# Report

## Table of Contents

# Introduction

[TBC]

# Literature Review

Following paragraphs are the literature review around the chosen topic.

## Full-Text Search Engine

Full-Text search engines perform linguistic searches against text data in target document or collection of text data based on the rules of a particular language (Microsoft Docs, 2020). Full-Text search engines consider the basic unit of Full-Text search as a token rather than a word (Melton & Buxton, 2006). Full-Text queries can include simple words and phrases or multiple forms of a word or phrase. A Full-Text query returns any documents that contain at least one match (also known as a hit). A match occurs when a target document contains all the terms specified in the Full-Text query, and meets any other search conditions, such as the distance between the matching terms. It widely used in modern technology like web search(Google Search, Bing Search, etc.), desktop search(search for local files) and even email search.

As the need of semantic-aware Full-text search and indexing against databases and documents keeps increasing in the past decade, full-text search systems became available to no-expert users who aren't familiar with documents they are searching, don't know the exact vocabulary used to index relevant documents, or don't know how to formulated advanced search queries (Tekli, Chbeir, et al., 2018). All of those issues mentioned above result in noisy or irrelevant search results and false positives & false negatives in the search result (See Precision and Recall).

### Precision and Recall

Precision and recall are often used to measure the performance of information retrieval or extraction systems, precision (aka. positive prediction value) is the fraction of relevant results among all results returned (in other words, how many returned results are relevant?), while recall (aka. sensitivity) is the fraction of the amount of returned relevant results among all relevant results in the dataset (in other words, how many relevant results are returned?) (Makhoul, Kubala, et al., 1999) (see Appendix a.).

In general, there is a trade-off between "precision" and "recall". High precision means that fewer irrelevant results are returned as a result of the query (fewer false positives), while high recall means that fewer relevant results are missing (fewer false negatives). traditional text-matching systems such as the LIKE operator in SQL gives you 100% precision with no concessions for recall. A full text search facility gives you a lot of flexibility to tune down the precision for better recall (Erickson, 2008).

For example, the SQL LIKE operator can be extremely inefficient. If you apply it to an un-indexed column, a full scan will be used to find matches (just like any query on an un-indexed field). If the column is indexed, matching can be performed against index keys, but with far less efficiency

than most index lookups. Therefore, most full text search implementations use an "inverted index" system to improve performance (see Inverted Index) (Erickson, 2008).

### Inverted Index

Inverted Index is an index where the keys are individual terms, and the associated values are sets of records that contain the term. Full text search is optimized to compute the intersection, union, etc. of these record sets, and usually provides a ranking algorithm to quantify how strongly a given record matches search keywords (Erickson, 2008).

As mentioned earlier, non-expert users of full-text search system often don't know how to formulate advanced queries or don't know the vocabularies used when indexing the documents (Tekli, Chbeir, et al., 2018). To address said issue the authors designed and constructed a extended version of inverted index called $SemIndex$, which empowered standard inverted index with a semantic network and a general keyword query model, in order to yield semantic-aware results (ibid.).

### Morphological Analysis (Stemming)

[TBC]

### Ranking

[TBC]

## Semantic Search

"Semantics" refers to the concepts or ideas conveyed by words, and semantic analysis is making any topic (or search query) easy for a machine to understand. [TBC]

## Apache Tika

Tika The Apache Tika™ toolkit detects and extracts metadata and text from over a thousand different file types (such as PPT, XLS, and PDF). All of these file types can be parsed through a single interface, making Tika useful for search engine indexing, content analysis, translation, and much more. You can find the latest release on the download page. Please see the Getting Started page for more information on how to start using Tika. [TBC]

## Gantt Chart

The planning of this project is done by utilizing Gantt chart (See Appendix b.).

There're several complications result in a shift of priority to develop a functional cross-platform full-text search API which return search result either from CLI (command-line interface) or `JSON` via REST Web API, instead of building both cross-platform UI and API.

The main processes of this project were requirements analysis, implementation, testing and reporting.

