

# Rancang Bangun Plugin WordPress Penghitung Kepadatan Kata Kunci

Adhi Kusnadi<sup>1</sup>, Tirta Prawisuda<sup>2</sup>

Fakultas Teknik dan Informatika, Universitas Multimedia Nusantara, Gading Serpong, Indonesia  
adhi.kusnadi@umn.ac.id<sup>1</sup>, tirtaprawisuda@gmail.com<sup>2</sup>

Diterima 14 November 2016

Disetujui 12 Desember 2016

*Abstract—In optimizing a website, web owners are required to write articles that relate to the keywords and put into the website. Therefore is needed a keyword density plugin app to calculate keywords and suggest an article content writer to add or subtract keywords in the article. The keyword density counter is an app that counts keywords in an article. The results obtained from this application is the ratio of keywords to existing words of the pages of websites or articles. The benefits of using keyword density counters are content writers becoming aware that articles already have enough density, according Google's recommendation keywords density is 2-3% of article, then the article will give a signal to search engines. Results obtained is the same for validation plugin calculations after. Compared to other similar plugins. Keyword density counters have been created with 50.85% faster than other similar plugins.*

*Index Terms—keyword, density, counter, article, plugin.*

## I. PENDAHULUAN

Dalam dekade terakhir, ketertarikan akan mesin pencari internet telah sangat meningkat. Hal tersebut merupakan fakta bahwa mesin pencari internet telah menjadi populer baik sebagai mesin pencari informasi dan sebagai media iklan secara online. Universitas Teknik Georgia mengadakan survei dan mendapatkan hasil bahwa mesin pencari internet dipakai oleh 85% pengguna web. Mesin pencari terkemuka seperti Google melaporkan bahwa pendapatan pada tahun 2005 sebesar 1,578 miliar USD dan 1,559 miliar USD berasal dari iklan secara online [1]. Terdapat berbagai faktor untuk menaikkan peringkat sebuah situs, salah satu faktor yang dominan adalah kata kunci [14].

Kata kunci tersebut berada di berbagai tempat, seperti pada hostname, pada tag judul di bagian HyperText Markup Language (HTML) header, pada meta-description tag, pada bagian dari Uniform Resource Locator (URL), dan pada isi artikel dalam situs. Hal ini membantu pemilik web untuk mengoptimalkan halaman web dan mendapatkan posisi yang lebih baik di halaman hasil pencarian mesin pencari. Dalam mengoptimalkan sebuah situs web, pemilik web diharuskan untuk menulis artikel yang berhubungan dengan kata kunci yang ditarget dan dimasukkan ke dalam situs web.

Pada penulisan artikel diperlukan aplikasi pembantu untuk menghitung kata kunci dan menyarankan penulis konten artikel untuk menambah atau mengurangi kata kunci yang ada pada artikel. Oleh karena itu diperlukan sebuah aplikasi untuk menemukan kata dalam sebuah artikel. Berdasarkan latar belakang tersebut, maka dibuatlah sebuah plugin WordPress untuk menghitung kata kunci pada artikel yang telah ditulis oleh penulis konten.

## II. CONTENT MANAGEMENT SYSTEM, PERHITUNGAN KEPADATAN KATA KUNCI

### A. Content Management System

Content Management System (CMS) adalah perangkat lunak yang dapat memublikasi, mengubah, memodifikasi, mengatur, dan menghapus konten dari sebuah situs web. Sistem CMS menyediakan prosedur untuk mengelola alur kerja dalam lingkungan kolaboratif. Prosedur yang diberikan dapat berupa langkah manual maupun otomatis. CMS biasanya digunakan untuk menjalankan sebuah situs yang mempunyai konten seperti blog ataupun berita [11]. CMS terbukti merupakan sebuah aset penting bagi perusahaan untuk mengelola konten situs web dan portal secara efisien dan efektif (Rockley dan Cooper, 2012).

