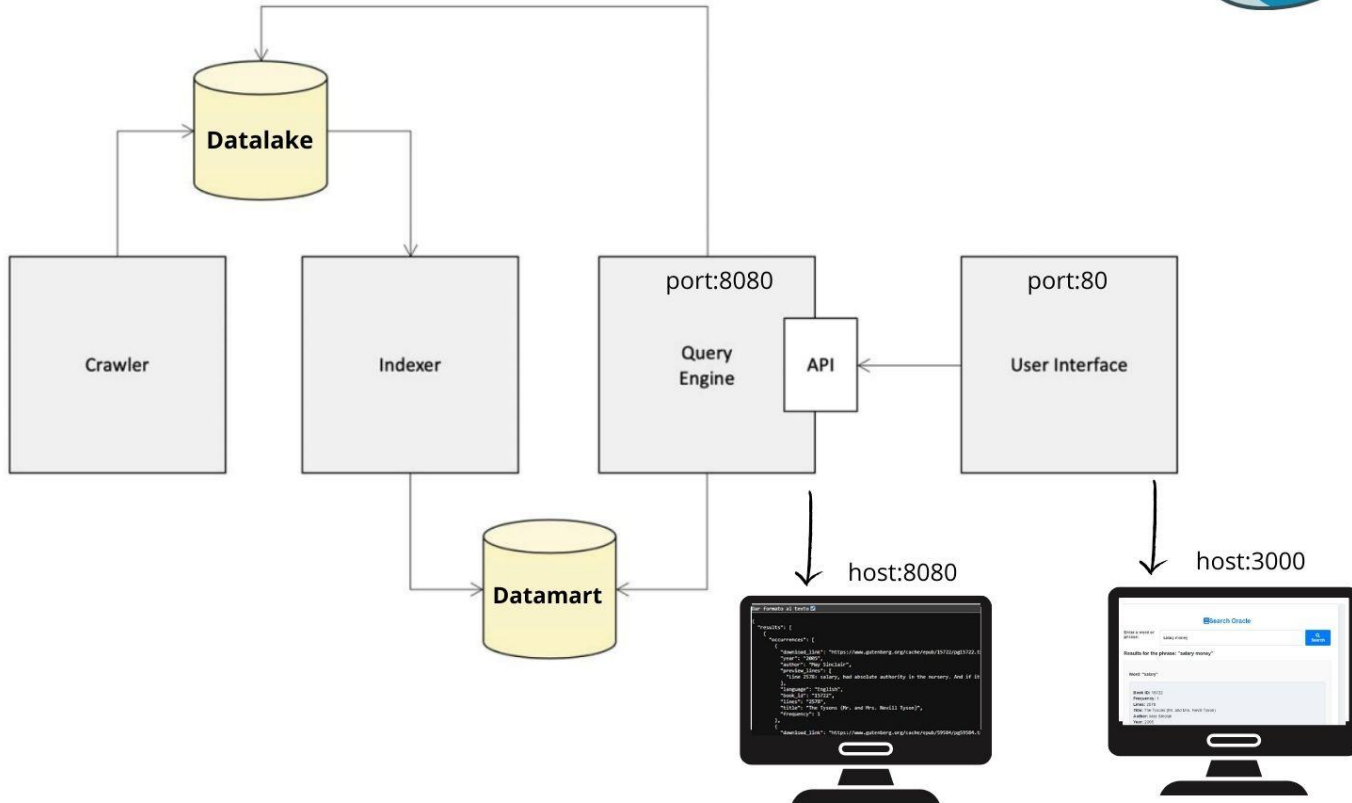# Search Engine

Part 2: Professionalization of the Project with Java Code

Search Oracle:
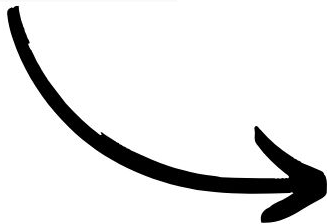
Álvaro Rodríguez González, Lucía Afonso Medina, Alejandro Alemán Alemán,
Farid Sánchez Belmadi, Néstor Ortega Pérez.

