

**ALGORITMA PENCARIAN *STRING* DENGAN
MENGUNAKAN *BRUTE FORCE* PADA PERPUSTAKAAN
UNIVERSITAS MUHAMMADIYAH JEMBER**

Disusun Untuk Melengkapi dan Memenuhi Syarat Kelulusan Program Strata 1
Jurusan Teknik Informatika Fakultas Teknik
Universitas Muhammadiyah Jember



Oleh :

ACHMAD HADI PRAYOGA

NIM. 13 1065 1188

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH JEMBER**

2019

ABSTRAK

Algoritma Pencarian *String* Dengan Menggunakan *Brute Force* Pada Perpustakaan Universitas Muhammadiyah Jember

¹Achmad Hadi Prayoga (1310651188), ²Reni Umilasari M.Si.
Program Studi Teknik Informatika, Fakultas Teknik
Universitas Muhammadiyah Jember
Email : pea.yoga@gmail.com

Perpustakaan Universitas Muhammadiyah Jember merupakan unit kerja yang ikut menunjang kemajuan dan perkembangan Universitas Muhammadiyah Jember. Perpustakaan merupakan salah satu fasilitas yang dimiliki oleh institusi pendidikan dalam menjalankan layanannya pada bidang pendidikan. Peranan perpustakaan sangat penting dalam menyediakan buku-buku ilmiah untuk kalangan mahasiswa dan dosen dengan upaya memberikan pelayanan yang maksimal.

Pencarian *string* adalah sebuah masalah klasik dalam dunia teknologi informasi. Algoritma *String Matching* dapat dijadikan alternatif dalam menyelesaikan permasalahan tersebut. Untuk melakukan pencarian sebuah data atau informasi tidak terlepas dari pencocokan *string* dimana dari hasil pencocokan inilah akan ditemukan pola kalimat yang dicari. Dalam penelitian ini membahas tentang penerapan algoritma brute force dalam melakukan pencocokan sebuah *string*. Algoritma ini melakukan pencocokan *string* dengan menggeser satu persatu *pattern* dan menyesuaikannya dengan teks hingga antara *pattern* dan teks memiliki pola yang sama.

Aplikasi pencarian membantu perpustakaan dalam memberikan layanan terhadap pengunjung perpustakaan untuk melakukan pencarian terhadap katalog buku perpustakaan dalam waktu singkat. Dari hasil pengukuran tingkat kecepatan pencocokan *string* diperoleh waktu rata-rata dalam pencarian senilai 0.000007528790 *seconds*.

Katakunci : Perpustakaan, Pendidikan, Pencarian, *String*, *Brute Force*.

ABSTRACT

String Search Algorithm Using Brute Force at the Muhammadiyah Jember University Library

¹Achmad Hadi Prayoga (1310651188), ²Reni Umilasari M.Si.
Informatichs Engineering Study Program, Faculty of Engineering

Muhammadiyah University of Jember

Email : pea.yoga@gmail.com

Muhammadiyah Jember University Library is a work unit that supports the progress and development of the University of Muhammadiyah Jember. The library is one of the facilities provided by education in carrying out its services in the field of education. The role of libraries is very important in providing scientific books for students and lecturers who provide maximum service.

String search is a classic problem in the world of information technology. Matching String Algorithms can be used as an alternative to the solution. To do a search for data or information can not be separated from the matching string Where the matching results will be found in the sentence patterns sought. In this study discuss the application of violent algorithms in matching a string. This algorithm makes string matching by shifting one pattern at a time and adjusting it with the text so that between the pattern and the text has the same pattern.

Library assistance search application in providing services to library visitors to search the library book catalog in a short time. From the results of the assessment of the level of string matching obtained by the average time in the search obtained 0.000007528790 seconds.

Keyword : *Library, Education, Search, String, Brute Force.*

PENDAHULUAN

1. Latar Belakang

Perpustakaan merupakan salah satu fasilitas yang dimiliki oleh institusi pendidikan dalam menjalankan layanannya pada bidang pendidikan. Peranan perpustakaan sangat penting dalam menyediakan buku-buku ilmiah untuk kalangan mahasiswa dan dosen dengan upaya memberikan pelayanan yang maksimal. Oleh karena itu pihak perpustakaan selalu menambah jumlah buku yang disediakan, sehingga tahun demi tahun berjumlah sangat banyak dan juga jumlah anggota perpustakaan bertambah dari tahun ke tahun. Dengan banyaknya jumlah buku maka petugas perpustakaan tentu akan mengalami kesulitan dalam hal pencarian data katalog buku yang terdapat pada perpustakaan.

