Database design for Phase 1 submission:

For the phase 1 submission, to increase our efficiency we have not implemented our full database design. Instead, we appended each of the crawler entries into the required format. Then, we used the toTextFile function which adds the db into a text file.

However, for the final submission we do aim to implement our full database design which is outlined below.

Final database design:

Mapping indexes for words and URL:

By creating mapping indexes for the page URLs as well as the words, we can increase the efficiency of the retrieval. In the inverted file, instead of using the URLs and the actual words, we will be using the Page-ID and the Word-ID.

Page URL	Page-ID

Keyword	Word-ID

Forward index:

A forward index will be created to keep a track of the words in each document. This will be useful for when we need to delete or update the inverted file.

Page-ID	Keywords

Inverted files:

We will be creating two inverted files, one for the body of the pages and one for the title. Both files will include information about the term frequency and the position of the term in each document. The following is the design of the inverted file, we will use the same format for both the inverted files. The needed information for the vector space model is available as the term frequency will be stored and the document frequency can be found by keeping a count of the number of postings for each word. The total number of documents in the collection and the tfmax, will be stored separately.

Word-ID	Page-ID, term frequency, position

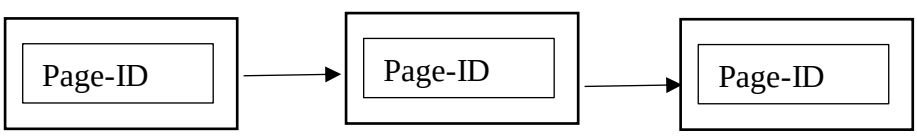
Page properties:

This database will contain information about the page, it will include the page title, last date of modification and the size of the page.

Page-ID	Page title, Last date of modification, size of the page

Adjacency list for parent-child relation

To track the parent-child link relation, we will use an adjacency list.



Database design: