Logo, company name

Description automatically generated

**School of Computer Science**

**Master of Applied Computing (M.A.C)**

**Subject Code: COMP8157**

**Subject Name: Advanced Computing Concepts**

**Prof Name: Dr. Mahdi Firoozjaei**

**Group Project by:**

**Group 11**

**Hamza Baig (110089314)**

**Dipen Vinodbhai Kalal (110094706)**

**Talha Mohammed Shamoon Choudhary (110087321)**

**Adeel Ahmed (110091296)**

**What is Web Search Engine?**

A place where user searches their query such as set of words or some meaningful words to look for on the internet. Which eventually results in the list of the webpages, Images, videos, content, or the material that gives some meaningful content related to performed search query.

How does it work?

It performs various operation to reach it’s end results. Firstly, the crawling is done to get added to the searching index. Which basically scans the entire web pages on that website. Hyperlinks plays the crucial role in linking the pages whether its internally linking or the external linking which supports the web page to reach the higher ranking on the search engine.

**Features of this web search engine.**

* Crawling
* Cache
* Searching Keyword
* Ranking
* Auto Correct
* Auto Complete

**Crawling**

Crawling is the method that search engine web crawlers use to explore, download, and extract links from a page to find other pages.

Search engine-recognized pages are frequently crawled to see if their content has changed since the last time it was indexed. After a page has been crawled, a search engine may notice changes, in which case it may update its index.

Typically, search engines will try to crawl and index every URL they come across.

However, search engines will often be unable to read the content of a file other than the associated filename and metadata if the URL is for a non-text file type as an image, video, or audio file.

Non-text file types may only be able to have a small amount of information extracted by a search engine, but they can still be indexed, appear in search results, and get traffic.

Implementation of crawling in Java

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Output

**Text

Description automatically generated**

**Cache**

(Taking example of google caching)

A type of preserved duplicate of a website that may be obtained from the servers is the Google cache. As soon as a robot accesses the page, a copy of it is made. There are two different types of caches: the proxy cache and the browser cache (such as Firefox, Chrome, etc.). Google therefore caches, indexes, and categorizes all websites. To determine its relevance to a query, Google evaluates the cached version. In general, Google's servers are substantially faster than many other web servers, so accessing a page's cached version is frequently quicker than accessing the page directly. Each organic result (blue link) typically includes a link to the cached version.

Implementation of caching in Java

After crawling it automatically caches the crawled web pages in the cache.txt file for the next time, so it doesn’t have to load the link again and again.

Text

Description automatically generated

**Searching and Ranking**

Each search engine ranks websites in a different way. Each and every search engine will use its own algorithm to determine the ranks in a unique way. Additionally, no Search Engine will return the same results due to the various algorithms each Search Engine employs. For instance, it's quite unlikely that Google and Yahoo will ever return the identical search result. The pages of the website are ranked by search engines based on a number of parameters.

Ranking factors for search engines:

1) On-Page Elements

2) Off-Page Elements

* On-Page Elements

The website page and the keywords are affected by on-page elements. Each website must include on-page elements in order to rank. The publisher has some influence on these variables. factors like the website's page content, title tags, and Meta tags

* Off-Page Elements

These elements are those that enhance a website's ranking outside the confines of individual web pages. The publisher has no direct influence on these ranking variables. Off-page factors don't involve text or other content found on a website's page.

Here in our code the searching and ranking works on the max number of the occurrence of the keyword used for searching the crawled URL and so. The Maximum repetition the higher the page rank.

Implementation of Searching and Page Ranking in Java

Graphical user interface, text, application

Description automatically generated

Text

Description automatically generated

Output

Graphical user interface, application

Description automatically generated

Text

Description automatically generated with medium confidence

**Auto Correct**

Nowadays, many search engines rectify "mistakes" in your search phrases automatically when you conduct an online search: The websites adjust your search if it matches a more popular word or spelling when you submit it.

When looking for a more unusual term or phrase, this can get in the way but is useful if you make a typo. As a result, the search engines allow you to select which of the two searches you would like to perform in addition to informing you of any such correction.

Implementation of Auto Correct in Java

Text

Description automatically generated

Output

Graphical user interface, text

Description automatically generated with medium confidenceGraphical user interface, application, website

Description automatically generatedText

Description automatically generated

**Auto Complete**

The search engine will make many predictions about how your question might be finished while you enter. On a university website, for instance, when you put in "course," it proposes "courses in English," "courses in geography," or "course catalogue." Because autocomplete also completes words, for example, if you type "lect," the engine will suggest "lecture" or "lecturer."

Here we have implemented the auto complete feature based on the user history, which allows user to find the related words from the search history performed.

Implementation of Auto Complete in Java

Graphical user interface, text, application

Description automatically generated

Output

Background pattern

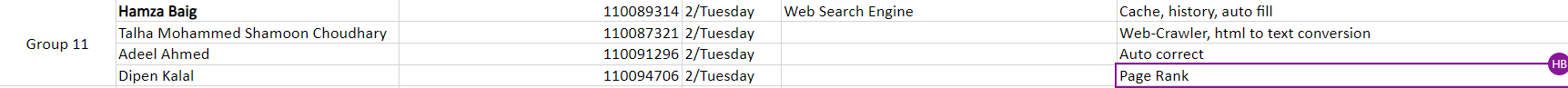
Description automatically generated with low confidence

**Folder Structure and Additional files used for dependency**

Graphical user interface

Description automatically generated with medium confidence

**Contribution List**

****