Situs berbagai perusahaan saat ini biasanya merupakan kombinasi antara konten yang tidak terstruktur dan logika aplikasi untuk membuat aplikasi berbasis web. Aplikasi berbasis web ini mewujudkan interaksi yang lebih personal dengan para pengguna, dan meningkatkan kinerja perusahaan dengan menyediakan layanan mandiri bagi para karyawan, rekan kerja, penyedia barang dan pelanggan. Aplikasi berbasis web yang semakin berkembang, secara tidak langsung meningkatkan peranan dan pentingnya CMS sebagai bagian dari infrastruktur aplikasi perusahaan. Dengan adanya CMS yang terintegrasi dengan sebuah situs web akan memberikan suatu nilai lebih yang akan meningkatkan fungsionalitas dan fleksibilitas dari situs web tersebut, terlebih pada situs web yang tujuan pemanfaatannya sebagai media promosi dan membangun citra konsumen, dimana kontinuitas dan inovasi dalam pemasaran produk-produk secara berkala

dan berkesinambungan sebagai suatu hal yang memegang peranan penting dalam tercapainya target pemasaran.

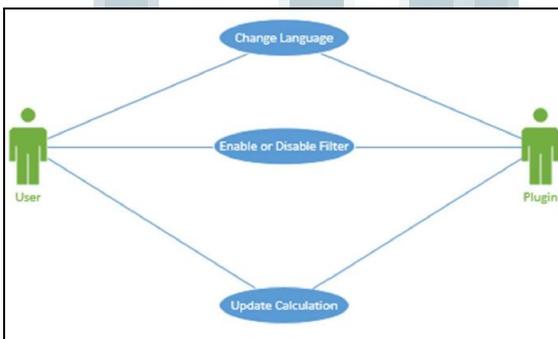
**B. Penghitung Kepadatan Kata Kunci**

Penghitung kepadatan kata kunci adalah aplikasi yang menghitung kata kunci dalam artikel. Hasil yang didapatkan dari aplikasi ini adalah rasio kata kunci dengan kata yang ada pada halaman situs web atau artikel. Manfaat menggunakan penghitung kepadatan kata kunci yaitu penulis konten menjadi sadar bahwa artikel yang ditulis sudah memiliki kepadatan yang cukup. Apabila kepadatan kata kunci yang ada pada artikel sesuai dengan rekomendasi Google yaitu sebanyak 2-3%, maka artikel akan memberikan signal kepada mesin pencari. Signal ini berisi bahwa artikel yang ada pada situs web mengenai topik spesifik, dimana topik tersebut adalah kata kunci yang ditarget [12].

**III. RANCANGAN DIAGRAM UML DAN TAMPILAN ANTARMUKA**

**A. Diagram Use Case**

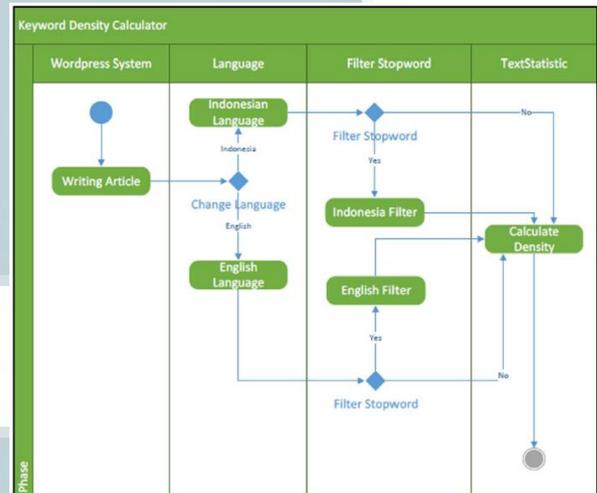
Dalam gambar 2 dapat dijelaskan bahwa diagram use case ini memiliki 2 aktor yaitu User dan Plugin, dan memiliki 3 use case yaitu Change Language, Enable or Disable Filter, dan Update Calculation. Aktor User dapat mengubah bahasa yang digunakan sesuai dengan artikel yang diisikan ke dalam WordPress sehingga plugin dapat menghapus kata-kata umum yang dianggap tidak memiliki makna. Aktor User juga dapat menyalakan atau mematikan fungsi penghapusan kata-kata umum sesuai keinginan User. Aktor User dapat menyuruh plugin untuk memperbarui perhitungan kepadatan kata kunci. Aktor Plugin merupakan plugin yang dibuat dalam penelitian ini. Plugin memiliki pengaturan awal yaitu Bahasa Indonesia dan menggunakan filter stopwords. Plugin akan memperbarui hasil perhitungan kepadatan kata kunci apabila user menggunakan use case update calculation.



