International journal of science, engineering, and information technology
Volume 07, Issue 01, December 2022
Journal homepage: https://journal.trunojoyo.ac.id/ijseit

DESIGN AND IMPLEMENTATION SEARCH ENGINE USING METHOD VSM (Vector Space Model)

Khotibul Umam[a], Yudha Dwi Putra Negara[b]

[a]Madura State Islamic Institute, Pamekasan, Indonesia, [b]Author afiliation, City, Country
[b]Department of Informatics Engineering University of Trunojoyo Madura, Bangkalan, Indonesia

ABSTRACT

Along with the rapid increase in the amount of information on the internet, many internet visitors feel confused in getting information about notebooks available online on the internet. Therefore, it is necessary to compile or organize information about notebooks from several web addresses that contain information about brands, specifications, prices of notebooks, so that it can make it easier for visitors to carry out the process of searching for information about brands, specifications, notebook prices and get information that is in accordance with visitor wishes. A special search engine is an application that functions to find specific information based on keywords entered by visitors. This search engine will be able to provide information about brands, specifications and prices of notebooks easily and quickly. In the process of working this search engine, later all the information on the internet site will be extracted and then stored in a database. then displayed to visitors. Making this final project produces a special search engine application that helps visitors get information about brands, specifications, and prices of notebooks from several webs that are determined directly. From several experimental results, importing brand keywords, specifications and notebook prices in the search engine resulted in 80% precision and 94% recall.

Keywords: *Notebooks. Keyword, Web Crawler, Extract*.

Article History
Received 20 September 22 Received in revised form 20 October 22 Accepted 2 November 22

1. Introduction

With the rapid increase in the amount of information on the internet, many internet visitors feel confused in getting information from the internet itself. Often the information obtained and displayed by the internet is still not in accordance with what internet visitors want. For example, when you want to know the price of a notebook online, when a visitor inputs the price of their notebook on a query processor or search engine such as Google, what appears is the information links from the website address, not the price information from the notebook.

Therefore, it is necessary to compile or organize notebook information from several web addresses containing information about brands, specifications, notebook prices, so that it can make it easier for visitors to search for information about brands, specifications, notebook prices and get information that suits their needs. visitors. Search engines are generally used to find the data and information needed. By using a search engine, the user can determine what data to search for and the limits so that only data that matches the criteria will be displayed. A special search engine is an application that functions to find specific information based on keywords entered by visitors.

For that we need a special search engine design, to help internet visitors get information about the brand, specifications and prices of notebooks from several websites that contain notebook information.

Not all website addresses that contain brand information, specifications and notebook prices will be crawled. But only on certain website addresses that have been predetermined. The main hope of developing this software is to help internet visitors get information about brand data, specifications and notebook prices from several predetermined website addresses quickly and accurately.

2. Literature Review

2.1. Search Engine

A search engine is a computer program specifically designed to help someone find files stored on a computer, for example on a public web server on the web (www) or on your own computer. Search engines allow us to request media content with specific criteria (usually containing a phrase or word we want) and obtain a list of files that meet these criteria. Search engines usually use an index (which has been created previously and updated regularly) to search for files after visitors enter search criteria [2].

* *Corresponding author.* Phone :
E-mail address: yudha.negara@trunojoyo.ac.id.

E-ISSN 2548-4214

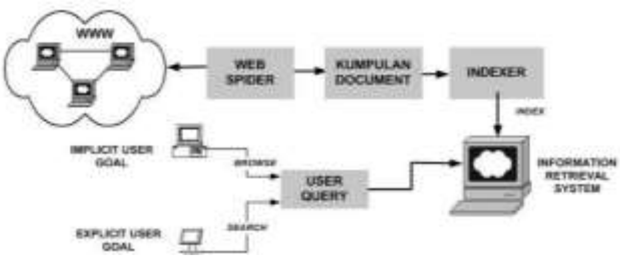

Figure 1. Internet search engine design

2.2. Information Retrieval

Information Retrieval (IR) is a field of computer science concerned with retrieving information about a subject from a collection of data objects. This is not the same as Data Retrieval, which in the context of a document consists primarily in determining which documents from the collection contain keywords from the visitor's query. Information Retrieval relates to satisfying visitor needs [4].

