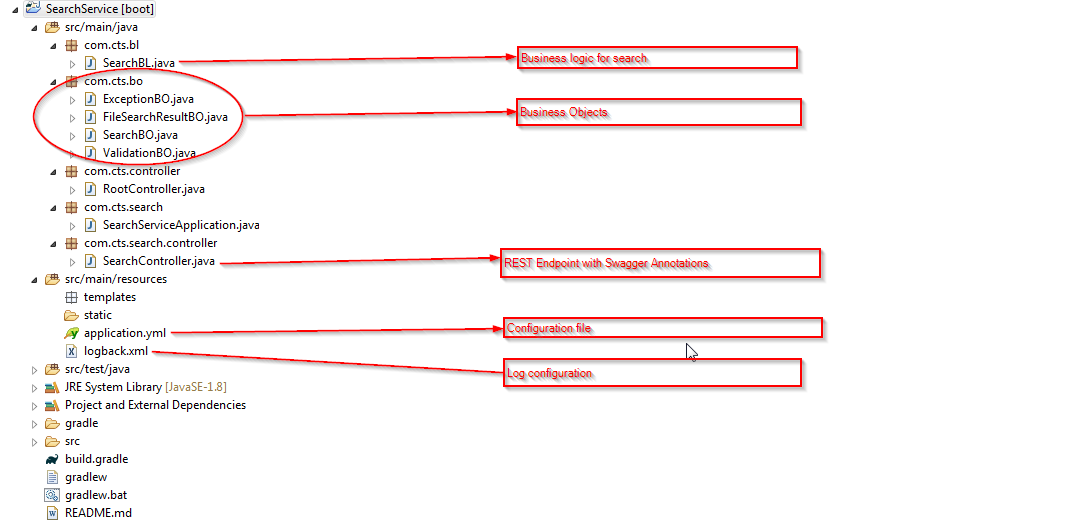
Code Directory Structure



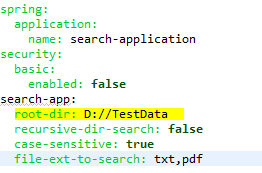
Src code folder contains all the source code

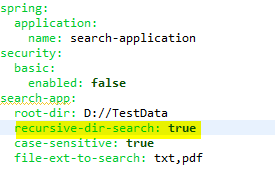
Build.gradle is the build file

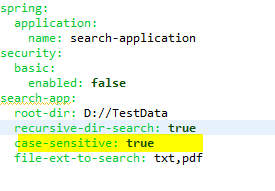
Main components



Some Important Parameters

* Setting the base directory to search: Open the application.yml file and change the following. 
* Setting the recursive directory search: By default the application only serach for all files under the root directory and it’s subdirectory. To make it only limited to root directory change the following parameter to false



* Setting case sensitive or insensitive search: Change the following parameter to false to make the search case insensitive. 
* By default txt files and pdf files will be picked but search logic is implemented only for txt file.

How to run:

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

Go to the directory containing the build.gradle from command prompt and issue the command.

gradle bootRun

This will download all required files and build the application and fire the application after successful build.

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

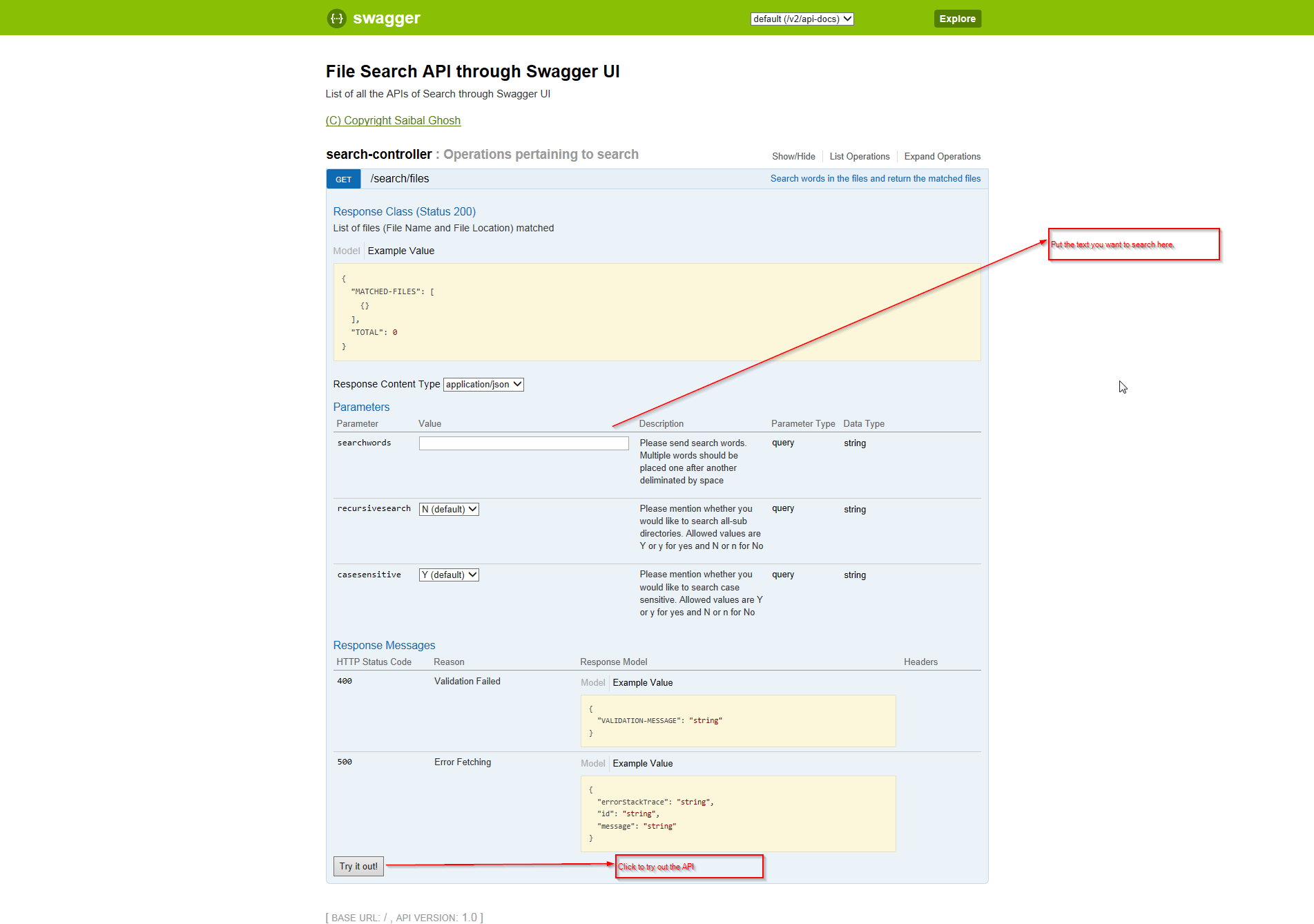
Now open the browser and give the following URL.

<http://localhost:8080/swagger-ui.html>

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

Sample URL to test out the service (change the highlighted word with the one you would like to search.

http://localhost:8080/search/files?searchwords=*Saibal%20kumar*&recursivesearch=Y&casesensitive=Y



A sample run can be as follows.