Gambar 2. Use Case Plugin

**B. Diagram Activity**

Dalam gambar 3. yang berisi activity diagram tentang bagaimana plugin ini bekerja. Alur pekerjaan dimulai dengan state Writing Article yang berfungsi untuk menuliskan artikel ke dalam aplikasi. Setelah melewati state Writing Article, alur berlanjut ke decision Change Language yang apabila dipilih English, maka alur akan berlanjut ke state English Language sedangkan apabila dipilih Indonesia, maka alur akan berlanjut ke state Indonesian Language. State English Language akan meneruskan alur ke decision Filter Stopword dan apabila user menggunakan Filter Stopword, maka alur akan berlanjut ke state English Filter lalu alur dilanjutkan ke state Calculate Density dimana kepadatan kata kunci akan dihitung sesuai dengan pilihan bahasa dan filter yang dipilih user. Apabila user tidak mau menggunakan filter, maka alur dari decision Filter Stopword akan langsung menuju state Calculate Density yang berarti kata-kata umum juga akan dihitung. State Indonesian Language juga akan meneruskan alur ke decision Filter Stopword dan apabila user menggunakan fitur tersebut, maka alur akan dilanjutkan ke state Indonesia Filter dan dilanjutkan lagi ke state Calculate Density untuk menghitung kepadatan kata kunci, sedangkan apabila user tidak menggunakan fitur tersebut, maka alur akan langsung diteruskan ke state Calculate



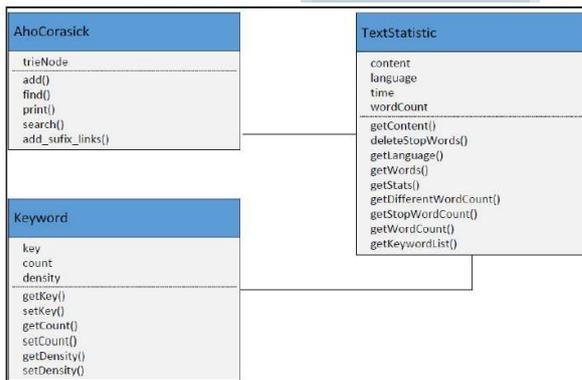
Gambar 3. Activity Diagram

**C. Diagram Class**

Format persamaan merupakan suatu pengecualian di dalam spesifikasi *template* ini. Anda harus menentukan apakah akan menggunakan jenis tulisan Times New Roman atau Symbol (jangan jenis tulisan yang lain). Bila Anda membuat beberapa persamaan berbeda, akan lebih baik bila Anda mempersiapkan persamaan tersebut sebagai gambar dan menyisipkannya ke dalam artikel Anda setelah diberi *style*.

Dalam Gambar 4 dapat dilihat bahwa diagram class ini memiliki 3 class yaitu, TextStatistic,

AhoCorasick, dan Keyword. Class TextStatistic memiliki variabel content, language, time, dan wordCount. Class TextStatistic juga memiliki fungsi getContent, deleteStopWords, getLanguage, getWords, getStats, getDifferentWordCount, getStopWordCount, getWordCount, getKeywordList. Class TextStatistic mempunyai relasi dengan class lainnya yaitu class AhoCorasick dan class Keyword. Relasi ini terbangun ketika class TextStatistic membangun data yang berisi kata kunci dan dimasukkan ke dalam class Keyword dan untuk pencarian kata kunci yang berada pada content dilakukan dengan membangun trie dari class AhoCorasick lalu dilakukan pencarian kata kunci.