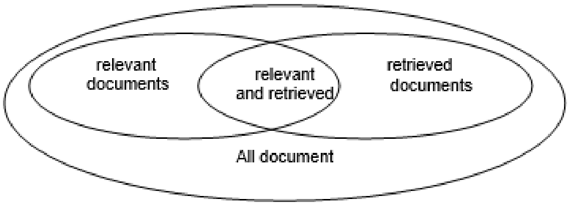

Figure 2. Relationship between Relevant and Retrieved Documents

Every search engine has the ability to retrieve information on various HTML pages that they store in their database. The performance level of a search engine is generally measured by two parameters, namely recall and precision [5].

Precision is the probability of the number of relevant documents from all documents retrieved compared to the total number of documents retrieved. Mathematically, the precision can be written as:

$$Precision = \frac{|\{Relevant\} \cap \{Retrieved\}|}{|\{Retrieved\}|}$$

Figure 3. Formula for calculating precision

Information :
Precision: the level of precision
{Relevant} : a collection of relevant documents
{Retrieved} : document found

In its application to search engines, this amount of precision is very difficult to calculate because the amount of information is very large, and the amount of information retrieved is always increasing and being updated every time. This update increases the number of retrieved documents and may increase the number of relevant documents. However, this addition tends to reduce the level of precision of a search engine because the number of documents that must be retrieved must be far more than the relevant documents [5].

Recall is the probability of the relevant documents being retrieved compared to the number of relevant documents. Mathematically recall is written as:

$$recall = \frac{|\{Relevant\} \cap \{Retrieved\}|}{|\{Relevant\}|}$$

Fugure 4. Formula for calculating recall

Recall is a measure of the sensitivity of an information retriever in finding information that is relevant to the entered query. This is closely related to the ability of this information retriever to store documents and other information, especially information relevant to the information needed [5].

The web search process has two main parts: off-line and on-line. The off-line section is executed periodically by the search engine and downloads a sub-set of the web to build a collection of pages, which are then turned into a searchable index. The on-line section is executed every time a visitor's request is executed and uses the index to select several candidate documents sorted according to an estimate of what is relevant to the visitor's needs.

2.3 Web Crawler

Web crawler is a program that gathers information about something, the results of which will be stored in a database. A web crawler will walk through web pages and collect documents or data in them. Furthermore, the web crawler will build an index list to facilitate the search process [1].

Web crawlers work automatically by providing several website addresses (URLs) to visit and storing all the information contained therein into the database that has been provided. Every time a web crawler visits a website, it will record all the links on the page it visited and then visit it again one by one [1].

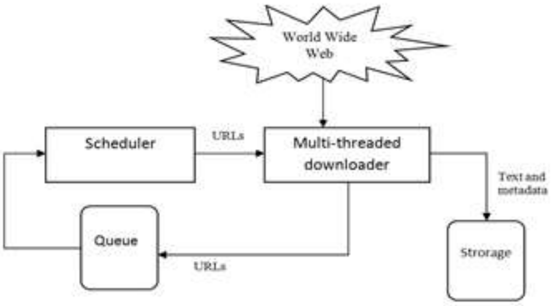

Figure 4. General design Web crawler architecture

The crawling strategy that will be implemented is largely determined by the crawler usage scenario. There are at least four scenarios that are often used so far [8].

1. Take a snapshot of a web page as complete as possible with a breadth-first Search engine, often used on large-scale commercial search engines such as Google, Yahoo etc.
2. Periodically updating retrieved web pages to keep search engine indexes up to date. This process can be done efficiently by implementing heuristics that will update the most important web pages more frequently. Optimization can be done based on observational data on the update history of a web page.
3. Crawling Hidden Web; explore web pages generated from the database, which can only be obtained by filling out the form provided on the web page. This work is quite strenuous because of its enormous scale (early studies suggest it is 400 to 500 times larger than the static Web).
4. Focused Crawling: explore web pages focused on a specific topic.