# Requirements Analysis

[sth. about requirement analysis in general]

## Functional Requirements and Progress

| Icon | ☐ | ☑ | 🚧 |
|---|---|---|---|
| Meaning | Planed | Done | WIP |

- API

  - ☑ 64-bit multi-platform support.
  - ☑ Indexing.
  - ☑ Searching.
  - ☐ Semantic Search. !!!
  - ☑ Web API + API Documentation.

- File type support

  - ☑ Microsoft Office 2007+ documents support.
    - ☑ `.docx`, `.docm` | Word 2007+ documents.
    - ☑ `.pptx`, `.pptm` | PowerPoint 2007+ documents. ⚠
    - ☑ `.xlsx`, `.xlsm` | Excel 2007+ documents. ⚠
  - ☐ Plain-text documents.
    - ☑ `.txt` | Text documents.
    - ☐ `.md` | Markdown documents. 🚧
    - ☐ `.rtf` | Rich text format documents.
  - ☐ `.pdf` | PDF documents.
    - ☑ Text-based PDF.
    - ☐ Image-only PDF. 🚧
  - ☐ `.epub`, `.mobi`

- Testing

| Icon | Meaning |
|------|---------|
| ✅ | Fully Tested |
| ✓ | Partially Tested<br>( Basic functionalities ) |

| File Type | Testing Status |
|-----------|----------------|
| `.txt` | ✅ |
| `.docx` , `.docm` | ✅ |
| `.pptx` , `.pptm` | ✓ |
| `.xlsx` , `.xlsm` | ✓ |
| `.pdf` | ✓ |

# Implementation

## API Design

[Class Diagram]

[API WorkFlow]

## Tools

### C#

### .NET Core

### ASP.NET Core

### Visual Studio Code

### Third-party Libraries

This project also utilizes several third-party libraries to implement certain functionalities.

### LiteDB

LiteDB is a embedded No-SQL single-file Database. It's used in this project as an object storage for indexed data.

## Optical Character Recognition

OCR Library

## Office Open XML

Open-XML SDK is an SDK that provides tools for working with Office Word, Excel, and PowerPoint documents developed by OfficeDev Team at Microsoft.

### Word 2007

```
issues & difficulties
```

### Excel 2007

```
issues & difficulties
```

### PowerPoint 2007

```csharp
protected override void Index()
{
    var startTime = DateTime.Now;
    Console.Write($"Indexing {Name}");

    # region PowerPoint
    var ppt = PresentationDocument.Open(path: Location, isEditable: false);
    using var slides = ppt.PresentationPart.SlideParts.GetEnumerator();

    while (slides.MoveNext())
    {
        var slide = slides.Current;
        using var text = slide.Slide.Descendants<TextBody>().GetEnumerator();
        while (text.MoveNext())
        {
            this.AddToIndex(texts: text.Current.InnerText);
        }
    }
    #endregion

    Console.Write($" >==> {Thumbnail.Count} unique words. {(DateTime.Now - start
}
```

# References

- Erickson, 2008. sql - What is Full Text Search vs LIKE - Stack Overflow. Accessed on July 1st, 2020.

- Makhoul, J., Kubala, F., Schwartz, R. and Weischedel, R., 1999, February. Performance measures for information extraction. In Proceedings of DARPA broadcast news workshop (pp. 249-252).

- Melton, J. Buxton, S., 2006. Chapter 13 - What's Missing? - Querying XML | ScienceDirect. Accessed on July 1st, 2020.

- Microsoft Docs, 2018. Full-Text Search - SQL Server | Microsoft Docs. Accessed on July 1st, 2020.

- Reiner, K. Chichao, C. Farzin, M. Ravi, K.,2006. Searching with context | ACM Digital Library. Accessed on July 1st, 2020.

- Tekli, J., Chbeir, R., Traina, A., Traina, C Jr., Yetongnon, K., Ibanez, C., Assad, M., Kallas, C., 2018. Full-fledged semantic indexing and querying model designed for seamless integration in legacy RDBMS. Accessed on July 2nd, 2020.

# Appendix:

a. Precision & Recall | [Back to content](#)

b. Gantt Chart | [Back to content](#)