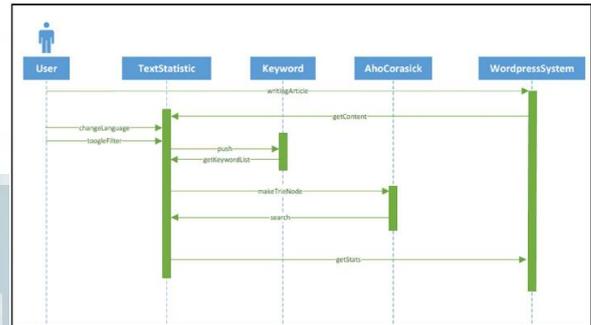


Gambar 4. Class Diagram

D. Sequence Diagram

Dalam gambar 5 terlihat diagram sequence yang berfungsi menjelaskan bagaimana sistem bekerja dari awal hingga berhasil menghitung kepadatan kata kunci. Berawal dari aktor User melakukan writingArticle ke dalam WordPressSystem. Di sini user menuliskan artikel yang akan dihitung oleh plugin ke dalam sistem WordPress. Setelah user menuliskan artikel, TextStatistic mendapatkan content dengan getContent dari WordPressSystem. Di sini class TextStatistic mengambil isi artikel yang sudah dituliskan oleh user. User juga menggunakan changeLanguage untuk mengubah bahasa yang sesuai dengan artikel yang ditulis dan mengirimkan data berupa bahasa yang digunakan ke class TextStatistic. Fungsi toggleFilter yang digunakan oleh user berguna untuk memberikan data berupa flag apakah filter stopwords digunakan atau tidak ke dalam class TextStatistic. Setelah content, bahasa, dan flag filter stopwords didapat, maka TextStatistic akan membangun data yang berisi kumpulan kata kunci dan dimasukkan ke dalam Keyword dengan menggunakan fungsi push. Setelah array Keyword terbentuk, data yang ada pada array tersebut ditarik oleh class TextStatistic dan dikirimkan ke class AhoCorasick untuk membangun trieNode. Setelah dibangun trieNode, class TextStatistic akan mencari kata kunci dengan menggunakan trieNode yang ada dan menghitung kepadatan kata kunci yang ada pada artikel. Setelah dihitung, hasil yang didapat

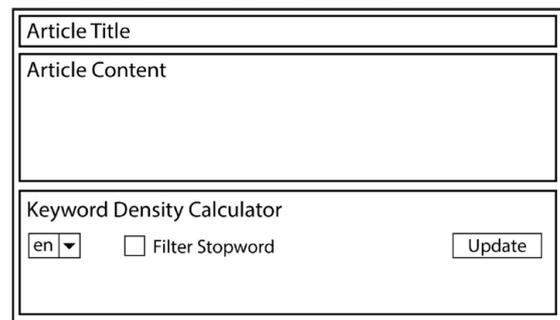
ditampilkan ke dalam WordPressSystem dengan fungsi getStats.



Gambar 5. Sequence Diagram

E. Rancangan User Interface

Sebagai tahap awal pembuatan aplikasi, tampilan antar muka yang akan digunakan dirancang terlebih dahulu. Gambar 6 adalah rancangan tampilan antar muka yang digunakan di dalam aplikasi. Gambar 6 merupakan gambar rancangan tampilan antar muka pada bagian awal. Pada halaman ini terdapat text box di bagian atas aplikasi yang berguna sebagai tempat user mengisikan judul dari artikel yang akan ditulis dan dihitung. Bagian bawah dari text box tersebut terdapat text area di mana user dapat menuliskan isi artikelnya ke dalam text area. Kepadatan kata kunci yang akan dihitung hanyalah kata kunci yang berada pada artikel di dalam text area tersebut. Setelah user menulis artikel, user dapat memilih bahasa yang digunakan sesuai dengan artikel yang ditulis pada combo box yang ada di bagian kiri. Setelah dipilih, apabila user ingin menggunakan filter stopwords, user juga dapat mengaktifkan atau menonaktifkan filter dengan cara menekan check box dengan label Filter Stopword. Setelah mengatur bahasa dan fitur filter stopwords, user dapat menekan tombol update yang berfungsi untuk menghitung kata kunci yang berada di dalam text area.



