**Phase 1 RocksDB Database Schema**

Group 19

|  |  |
| --- | --- |
| **RocksDB Name** | **Mapping** |
| WordMappingIndex | word -> wordID   * word: String * wordID: int * A keyword is converted into the corresponding wordID using Hash algorithm.   Explanation/Notes:   * The word inserted into this database is not the original keyword in the document but the stemmed version. * The same stemming algorithm will be used for queries to ensure finding correct wordID. * Hash algorithm with conflict properly handled will be used. It’s a quick algorithm to get the index. |
| URLMappingIndex | URL -> pageID   * URL: String * pageID: int * A URL is converted into the corresponding pageID using Hash algorithm. |
| PageInfo | pageID -> page<page title, URL, last modified date, size>   * pageID: int * page: Page   + Page is a class which contains:   + title: String   + url: String   + modDate: Date   + size: int   Explanation/Notes:   * The Page class is used to store the information of a web page. Accessing these information directly from pageID is easy by using this database. |

|  |  |
| --- | --- |
| InvertedIndex | wordID -> {pageID, <word position>}   * pageID: int[] * pageMap: HashMap<Interger, ArrayList<Interger>>   + pageMap uses pageID as the key. The value corresponding to the key is the positions of the wordID in this specific pageID. |
| ForwardIndex | pageID -> {wordID, <word position>}   * wordID: int[] * wordMap: HashMap<Interger, ArrayList<Interger>>   + wordMap uses wordID as the key. The value corresponding to the key is the positions of the specific wordID in this pageID.   Explanation/Notes:   * Store the words appear in a page along with their positions so that searching for phrases will be easier. |