Docker

Datalake

Crawler

Indexer

Datamart

port:8080

Query
Engine

API

port:80

User Interface

host:8080

host:3000

Connection between API and GUI

```
GET http://localhost:8080/search?phrase=salary%20money
```

```json
{
  "results": [
    {
      "occurrences": [
        {
          "download_link": "https://www.gutenberg.org/cache/epub/15722/pg15722.txt",
          "year": "2005",
          "author": "May Sinclair",
          "preview_lines": [
            "Line 2578: salary, had absolute authority in the nursery. And if it had been"
          ],
          "language": "English",
          "book_id": "15722",
          "lines": "2578",
          "title": "The Tysons (Mr. and Mrs. Nevill Tyson)",
          "frequency": 1
        },
        {
          "download_link": "https://www.gutenberg.org/cache/epub/59584/pg59584.txt",
          "year": "2019",
          "author": "Sir Richard Francis Burton",
          "preview_lines": [
            "Line 4310: the governor's salary will be dependent upon the Crown, and will not be",
            "Line 4656: to civilians.[156] The salary of this high official was $500 in 1772; it",
            "Line 4750: only salary. The functions of the Amtmenn, Sýslumenn, and of the"
          ],
          "language": "English",
          "book_id": "59584",
          "lines": "4656, 6496, 15170, 12963, 6132, 4310, 6137, 6523, 4750, 6175, 6367, 8223",
          "title": "Ultima Thule; or, A Summer in Iceland. vol. 1/2",
          "frequency": 12
        },
        {
          "download_link": "https://www.gutenberg.org/cache/epub/72041/pg72041.txt",
          "year": "2023",
          "author": "William Le Queux",
          "preview_lines": [
            "Line 5440: acquaintance had told him, and she would get a decent salary from her"
          ],
          "language": "English",
          "book_id": "72041",
          "lines": "5440",
          "title": "A woman's debt",
          "frequency": 1
        }
      ],
      "word": "salary"
    },
    {
      "occurrences": [
        {
```

hello

hola

salut
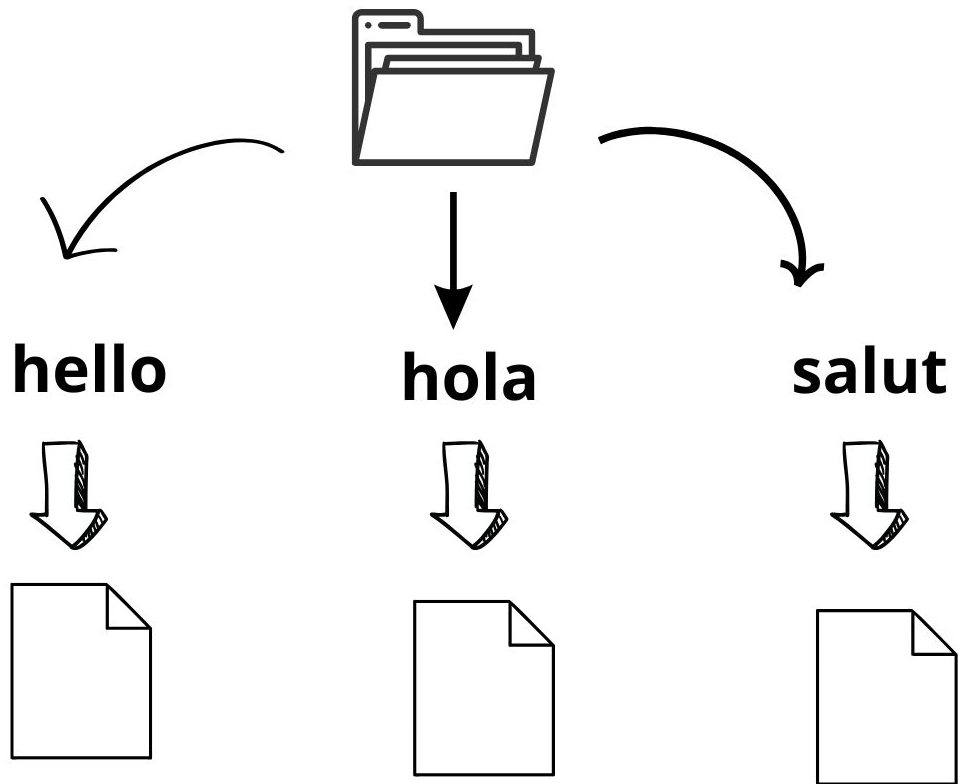
Serialization

One Word per File

```
words/acquisitions
1    72957 4012
2    20955 7803
3    |
```