Gambar 6. Rancangan Halaman Awal

Setelah user menekan tombol update, maka aplikasi akan menampilkan tabel-tabel yang berisi tentang kata kunci, jumlah, dan kepadatan kata kunci tersebut di dalam sebuah artikel seperti gambar 7. Hasil perhitungan tersebut akan ditampilkan ke dalam tabel Single Words. Di dalam tabel Text Statistic

merupakan hasil perhitungan aplikasi mengenai jumlah kata yang ada pada artikel yang ditulis, jumlah stopword, dan juga jumlah perbedaan kata yang ada.

Article Title

Article Content

Keyword Density Calculator

en  Filter Stopword

Text Statistic

Tabel hasil perhitungan Single Words		
Jumlah Kata	Jumlah Perbedaan Kata	Stopwords

Single Words

Tabel hasil perhitungan Single Words		
Jumlah	Kepadatan	Kata Kunci

Meta Keyword Suggestion : 8 kata terbanyak yang cocok untuk dijadikan kata kunci artikel

Gambar 7. Rancangan Halaman Hasil

#### IV. IMPLEMENTASI DAN HASIL

##### A. Pengujian Validasi

Berikut ini merupakan tampilan antar muka aplikasi penghitung kata kunci

Article Title

Permalink: <http://localhost:1234/tester/2015/12/01/article-title/> Edit View Post

Add Media

Article Content

Word count: 2 Draft saved at 12:09:49 am. Last edited by sirta on January 22, 2016 at 8:40 am

Keyword Density Checker Aho-Corasick Algorithm

Language: en  Filter stopwords:

Gambar 8. Tampilan Antar Muka Aplikasi

Gambar 8 merupakan tampilan antar muka pada halaman awal aplikasi dimana user dapat menuliskan judul artikel dan mengisi konten artikel. Di text box paling atas merupakan tempat user mengisi judul artikel dan text area yang berada di bawah text box tersebut merupakan tempat user mengisi konten artikel. Setelah user mengisi judul artikel dan isi artikel, user dapat memilih bahasa yang sesuai dengan isi artikel pada combo box yang berada pada bagian bawah text area. Bahasa yang dapat dipilih pada combo box ini antara Bahasa Inggris atau Bahasa Indonesia. Di samping combo box terdapat check box yang berguna untuk mengaktifkan atau menonaktifkan filter stopwords. Filter ini berguna untuk menghapus semua kata-kata umum yang ada pada isi artikel saat perhitungan

##### B. Validasi

Plugin penghitung kepadatan kata kunci diuji untuk mencari tahu apakah hasil yang didapat dari plugin yang sudah diimplementasi valid atau tidak. Pengujian validitas menggunakan plugin lain yang serupa dan membandingkan hasil yang didapat apakah sesuai atau tidak dengan hasil yang didapatkan dari plugin yang diimplementasi dalam penelitian ini.

Pengujian validitas pada plugin penghitung kepadatan kata kunci yang sudah diimplementasi dilakukan dengan menghitung satu artikel berbahasa Inggris dan satu artikel berbahasa Indonesia dengan jumlah kata yang berbeda. Artikel yang digunakan adalah artikel yang berjumlah 1000 kata, baik Bahasa Inggris maupun Bahasa Indonesia.

##### C. Hasil Validasi

Berikut hasil validasi untuk artikel berbahasa Inggris dapat dilihat pada gambar 8(a) dan untuk artikel berbahasa Indonesia dapat dilihat pada gambar 8(b). Plugin 1 adalah plugin yang diimplementasikan Algoritma Aho-Corasick dalam penelitian ini, sedangkan plugin 2 adalah plugin yang digunakan sebagai tolak ukur ketepatan yaitu SEO Content Helper.