2.4. Regular Expression

Regular expression or often referred to as Regex is a formula for searching the pattern of a sentence/string. Often people think that regex is difficult and confusing. But actually, regex is very helpful in finding sentence patterns. So that experiments on all possible sentence patterns do not need to be carried out [9].

Regular expressions are generally used by many word processors / text editors and other tools to search and manipulate sentences based on a certain pattern. Many programming languages that support Regular expressions such as PHP, Perl, VB and Tcl.

A very good reason to use regex is that they are very powerful. At low levels the regex can search for a fragment of a word. At a high-level regex is able to control the data. Good search, delete and modify. Let's think about how to find a file on the hard disk. Often used the character “?” and "*". Visiting the character “?” means that a file that contains a certain character is being searched and the character "*" means that zero or more characters are being searched. Some commonly used patterns in regex are shown in table 1 below.

Table 1. Regular expression pattern

Pola	Penjelasan
[ ]	bracket expression. matches a single character in brackets, eg pattern "a[bcd]i" matches the strings "abi", "aci", and "adi". Visiting the range of letters in brackets is allowed, for example: the pattern "[a-z]" matches any character between the strings "a" to "z". pattern [0-9] matches any of the numbers. if you want to find the character "-" too, the character must be placed in front or behind the group, for example: "[abc-]".
[^]	matches a character not in brackets, as opposed to above. eg: pattern "[^abc]" matches any single character except "a", "b", "c".
?	matches the previous zero or one character. eg: pattern "died?" matches the strings "die" and "died".
+	matches one or more previous characters. for example: "yu+k" matches "yuk", "yuuk", "yuuuk", and so on.
*	matches the previous zero or more characters. eg: pattern "hu*p" matches the string "hp", "hup", "huup" and so on.
{x}	matches the previous character x number of characters. eg: pattern "[0-9]{3}" matches any number that is 3 digits in size.
{x,y}	matches the previous character any number of x to y characters. eg: pattern "[a-z]{3,5}" matches all lowercase letters consisting of 3 to 5 letters
!	if it is placed in front of the pattern, it means "not". e.g. pattern "!a.u" matches any string except "alu", "foo", "ash", "asu", "aiu", and so on
^	if placed in front of the pattern, will match the beginning of a string.
$	if placed behind the pattern, will match the end of a string
( )	grouping. used to group characters into single units. strings that match the pattern in parentheses can be used in subsequent operations. kind of variable.
\	escape characters. returns the metacharacter function to a regular character. on some systems it can mean the opposite, namely metacharacter using an escape character in front of it

2.5. HTML

HTML stands for Hypertext Markup Language and is the standard language used to display web documents. HTML elements consist of 3 basic parts, namely the opening tag, the content of the element and finally the closing tag. Every web page should have at least 4 main elements, namely html, head, title, and body [10].

```
<HTML>
<HEAD>
<TITLE>contoh1.htm</TITLE>
</HEAD>
<BODY>
            Kepala atau kop dokumen
</BODY>
</HTML
```

Figure 5. HTML document structure

2.6. URL (Universal Resource Locator)

URL stands for Uniform Resource Locator is a series of characters according to a certain standard format, which is used to indicate the address of a resource such as documents and images on the internet.

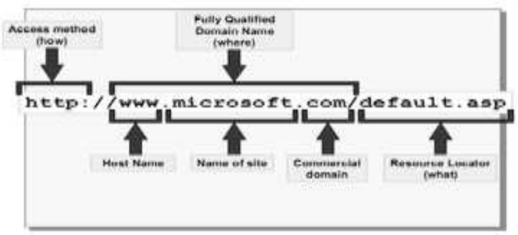

Figure 6. URL Overview

2.7. URL Normalization

URL normalization is a way to change the input URL format to a standard, URL normalization can be used to keep crawler data duplication [12].

