Web Search Engine

Instructor – Dr. Ikjot Saini

TA - Rahul Raveendran, Rishav Chaterjee

Group-5
- Srishti Jain (110026562)
- Margaret Arulmalar Rebeka (110026527)
- Siddharth M. Paliwal (110036256)

INTRODUCTION

Problem Statement

Develop a web search engine using following concepts from class:

1. Web Crawler
2. HTML to TEXT using JSOUP
3. Pattern Matching using TST
4. Page Ranking
5. Caching

Roles & Responsibilities

1. Siddharth
   - Web Crawler
   - Database Integration
   - Web integration
2. Margaret
   - HTML to TEXT using JSOUP
   - Page Parser
   - Integration between classes
3. Srishti
   - Pattern Matching
   - Page Ranking
   - Latency Optimization

Research

Feature Specific

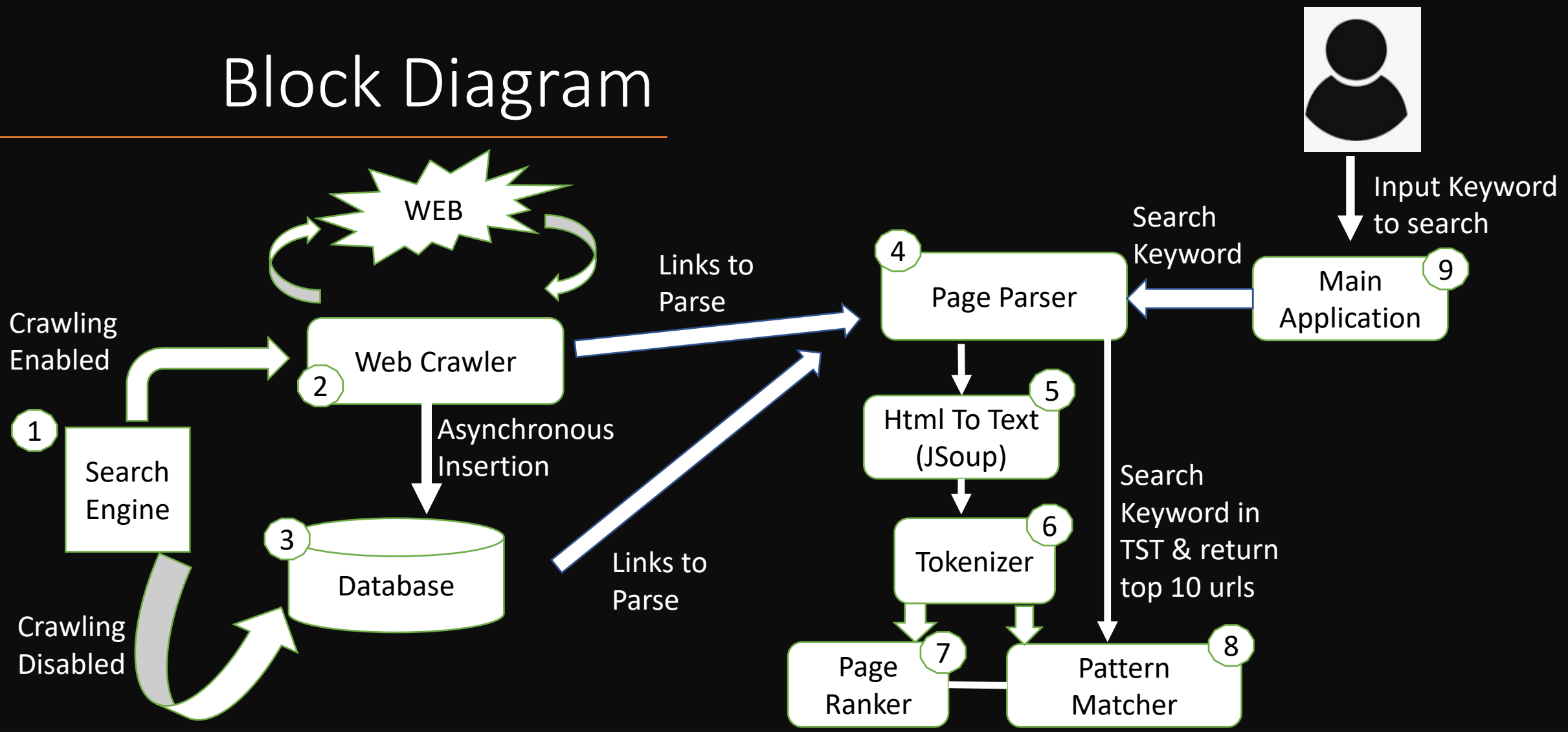
1. Page Ranking
2. Pattern Matching

Latency Optimization

1. Incorporating Database
2. Multithreading

Methodologies

Block Diagram



Functionalities

1. Web Crawler
2. Database Integration
3. HTML to TEXT using JSOUP
4. Page Parser
5. Pattern Matching
6. Page Ranking

Functionalities

Web Crawler

Recursively iterates over web links and stores the links referenced by a page in a list, thereby creating a hash of pages vs the links referenced.

Database Integration

In order to improve latency from web-crawler, caching was integrated using Database.

Functionalities

HTML to TEXT using JSOUP

Convertor to generate URL content in text format and remove common words.

Page Parser

Page parser holds the responsibility to parse the web pages into text, tokens and later feed them to pattern matcher & page ranker. It also acts as an interface to aid with interaction between main application and pattern matcher.

Functionalities

Pattern matching

Works on matching keywords with token generated for each file. A single TST is maintained for all files and for every token a TreeSet is maintained which stores URLs and their ranks.

Page Ranking

Rank is directly proportional to the occurrences of a token in URL’s content.