One File

```
words
 1    pioneers
 2    - 65466 405
 3    resume
 4    - 41594 1872 7345
 5    fraternize
 6    - 41594 3526
 7    artistic
 8    - 65466 1200 371 724 710 727 1036 669 1150 1502 1167
 9    spreading
10    - 41594 5414
11    gingerbread
12    - 61074 1009 962 2891 1437
13    amusing
14    - 41594 7222
15    enchantment
```
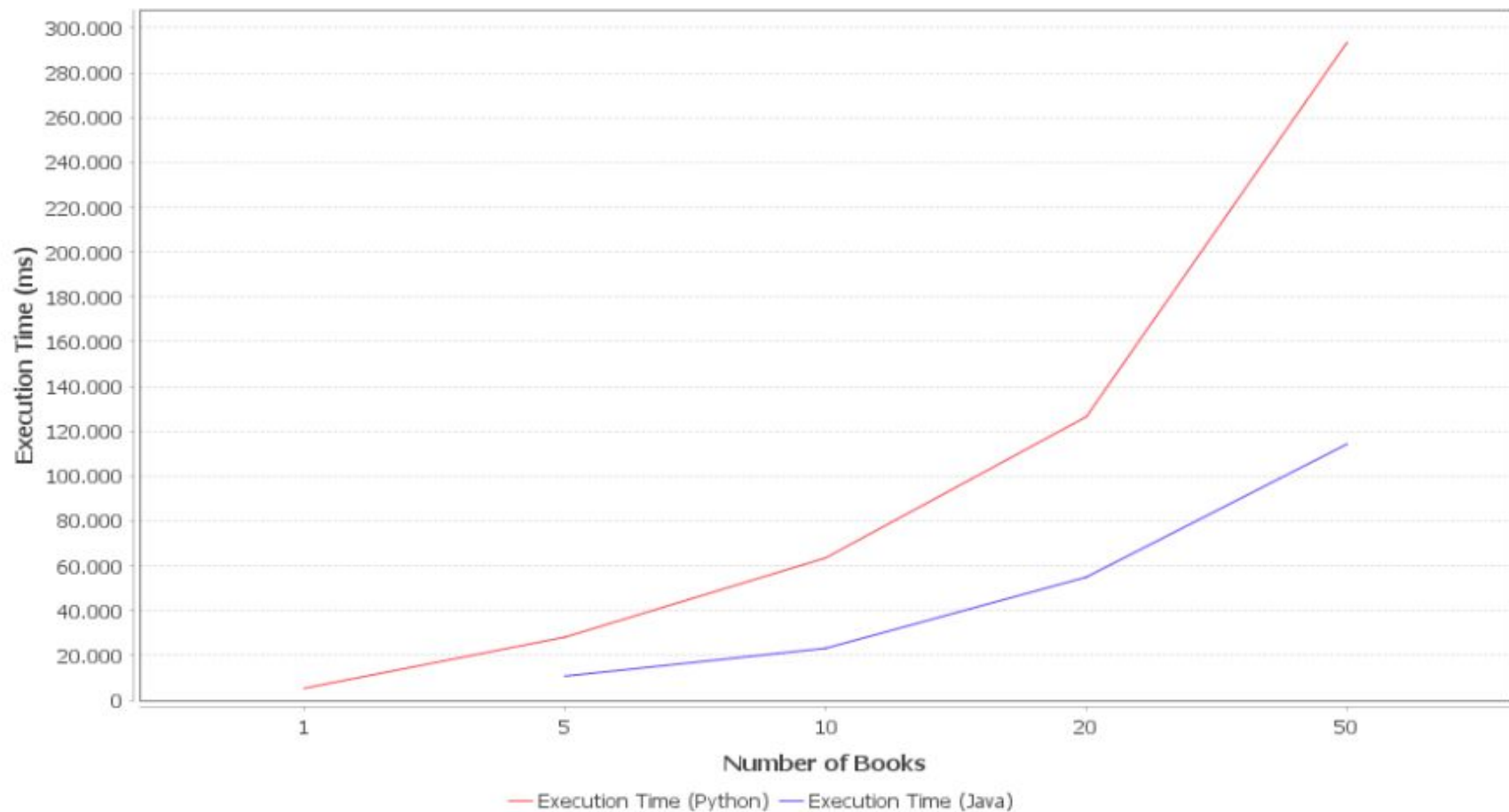
Metadata

```
metadata/metadata

 1    56028
 2    The Romance of the Harem
 3    Anna Harriette Leonowens
 4    2017
 5    English