Tabel 1. Hasil Validasi. (a) Artikel Inggris. (b) Artikel Indonesia

Kata Kunci	Jumlah Kata Plugin 1	Jumlah Kata Plugin 2	Kepadatan Kata Kunci Plugin 1	Kepadatan Kata Kunci Plugin 2
Steam	41	41	3,83%	3,70%
Wallet	37	37	3,45%	3,34%
Your	24	24	2,24%	2,17%
Game	15	15	1,40%	1,35%

(a)

Kata Kunci	Jumlah Kata Plugin 1	Jumlah Kata Plugin 2	Kepadatan Kata Kunci Plugin 1	Kepadatan Kata Kunci Plugin 2
Kamu	56	56	5,47%	5,41%
Yang	39	39	3,81%	3,76%
Steam	30	30	2,93%	2,90%
Wallet	21	21	2,05%	2,03%
Membeli	20	20	1,95%	1,93%
Untuk	18	18	1,76%	1,74%
Dengan	18	18	1,76%	1,74%
Akan	17	17	1,66%	1,64%
Game	16	16	1,56%	1,54%

(b)

#### D. Data Validasi

Selain proses validasi ketepatan, plugin penghitung kepadatan kata kunci juga diuji untuk mencari tahu apakah kecepatan pencarian kata kunci dari plugin yang diimplementasi dalam penelitian lebih baik daripada plugin penghitung kepadatan kata kunci lainnya. Pengujian kecepatan menggunakan plugin lain yang serupa tapi tidak menggunakan Algoritma Aho-Corasick melainkan secara brute-force dan membandingkan kecepatan yang dibutuhkan dalam mencari kata kunci.

#### E. Data Validasi

Pengujian validitas pada plugin penghitung kepadatan kata kunci yang sudah diimplementasi algoritma Aho-Corasick dilakukan dengan menghitung lima artikel berbahasa Inggris dan lima artikel berbahasa Indonesia dengan jumlah kata yang berbeda. Artikel yang digunakan adalah artikel yang berjumlah 200 hingga 1000 kata baik Bahasa Inggris maupun Bahasa Indonesia.

#### F. Hasil Validasi Kecepatan

Berikut hasil validasi untuk artikel berbahasa Inggris dapat dilihat pada tabel 2. Plugin 1 adalah plugin penghitung kepadatan kata kunci yang diimplementasi Algoritma Aho-Corasick, sedangkan plugin 2 adalah plugin yang digunakan sebagai tolak ukur kecepatan yaitu SEO Tool.

Tabel 2. Validasi Kecepatan

No.	Jumlah Kata	Bahasa	Waktu Plugin 1 (ms)	Waktu Plugin 2 (ms)	Selisih Waktu
1	200	Indonesia	7	12	5
2	420	Indonesia	7	17	10
3	600	Indonesia	9	18	9
4	800	Indonesia	9	17	8
5	1000	Indonesia	9	18	9
6	500	Inggris	7	15	8
7	675	Inggris	9	16	7
8	720	Inggris	7	21	14
9	800	Inggris	10	21	11
10	1000	Inggris	12	20	8

Waktu yang didapat berasal dari *console log* yang disimpan pada kedua plugin. Waktu dihitung mulai dari ketika plugin mencari dan menghitung kata kunci yang ada pada artikel. Dari hasil yang didapat, total waktu yang dibutuhkan plugin 1 adalah 86 ms, sedangkan total waktu yang dibutuhkan plugin 2 adalah 175 ms. Selisih waktu dari kedua plugin tersebut adalah 89 ms. Setelah selisih waktu dibandingkan dengan waktu yang dibutuhkan plugin 2, maka didapatkan hasil bahwa plugin 1 lebih cepat 50,85% dibandingkan plugin 2.