Code Walkthrough & DEMO

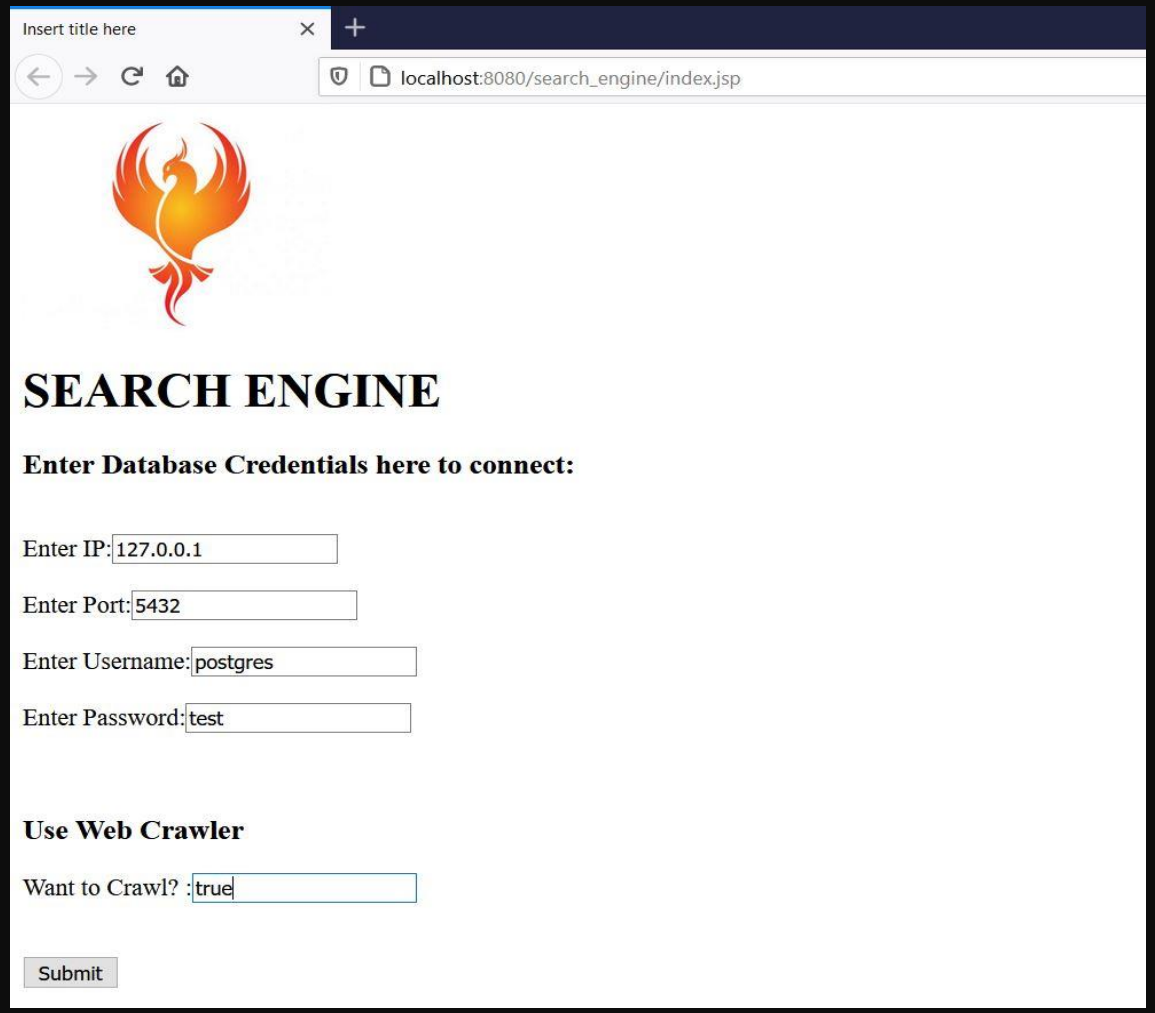
Web Integration

Server Used: Apache Tomcat Version 8.0

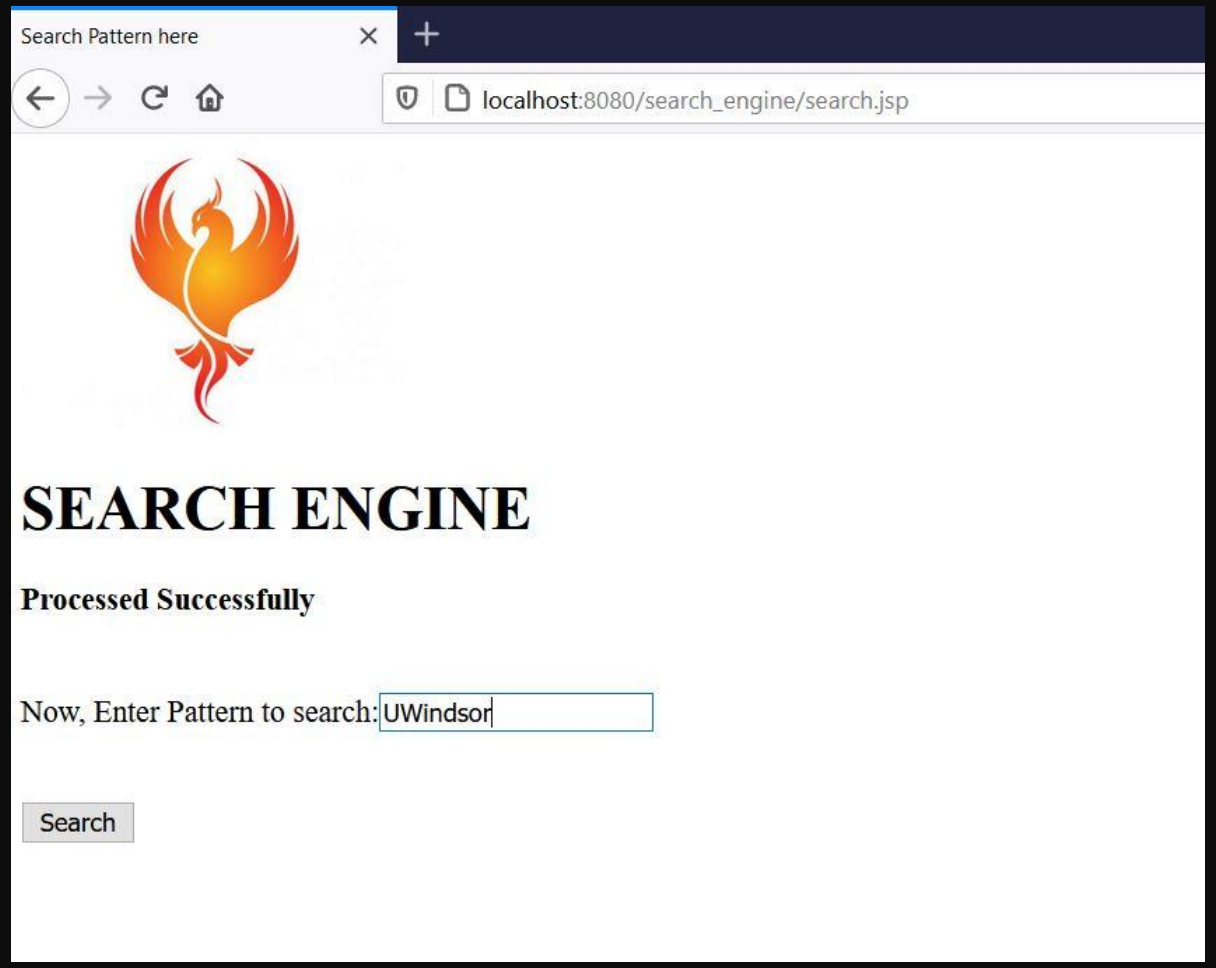
Dynamic Web Module Version 3.1

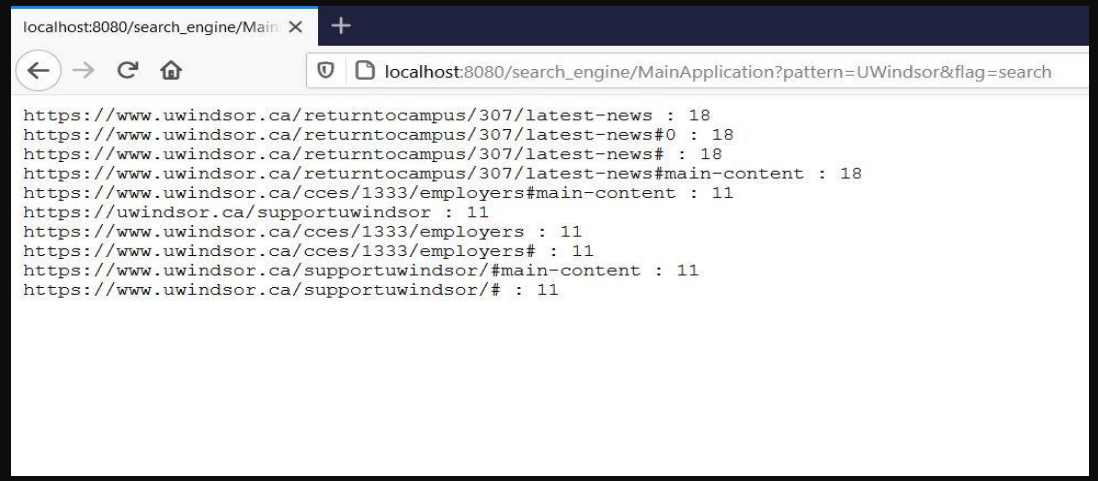
Eclipse Enterprise Edition

Advanced Java Concepts:
JSP, Servlet and JDBC



Web Integration





Conclusion

Application Analysis

1. Overall Understanding
2. CPU Time
3. Latency Bottlenecks

Future Enhancements

Features to Add:

1. Suggestion based search
2. Regular Expression Search
3. Searching a phrase
4. Runtime crawl

Latency Optimization:

1. Multithreading
2. Caching
3. Concurrency

References

1. Source code shared for Lab 4 (Text Processing) – TST & Queue
2. Lab Assignment snippets for JSoup, CPU time calculation & tokenizers.
3. Web links referred:
   https://www.culturainteractive.com/seo/block-web-crawlers-from-certain-web-pages/
   https://www.enterprisedb.com/postgresql-tutorial-resources-training?cid=437
   https://www.link-assistant.com/news/google-page-rank-2019.html
   https://www.javatpoint.com/java-hashmap
   https://www.javatpoint.com/java-treeset
   https://www.geeksforgeeks.org/comparator-interface-java/
   https://www.enterprisedb.com/postgres-tutorials
   https://www.tutorialspoint.com/jsoup/jsoup_extract_text.htm
   https://stackoverflow.com/questions/45206854/insert-date-into-database-postgres-jdbc
   https://zetcode.com/java/postgresql/
   https://www.postgresql.org/docs/9.1/arrays.html
   https://static.googleusercontent.com/media/research.google.com/en//pubs/archive/34439.pdf