Pencarian *string* adalah sebuah masalah klasik dalam dunia teknologi informasi. Kebutuhan untuk mencari informasi yang berguna dalam suatu data yang besar adalah salah satu masalah klasik ini. Selain untuk mencari informasi yang tepat, pencarian *string* juga diharapkan untuk dapat menuntaskan pekerjaannya dengan cepat. Sehingga hasil dari pencarian itu dapat digunakan secara tepat, cepat, dan efisien. Contoh dari penggunaan pencarian *string*, antara lain adalah *search engine* untuk *website*, pencarian kata di suatu dokumen, pembuatan program kamus, dan *data mining*.

Algoritma *String Matching* dapat dijadikan alternatif dalam menyelesaikan permasalahan tersebut. Terdapat banyak algoritma *string matching* yang telah dikembangkan diantaranya seperti *Brute Force*, *Knuth-Morris-Pratt*, *Boyer-Moore*, *Karp-Rabin* dan lain sebagainya. Dalam penelitian ini, penulis menggunakan salah satu variasi dari algoritma *string matching* yaitu algoritma *Brute Force*. Algoritma *Brute Force* melakukan pengecekan pada setiap posisi di dalam *text* dari karakter pertama hingga karakter terakhir. Kemudian seusai proses pencocokan karakter pertama terjadi, proses *shift* dilakukan tepat satu posisi ke arah kanan (Charras & Lecroq, 2004).

Penggunaan algoritma *Brute Force* sudah banyak dilakukan karena algoritma ini telah terbukti mampu menyelesaikan beberapa permasalahan dalam penelitian ilmiah. Algoritma *Brute Force* telah diterapkan oleh Lin dalam menyelesaikan penelitian perbandingan kesamaan koleksi dokumentasi. Penelitian lain yang menggunakan pendekatan algoritma *Brute Force* pernah juga dilakukan oleh Matej Crepinsek beserta rekan-rekannya dalam pengeksraksian bentuk tata bahasa dalam sebuah pemrograman. Algoritma *brute force* adalah algoritma untuk mencocokkan *pattern* dengan semua *teks* antara 0 dan n-m. Karakteristik *brute force* selalu berpindah tepat 1 langkah depan, dapat dilakukan pada urutan apa saja, karakter yang terjadi diharapkan 2^n (Fernando, 2009).

Pada penelitian ini, kami akan mencoba membuat suatu algoritma baru untuk mencari *string* dengan berbasis algoritma *brute-force* dan menambahkan kemampuan untuk memproses karakter dalam *string*.

Dari uraian permasalahan penulis memilih judul “***Algoritma Pencarian String Dengan Menggunakan Brute Force Pada Perpustakaan Univeristas Muhammadiyah Jember***”. Informasi yang dihasilkan oleh sistem ini adalah judul yang pada abstrak/sinopsis memiliki kemiripan dengan pencarian *user*.

2. Rumusan Masalah

Rumusan masalah yang timbul adalah:

1. Berapa tingkat akurasi sistem dalam melakukan pencarian *string* dengan menggunakan algoritma *Brute Force*.
2. Berapa rata-rata waktu (*seconds*) kecepatan algoritma *Brute Force* dalam melakukan pencarian *string*?

3. Tujuan Penelitian

Tujuan penulisan ini adalah untuk mempermudah pengguna dalam melakukan pencarian menggunakan *search engine*, diantaranya :