#### V. SIMPULAN

Simpulan dari penelitian yang dilakukan adalah rancang bangun plugin penghitung kepadatan kata kunci telah dibuat dengan validasi 50,85% lebih cepat dibanding dengan plugin lain yang serupa. Hasil yang didapat dari perhitungan plugin juga valid setelah dibandingkan dengan plugin lain yang serupa. Meskipun begitu, plugin masih kurang baik ketika menghitung kata kunci artikel tanpa kamus stopwords yang lengkap, karena apabila tidak menggunakan *filter stopwords*, maka kata-kata yang ditampilkan lebih sering hanya kata umum, sehingga akan membuat artikel meleset ketika membidik kata kunci untuk SEO.

#### DAFTAR PUSTAKA

- [1] Aho, A. V. dan Corasick, M. J. 1975. Efficient String Matching: An Aid to Bibliographic Search. Dalam *Communications of the ACM*, Volume 18 Number 6.
- [2] Chen, C. C. dan Wang, S. 2013. An efficient multicharacter transition string-matching engine based on the aho-corasick algorithm. Dalam *ACM Transactions on Architecture and Code Optimization (TACO)*, Volume 10 Issue 4.
- [3] Do, P., dkk. 2012. Improving a hierarchical pattern matching algorithm using cache-aware Aho-Corasick automata. Dalam *RACS '12 Proceedings of the 2012 ACM Research in Applied Computation Symposium*, hal. 26-30.
- [4] Gilmore, W. J. 2010. *Beginning PHP and MySQL: From Novice to Professional*, Fourth Edition. Springer Science, New York.
- [5] Google. 2010. *Search Engine Optimization Starter Guide* [online]. Tersedia dalam: <http://static.googleusercontent.com/media/www.google.com/en/webmasters/docs/search-engine-optimization-starter-guide.pdf> [diakses 18 Februari 2016].
- [6] Khan, Z. A. dan Pateriya, R. K. 2012. Multiple Pattern String Matching Methodologies: A Comparative Analysis. Dalam

- International Journal of Scientific and Research Publications, Volume 2 Issue 7.
- [7] Lee, T. H. dan Huang, N. L. 2013. A pattern-matching scheme with high throughput performance and low memory requirement. Dalam IEEE/ACM Transactions on Networking (TON), Volume 21 Issue 4.
- [8] Malaga, R. A. 2008. Worst practices in search engine optimization. Dalam Communications of the ACM - Surviving the data deluge, Volume 51 Issue 12.
- [9] Pao, D., dkk. 2010. A memory-efficient pipelined implementation of the aho-corasick string-matching algorithm. Dalam ACM Transactions on Architecture and Code Optimization (TACO), Volume 7 Issue 2.
- [10] Purnomo, W. D. 2015. Apa itu SEO [online]. Tersedia dalam: <http://semseomanagement.com/apa-itu-seo/> [diakses 18 Februari 2016].
- [11] Rockley, A. dan Cooper, C. 2012. Managing Enterprise Content: A Unified Content Strategy, Second Edition. New Riders, Berkeley.
- [12] Si, S. 2012. Understanding Keyword Density [online]. Tersedia dalam: <https://seo-hacker.com/keyword-density-tutorial/> [diakses 18 Februari 2016].
- [13] Spencer, J. 2015. Content Management System Comparison – WordPress VS Joomla VS Drupal [online]. Tersedia dalam: <http://makeawebsitehub.com/content-management-system-cmscomparison/> [diakses 18 Februari 2016].
- [14] Su, A. J., dkk. 2014. How to Improve Your Search Engine Ranking: Myths and Reality. Dalam ACM Transactions on the Web (TWEB), Volume 8 Issue 2.
- [15] Verens, K. 2010. CMS Design Using PHP and jQuery. Packt Publishing, Birmingham.
- [16] Williams, B., dkk. 2011. Professional WordPress Plugin Development. Wiley Publishing, Indianapolis.
- [17] Wilson, L. S. 2009. WordPress For Dummies, Second Edition. Wiley Publishing, Indianapolis.
- [18] Xing, B. dan Lin, Z. 2006. The impact of search engine optimization on online advertising market. Dalam Proceedings of the 8th international conference on Electronic commerce: The new e-commerce: innovations for conquering current barriers, obstacles and limitations to conducting successful business on the internet.