Here is an example of URL normalization [12]:

1. Deletion of String after #, because string sign in behind the # sign is a marker where the browser cursor position is placed.
   http://localhost/index.php#definition →http:/localhost/index.php
2. Removal of URLs containing without being followed by // such as mailto: javascript:fileto:
3. Deletion of String file index.php,index.html.index/asp http://localhost/index.php → http://localhost/
4. Marking URLs that do not have /
   http://localhost → http://localhost/
5. Giving the www mark on the hostname
   http://google.com/ → http://www.google.com
6. Deletion of // to /
   http://localhost// → http://localhost/

2.8. TF-IDF

The TF-IDF method [3] is a way to weight the relationship of a word (term) to the document. This method combines two concepts for calculating weights, namely, the frequency of occurrence of a word in a particular document and the inverse frequency of the document containing the word. The frequency of occurrence of a word in a given document indicates how important the word is in the document. The frequency of documents containing the word indicates how common the word is. So that the weight of the relationship between a word and a document will be high if the frequency of the word is high in the document and the frequency of the entire document containing the word is low in the document collection (database) [14].

$$w_{ij} = tf \times idf$$

$$w_{ij} = tf_{ij} \times \log \frac{N}{n}$$

Figure 7. General Formula for TF-IDF

Information :

wij = weight of word/term tj to document di

tfij = number of occurrences of the word/term tj in di

N = the number of all documents in the database

n = number of documents containing the word/term tj
(at least one word, namely term tj)

2.9. VSM *(*Vector Space Model*)*

The Vector Space Model or Term Vector Model method is an algebraic model for describing text documents (some objects) as vectors of identifiers. Usually used in information filtering, information retrieval, indexing and ranking of mutually relevant ones. 15]. The technique of this vector space model is to calculate the value of the angle cosine of two vectors, namely W from each document and W from keywords.

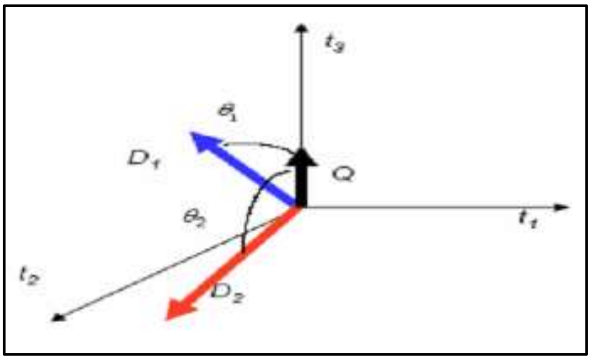

Figure 8. SVM

Information:

t = word in database

D = document

Q = keyword

The formula for calculating cosine similarity

$$similarity\ (\vec{d}_j, \vec{q}) = \frac{\vec{d}_j \cdot \vec{q}}{|\vec{d}_j| \cdot |\vec{q}|} = \frac{\sum_{i=1}^{t} (w_{ij} \cdot w_{iq})}{\sqrt{\sum_{i=1}^{t} w_{ij}^2 \cdot \sum_{i=1}^{t} w_{iq}^2}}$$

Figure 9. Formula cosine similarity

Information :

- similarity (dj, q) = The level of similarity of a document with a particular query
- wij = the weight value of the i-th word/term in the vector for the j document
- wiq = the weight value of the i-th word/term in the vector for the q-th query
- t = number of documents containing the word/term tj

(at least one word, namely term tj)

3. Research Methodology

in this case the first method is to design and create a special search engine application and web crawler that functions to retrieve information on the internet, namely about the brand, specifications, and price of notebooks, from the data that has been obtained, the web extraction process and removal of HTML tags uses Regular expressions. In the web data extraction process, the initial step is to select (crop) the part of the web page that only contains the information to be retrieved. After that, the result of the crop process is carried out by the process of removing (replace) the html tag using the regular expression function so that only the brand information, specifications and price of the notebook are left.

Data that is clean from HTML tags and produces information that is in accordance with what is desired, then the information is stored in a database and this information data will be further processed in the search engine using the VSM (Vector Space Model) method to calculate resemblance to the key word entered by the user.