1. Mengukur tingkat akurasi sistem dalam melakukan pencocokan/pencarian *string* menggunakan algoritma *brute force*.

2. Menghitung waktu rata-rata dalam melakukan pencarian *string* dengan mengimplementasikan algoritma *brute force*.

4. Batasan Penelitian

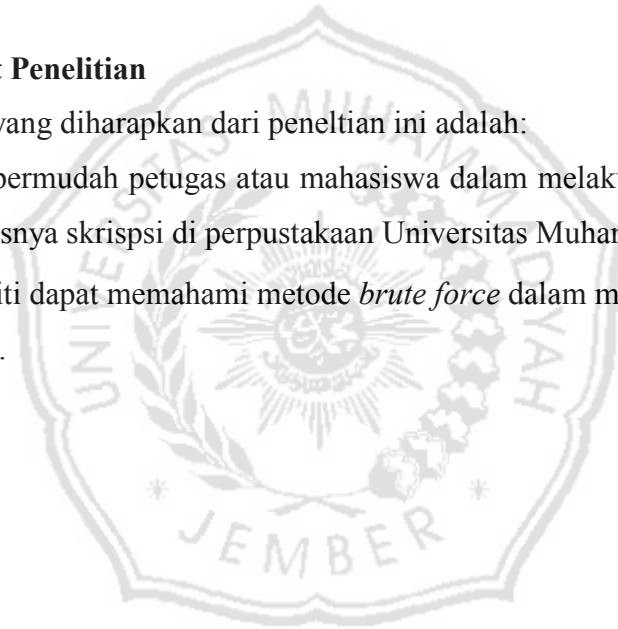
Agar pembahasan dalam skripsi ini lebih terarah maka dilakukan pembatasan-pembatasan seperti berikut:

1. Membangun sebuah aplikasi pencarian sebuah kata atau kalimat dengan algoritma *brute force*.
2. Data yang digunakan hanya tugas akhir mahasiswa Teknik Informatika.

5. Manfaat Penelitian

Manfaat yang diharapkan dari penelitian ini adalah:

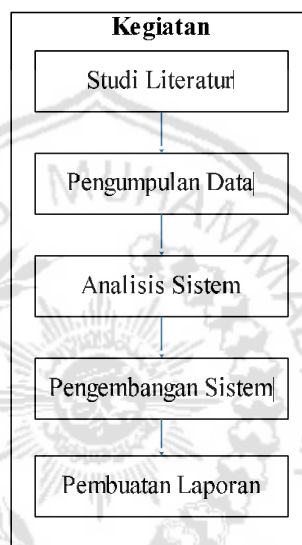
1. Mempermudah petugas atau mahasiswa dalam melakukan pencarian buku khususnya skripsi di perpustakaan Universitas Muhammadiyah Jember.
2. Peneliti dapat memahami metode *brute force* dalam melakukan pencarian *string*.



METODOLOGI PENELITIAN

1. Metode Penelitian

Untuk membantu dalam penyusunan penelitian ini, maka perlu adanya susunan kerangka kerja (*frame work*) yang jelas tahapan-tahapannya. Kerangka kerja ini merupakan langkah-langkah yang akan dilakukan dalam penyelesaian masalah yang akan dibahas. Adapun kerangka kerja penelitian yang digunakan seperti terlihat pada Gambar 3.1:



Gambar 3.1 Kerangka Kerja Penelitian

Berdasarkan kerangka kerja penelitian yang telah digambarkan, maka dapat diuraikan pembahasan masing-masing tahap dalam penelitian adalah sebagai berikut :

1. Studi Literatur

Pada tahap ini dilakukan pencarian landasan-landasan teori yang diperoleh dari berbagai buku dan juga internet untuk melengkapi perbendaharaan konsep dan teori, sehingga memiliki landasan dan keilmuan yang baik dan sesuai.

2. Pengumpulan Data

Pada tahap ini dilakukan proses pengumpulan data dengan metode wawancara dan observasi untuk melakukan pengamatan dan analisa terhadap proses pencairan buku yang sedang berjalan pada Perpustakaan

Universitas Muhammadiyah Jember sehingga mendapatkan data dan informasi yang dibutuhkan oleh peneliti.

3. Analisis Sistem

Pada tahap ini dilakukan identifikasi masalah pada sistem yang sedang berjalan. Dengan demikian, diharapkan peneliti dapat menemukan kendala-kendala dan permasalahan yang terjadi pada proses pencarian buku, skripsi di Perpustakaan Universitas Muhammadiyah Jember sehingga peneliti dapat mencari solusi dari permasalahan tersebut.

4. Pengembangan Sistem

Pada Tahap ini dilakukan Pengembangan sistem dengan menggunakan model *waterfall*.

5. Pembuatan Laporan

Pada tahapan ini dilakukan pembuatan laporan yang disusun berdasarkan hasil penelitian dengan menggunakan teknik pengumpulan data primer dan sekunder sehingga menjadi laporan penelitian yang dapat memberikan gambaran secara utuh tentang sistem yang sedang dibangun.

