Search Engine Optimization

**Introduction:**

Search Engine Optimization (SEO) is one of the core marketing strategy focused on the growing visibility in the search engine results. There are many aspects of SEO from the words on the webpage to the ways other website links to that webpage. It is useful for making sure that the site is structured in a way that search engines understand.

**Problem:**

To build an engine that will take any URL and identify the most relevant topic on the page.

**Input:** Any URL

**Output:** List of common topics that best describe the contents of the page

**Factors considered in building the Search Engine:**

There are different factors that can be considered in building the Search engine. For the purpose of this assignment I considered some of the On-page elements and General attributes.

The On-page elements are required to identify the most relevant information of the page that is being searched. Some of them are listed below.

* Uniform Resource Locator (URL) - URL is used to uniquely identify a webpage
* Title - It is useful to get the global title of the webpage
* Meta-description - It is used to provide concise summaries of the webpage
* Meta-keyword - It helps the search engine to identify the topic of the page. Each keyword accurately reflects the content of the webpage
* Header-1 - It is used to identify the main headings/titles of the contents in the webpage. It makes your font big
* Header -2 - It is used to identify the sub-headings of the contents in the webpage. It makes your font big, but it's size will be less compared to Header-1
* Image-list - It contains the image descriptions of all the images in the webpage, which will be useful for search engine optimization

The General attributes acts as a helpers to the Search engine. Some of the General attributes that are considered is listed below.

* Meta-robots - It tells the search engine what to follow and what not to follow. It helps the search engine to decide about what pages you want to hide from crawlers and what pages you want to index and look at
* Canonical link - It helps to tell the Search engine that certain similar URLs are actually one and the same

**External library / Tools used:**

I used **Jsoup** library for parsing the HTML webpage. It provides a easy and convenient java API to access the contents of the webpage. I used **Eclipse** for developing this application

**Design:**

1. **SEO\_Main.java:** It is the starting point of the application. It takes the URL input from the user (from command line arguments ), initiates a validity check of the URL, initiates a connection for accessing the required page and calls the appropriate class for processing the data and get the results.

2. **CheckURL.java:** It is used to check the validity of the URL entered by the user. If an invalid URL is entered by the user, an appropriate message will be shown and the program exits smoothly.

3. **HtmlParserConnection.java:** It is a **Singleton** class. In this class connection from the application to HTML webpage is set, so that the webpage data can be processed.

4. **OnPageElements.java:**  It has the header and setter methods for all the On-page elements ( URL, title, Meta-description, Meta-keyword, Header-1, Header-2, Image-list ). It is a Plain Old Java Object (POJO) class.

5. **GeneralAttributes.java:** It has the header and setter methods for some of the General attributes ( Meta robots, Canonical link ). It is a Plain Old Java Object (POJO) class.

6. **OnPageElementsHandler.java:** It is used for processing the On-page elements of a webpage and displays its results. It has the business logic for processing the On-page elements information by using the **Jsoup** API.

7. **GeneralAttributesHandler.java:** It is used for processing the General attributes of a webpage and displays its results. It has the business logic for processing the General attribute information by using the **Jsoup** API.

8. **OnPageElementsBuilder.java:** I used **Builder** pattern for setting and getting the values of On-page elements. Since the parameters required for On-page elements are expected to grow (since I have used only some), Builder pattern provides a better way to construct the object for the On-page elements.

9. **GeneralAttributesBuilder.java:** I used **Builder** pattern for setting and getting the values of General attributes of a webpage. Since the parameters required for General attributes are expected to grow (since I have used only some), Builder pattern provides a better way to construct the object for the General attributes of a webpage.

**Design Patterns Used:**

1. **Singleton Pattern:**

I used Singleton pattern for the HTML parser connection. It is a creational pattern. Only one instance of the **HtmlParserConnection** class is created. I developed using this design pattern for parser connection because, when there are many URL's to be processed, there is no need to create an object of this class, every-time an parser connection is made. Every URL, will use that object to create an connection to parse the HTML data of the webpage.

2. **Builder pattern:**

I used Builder pattern to build the objects of **OnPageElements** and **GeneralAttributes** class. It is a convenient way to build objects for these class using the pattern because the parameters for these classes are expected to grow. It is used to separate the object from its representation. It provides better readability for the programmers.

**Exception Handling:**

1. For the purpose of this assignment, If there is more than one URL or no URL for processing, An **Illegal argument** exception is thrown, which says "Only one URL is required".

2. For checking the URL validity, the URL entered is not in the correct format, **MalformedURLException** and **URISyntaxException** is thrown, which is caught and an appropriate message is displayed for entering the URL in the correct format.

3. If the entered URL is in the correct format, but if it not able to connect with the required page an **IOException** is thrown, which is caught and an appropriate message is displayed.

**Future works:**

1. Add additional parameters for measuring the Links like Domain authority, External followed links, general attributes like Page load time, Cache URL and getting the IP address/country from where the request to the page was originated , special markups for getting special information like (Twitter, Facebook.. etc ) and status codes'. It will result in better search engine optimization.

**Conclusion:**

I learnt a lot from this assignment. Looking forward for more challenging projects.