3.1. Search Engine System Design

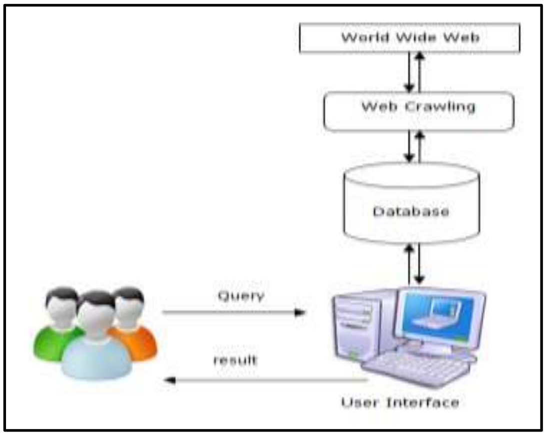

Figure 10. Search Engine System Design

In this general search engine system design, it describes the flow of this particular search engine system. In this system, web crawling performs searches on the world wide web and stores information on brands, specifications and prices of notebooks in the database. After the information is stored in an orderly manner, the last process is to perform a query (search) for the required information on the user interface based on the keywords entered.

The crawler program design is shown in Figure 11. The flow in the crawler system in general is as follows.

1. the crawler application gets the initial URL input from several web addresses online and then confirmed to the internet (www)
2. after being confirmed, the process of multithread downloader is carried out (download simultaneously at a time) web page based on the initial url earlier. The results of the download of the web page are stored in tb_raw.
3. In tb_raw, an offline process is carried out, first filtering URLs that enter the destination criteria, then if it has been filtered, the content extractor process is carried out (retrieving content from the desired web) and then stored in tb_content.
4. The second offline process on the offline extractor processes the extraction of the url from the downloaded web page and then saves it in tb_url to continue browsing the web page. This is done continuously as long as there is a queue of URLs.

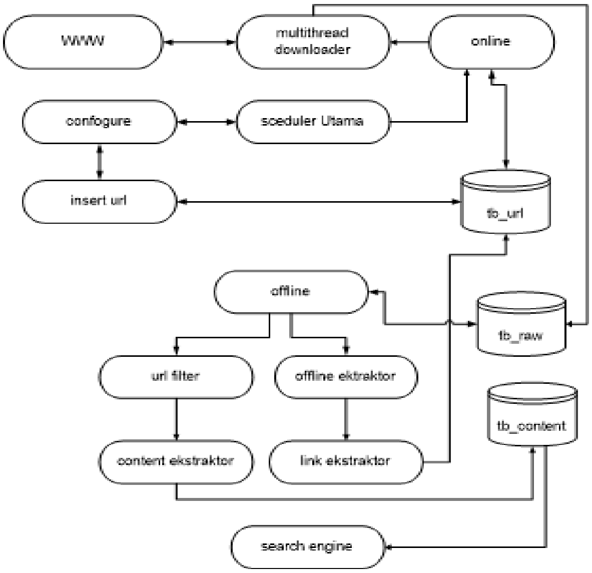

Figure 11. Crawler Web Design

The system design in this application is divided into 2 parts, namely the on-line process and the off-line process. The online section to deal with includes online scheduler process, html download and Multithread. As for the Offline process, it handles the scheduler offline process, link extractor and content extractor. The components in this application are quite a lot because each step of this process is made a separate program but is connected to each other. Starting from the connection checking process, URL input, web page download process, URL filter, content extraction, saving content to database, URL extraction, and looping settings.

4. Result

4.1. System Testing and Analysis

This section describes test data, and test scenarios. Run the Program on a web crawler

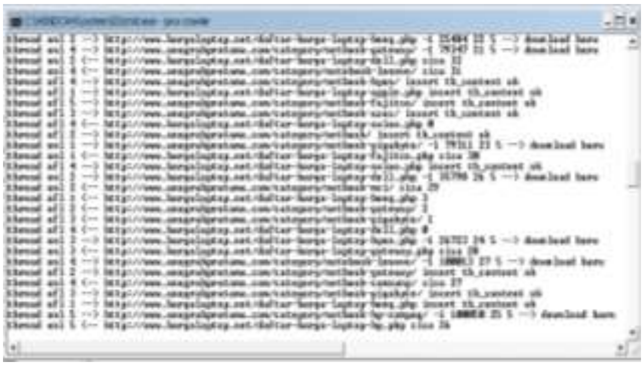

Figure 12. The process of running a web crawler.