2. Analisis Sistem

Tahapan yang dilakukan untuk menghasilkan pemahaman yang menyeluruh terhadap kebutuhan sistem sehingga diperoleh tugas-tugas yang akan dikerjakan sistem disebut analisis sistem. Tahapan ini dilakukan agar pada saat proses perancangan aplikasi tidak terjadi kesalahan yang berarti. Ada dua tahapan analisis dalam tugas akhir ini yaitu: analisis masalah dan analisis persyaratan.

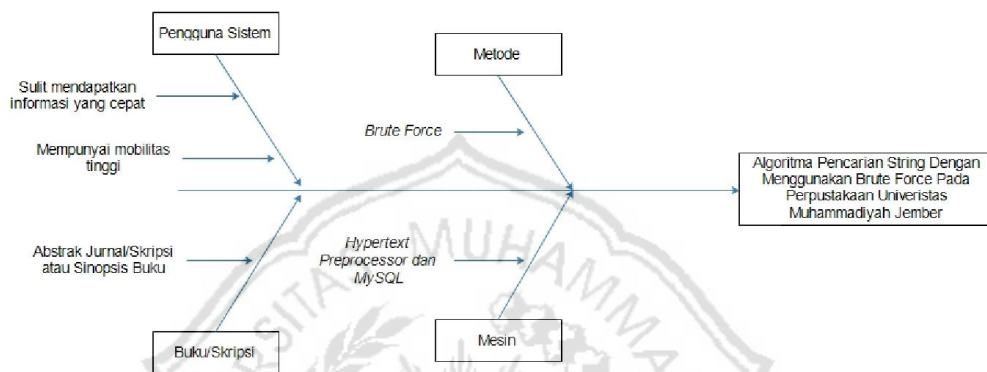
Memahami kelayakan masalah akan dibahas dianalisis masalah juga untuk menjelaskan fungsi-fungsi yang ditawarkan dan mampu dikerjakan sistem akan dibahas dianalisis persyaratan.

3. Analisis Masalah

Diagram Ishikawa (*fishbone diagram*) biasanya disebut juga diagram sebab akibat yang biasanya digunakan untuk mengidentifikasi masalah yang ada pada sistem yang akan dirancang. Dengan diagram ini dapat mengidentifikasi, mengeksplorasi dan menggambarkan suatu masalah dengan mengikutsertakan

sebab dan akibat dari permasalahan. Dengan demikian proses pencapaian sistem akan sangat dibantu dengan adanya Diagram Ishikawa ini.

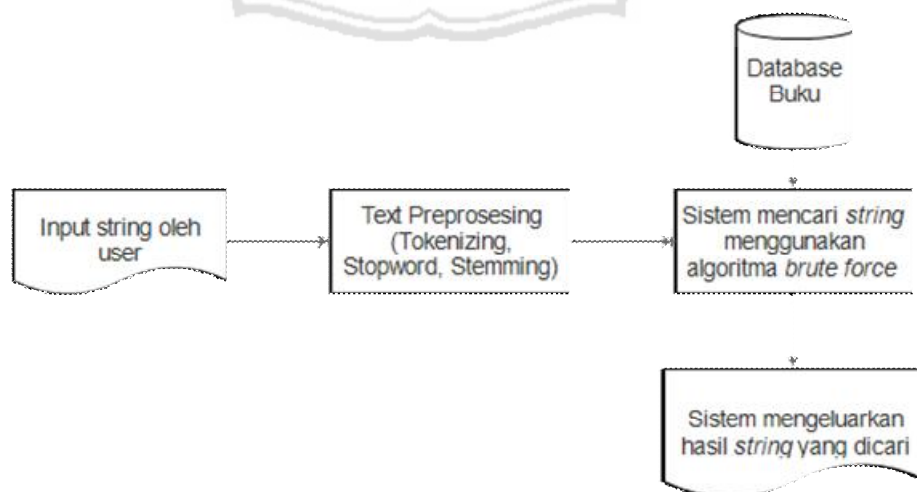
Permasalahan utama dari sistem yang akan dikembangkan pada tugas akhir ini adalah bagaimana mendapatkan pencarian *string* (buku). Selanjutnya masalah ini diuraikan ke beberapa kategori, antara lain : pengguna system, buku, mesin, dan metode. Maka dari keseluruhan kategori tersebut dijelaskan melalui diagram Ishikawa yang dapat dilihat pada Gambar 3.1.