 6    https://www.gutenberg.org/cache/epub/56028/pg56028.txt
 7    72957
 8    The aristocrats
 9    Gertrude Franklin Horn Atherton
10    2024
11    English
12    https://www.gutenberg.org/cache/epub/72957/pg72957.txt
13    8585
14    Roughing It, Part 4.
15    Mark Twain
```
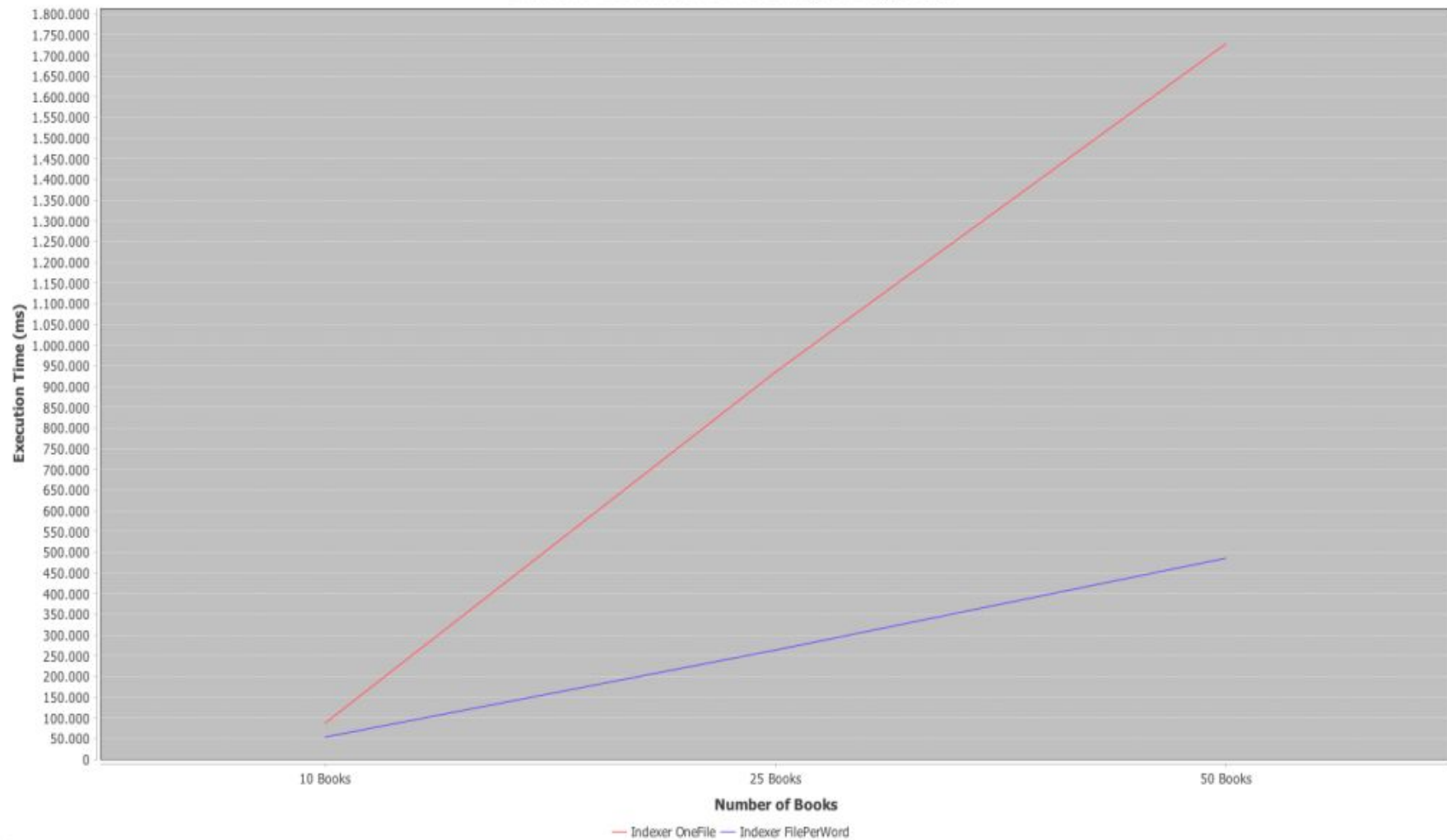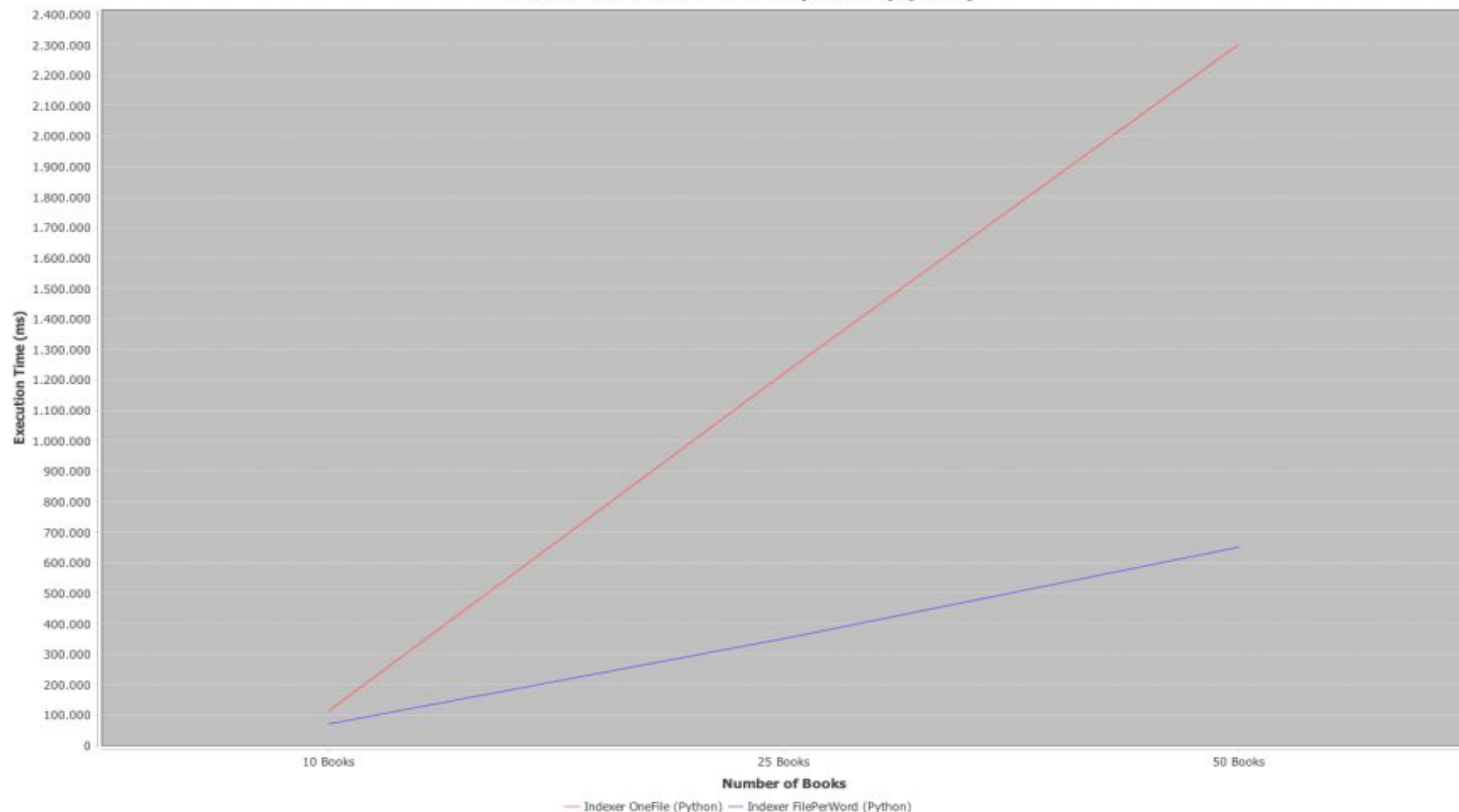
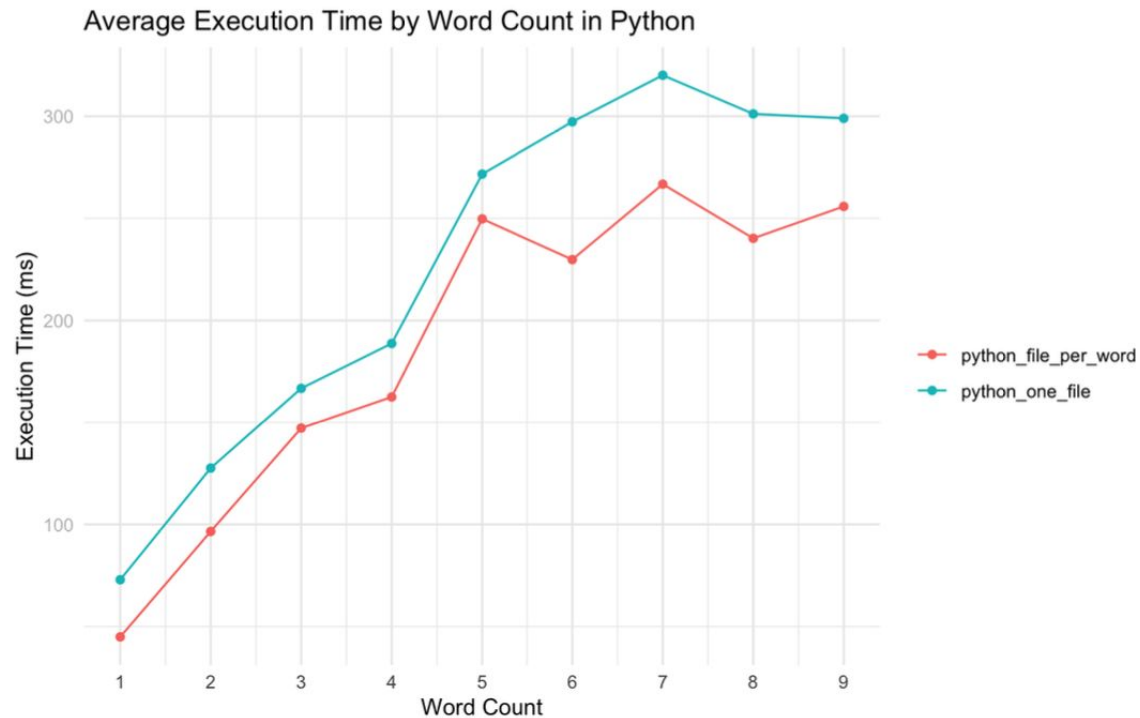# Interfaces
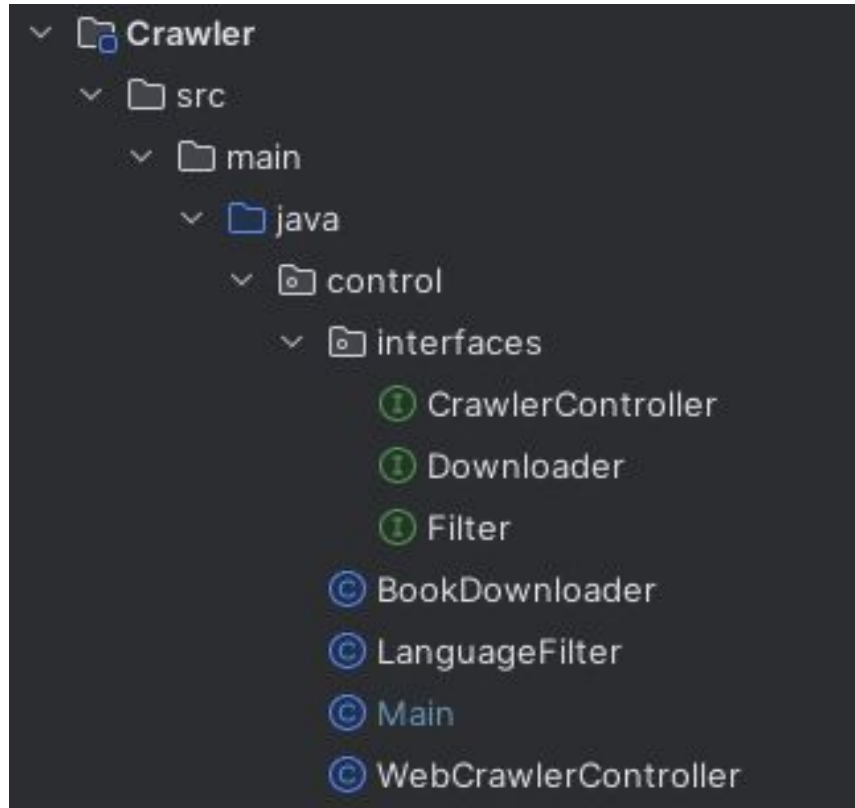
Crawler Execution Time Comparison: Java vs Python

Execution Time (ms) vs Number of Books

— Execution Time (Python) — Execution Time (Java)

Indexer Execution Time Comparison(Java)

Execution Time (ms)

1.800.000
1.750.000
1.700.000
1.650.000
1.600.000
1.550.000
1.500.000
1.450.000
1.400.000
1.350.000
1.300.000
1.250.000
1.200.000
1.150.000
1.100.000
1.050.000
1.000.000
950.000
900.000
850.000
800.000
750.000
700.000
650.000
600.000
550.000
500.000
450.000
400.000
350.000
300.000
250.000
200.000
150.000
100.000
50.000
0

10 Books   25 Books   50 Books

Number of Books

— Indexer OneFile — Indexer FilePerWord

Indexer Execution Time Comparison (Python)

Chart plotting Execution Time (ms) on the y-axis (0 to 2.400.000) versus Number of Books on the x-axis (10 Books, 25 Books, 50 Books).

Legend: Indexer OneFile (Python) — Indexer FilePerWord (Python)

Query Engine Benchmarking

Average Execution Time by Word Count in Python

Query Engine Benchmarking

Average Execution Time by Word Count in Java
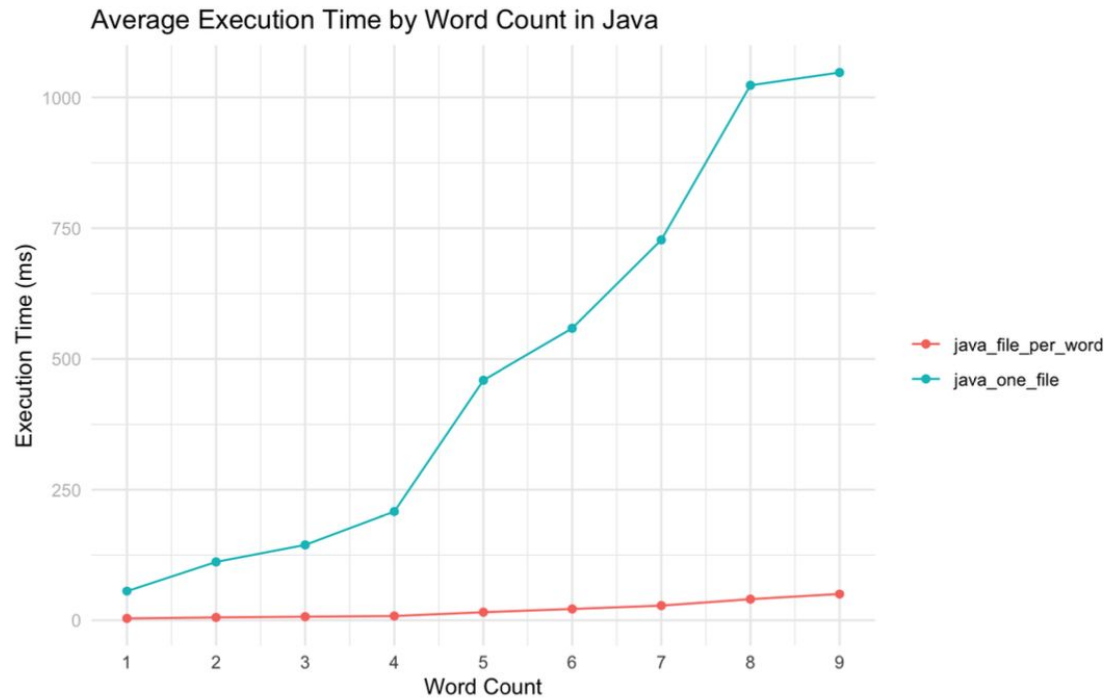
Conclusions

- Professionalized code implemented in Java
- Interface-based design
- Implementation of Parallelization
- New Graphical User Interface module
- API implemented
- New serialization format
- Docker execution