Machine coding round (1 Week)

Design and build an application that can search documents from a cloud storage service like Dropbox or
Google Drive on the content inside the document.

Programming languages to be used: Python/Java

Please keep these considerations in mind as you go about solving the problem:

1. Implement the solution as described below. Do your best to reflect your design and coding ability.
2. Implement clean modular code respecting OOPs (object oriented programming) paradigms.
3. The application will be demoed during the interview.
4. High level design digram of your solution. You can use tool attache as image or PDF.
5. Add proper README file for execution and installation steps and others important aspects.

In this coding assessment you will develop a basic search service for data stored in online storage services like
Google Drive, or DropBox. You can upload the code to github(preferred) or upload the project as a zip/tarball and
share by email/file sharing service.

Requirements:
1. Connect to an online data storage service of your choice (Google Drive, Dropbox, S3, Azure blob storage service of your choice) and use the provided APIs to fetch the files stored by the service, e.g.
    a. https://dropbox.github.io/dropbox-api-v2-explorer/.
    b. https://developers.google.com/drive/api/guides/search-files

2. You can have files that are in either .csv, .txt, .pdf or .png format. (Bonus : For the files which are not utf-8 encoded, you may choose to extract the text content from them using a library of your choice like Apache Tika(available for Java/python):
https://tika.apache.org/. Or tesseract (available for Java/python) (https://github.com/tesseract-ocr/tesseract)

3. Index the content within the files to provide maximum-text search capabilities on either the data of the file or the meta information of the file. You can use Elasticsearch / Postgres or any storage you think is suitable for this:
    a. https://github.com/elastic/elasticsearch#readme.
    b. https://github.com/postgres/postgres.

4. You will provide an API that takes a search term/token as input and returns a list of files and their HTTP
URLs that contain the term in their content, e.g.

curl http://<search-service-host>/search?q=”@mail.com”

5. You will provide a basic command line or postman that consumes the above API and displays the files matching the given query.

E.g
files in storage with content below

FilePath	Content
X../File1	a,b,c,d,e
Y../File2	c,d,e
X../File3	g,h

input 1 -> curl https://<search-service-host>/search?q=”c”
output -> X../File1
Y../File2

input2 -> curl https://<search-service-host>/search?q=”notfound-term”
output -> Empty

Delete file1 at source(cloud storage)
input 3 -> curl https://<search-service-host>/search?q=”c”
output -> Y../File2

Bonus Points Criteria

1. **Uses of AI Assistant**:
   - Candidates are allowed to use any AI assistant to help write their code.
   - They must include the exact prompts they used to generate the code.
2. **Code Quality**:
   - The code should be well-structured and modular follow OOPs (object-oriented programming) paradigms.
   - It should follow best practices for readability and maintainability.