The trial data in this application are 698 notebook brand data in the database and consist of 23 different brand classes taken from the details in Table 2.

Table 2 Notebook brand data in the database

No.	Merk	Amount (data)
1	Toshiba	59
2	Acer	45
3	Compaq	8
4	Dell	37
5	Sony	25
6	Asus	86
7	Lenovo	69
8	axioo	55
9	Fujitsu	44
10	Apple	23
11	Advan	9
12	A-Note	6
13	Byon	21
14	BenQ	33
15	HP	73
16	Zyrex	12
17	Gigabyte	8
18	Gateway	13
19	MSI	29
20	InNote	3
21	Viewsonic	1
22	eMachines	1
23	Aedupac	8

The trial scenario for this application is to calculate recall and precision by testing 10 users to use this application with 2 trial sessions. In the first session the user did the searching process 5 times based on the menu tab provided. From the results of each searching process, the recall value and precision will be calculated. And the average precision will be calculated from this test.

Table 3 Trial Scenario 1 user

Sesi	Tab	Menu
I	1	Merk
	2	Harga
	3	Merk + Harga
	4	Merk + Spesifikasi
	5	Merk + Spesifikasi+Harga
II	1	Merk
	2	Harga
	3	Merk + Harga
	4	Merk + Spesifikasi
	5	Merk + Spesifikasi+Harga

Results

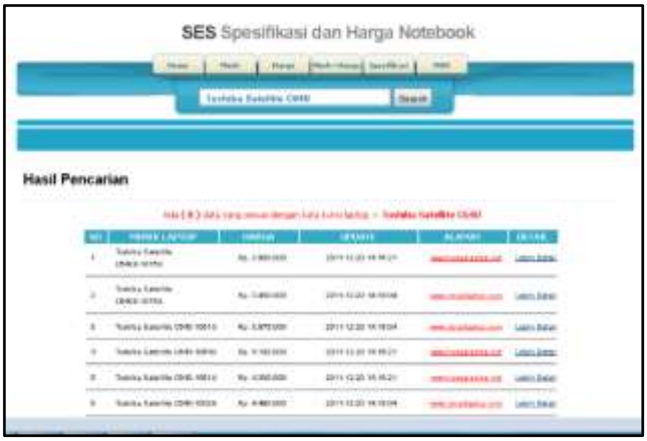

Figure 12. Uji coba dengan *query* "*Toshiba satellite c640*"

Figure 12 is an example of a test in a test scenario with the query "Toshiba satellite c640" and the test data in the database.

After testing the acquisition system for this particular search engine application, Table 4.3 is the result of the average recall and precision values based on each menu tab.

No	Menu	Nilai	
		Recall	*Precision*
1	Merk	0.9685	0.3945
2	Harga	1	1
3	Merk + Harga	0.7925	0.955
4	Merk + Spesifikasi	0.966	0.6915
5	Merk + Spesifikasi+Harga	1	1
Rata-rata		0.9454	0.8082

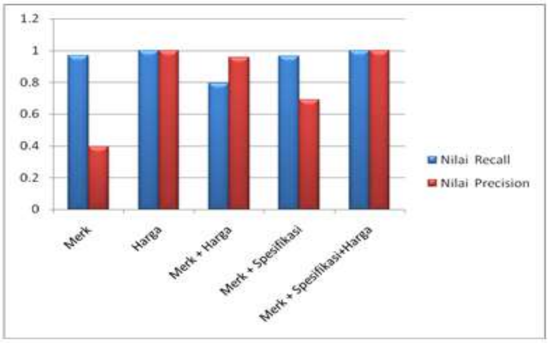

Figure 13. Graph of recall and precision values with a total of 50 keyword data

4. Conclusion

After completing the design and manufacture of a search engine system or application using the VSM (Vector Space Model) method to collect data about brands, specifications and prices of notebooks and conduct trials and evaluations, the following concluu\\usions can be drawn:

1. A search engine application for information on brands, specifications, and prices of nootbooks has been successfully created using the VSM (Vector Space Model) method which has a precision value of 80% and a recall value of 94%.
2. A program has been successfully created to extract information about notebook brands, specifications, and prices from five web pages using regular expression writing rules.

REFERENCES

[1] Utami, P.W. 2009. "Perancangan Dan Pembuatan *WebSearch engine* Aplikasi Panduan Pembelian Spesifikasi Komputer Rakitan *Online* Dengan Memanfaatkan *Google Gears*". Jurnal Tugas Akhir ITS Surabaya.

[2] Riyadi, T. 2009. Jenis-jenis *Search engine*. <*URL* http://www.trikaja.co.cc/Jenis-jenis-search-engine.pdf/> Diakses 11 April 2011

[3] Gozali, F. And Faezal, M.F. 2004. "Peranan *Web Spider* Dalam *Internet Search engine*". JETri, Volume 3, Nomor 2, Februari 2004, Halaman 17 - 32, ISSN 1412-0372.

[4] Dr. Baeza-Yates.R. 2004. "Effective *Web Crawling*". Dept. of Komputer Science - University of Chile. November 2004.

[5] Destrianti,G. 2011. "*Akurasi dalam Pencarian pada Search engines*". Makalah II2092 Probabilitas dan Statistik, 2010/2011.

[6] Krisna.V., Hu.W., Bhatia.A. "Design and Implementation of Mobile World Wide *Web Search engines*". Department of Electrical Engineering University of North Dakota Grand Forks, ND 58202-7165

[7] Ali, H.A. 2007 "Multi-Agent Sistem for Specific Domain Search Engine Based on Distributed Classification Approach". Komputers Engineering & Sistems Dept., Faculty of Engineering, Mansoura Univ. Mansoura, Egypt.

[8] Widyantoro, D.H. 2006. "Survey Arah Penelitian, Pengembangan Dan Penerapan Penjelajah Situs *Web*". Prosiding Konferensi Nasional Teknologi Informasi & Komunikasi untuk Indonesia. Bandung, 3-4 Mei 2006.

[9] Meliantara, A. 2010.Penerapan Reguler Expression Dalam Melindungi Alamat Email Dari Spam Robot Pada Konten Wordpress. <*URL* http://www.docstoc.com/docs/47645435/penerapan-regular-expression-dalam-menjaga-alamat-email-dari-spam.pdf/> Diakses 11 April 2011

[10] HUSNI. 2007. Pemprograman database berbasis *web*. Graha ilmu: Yogyakarta.

[11] Erawan.S.Kom.L. 2010.Pengertian word wide *web*. <*URL ndaa-unyuu.blogspot.com/p/pengertian-word-wide-web.pdf*/> Diakses 11 April 2011

[12] Erawan.S.Kom.L. 2010.Pemprograman *Web*. <*URL ndaa-unyuu.blogspot.com/p/pemrograman-web-lalang-erawan-s.pdf*/> Diakses 11 April 2011

[13] Arifin.M.S. 2011. "Desain dan Implementasi *Web crawler* untuk menghimpun *Website* Bahasa Indonesia". Tugas Akhir Universitas Trunojoyo .2011.

[14] Intan, R. dan Defeng, A.2006." Hard:Subject-Based Search Engine Menggunakan Tf-idf dan Jaccard's Coefficient".Jurnal Teknik Industri 8, 1:61-72

[15] Indraanandita, A., Susanto,B. dan Rachmat, A.2008. "Sistem Klasifikasi dan pencarian judul dengan menggunakan Metode Naïve Bayes dan Vector Sapace Model".Jurnal Informatika 4,2:9-18.