Gambar 3.1. Diagram Ishikawa untuk Analisis Permasalahan Sistem

4. Preprocessing

Preprocessing adalah tahapan awal dalam mengolah data yang di-*input* sebelum benar-benar memasuki proses tahapan utama pada suatu sistem. Untuk prosesnya dapat dilihat pada gambar 3.2 sebagai berikut.



Gambar 3.2. Preprocessing Pencarian *String*

5. Analisis dan Kebutuhan Non-Fungsional

Analisis kebutuhan non fungsional menggambarkan kebutuhan sistem yang menitikberatkan pada properti perilaku yang dimiliki oleh sistem, diantaranya kebutuhan perangkat lunak, perangkat keras sebagai bahan analisis kekurangan dan kebutuhan yang harus dipenuhi dalam perancangan sistem yang akan diterapkan.

1. Analisis Kebutuhan Perangkat Lunak

Perangkat keras komputer tidak berarti tanpa perangkat lunak begitu juga sebaliknya. Jadi perangkat lunak dan perangkat keras saling mendukung satu sama lain. Perangkat keras hanya berfungsi jika diberikan instruksi-intruksi kepadanya. Instruksi-instruksi inilah disebut dengan perangkat lunak. Dalam penelitian ini penyusun menggunakan beberapa perangkat lunak, antara lain :

1. PHP Version 5.6.36
2. MariaDB version 10.1.32
3. *Editor Notepad++*
4. Untuk membuat laporan (Ms.Word 2010)

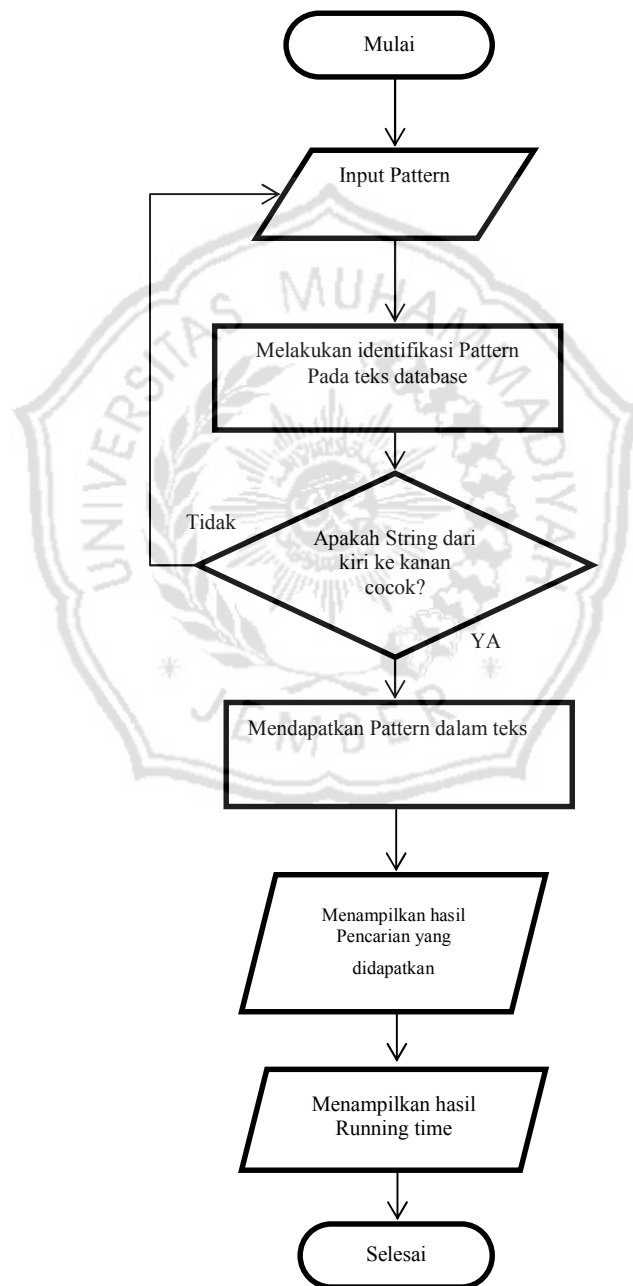
2. Analisis Kebutuhan Perangkat Keras

Berikut ini adalah spesifikasi perangkat keras minimum yang mendukung aplikasi pencarian informasi obat dengan baik, yaitu :

1. Perangkat Keras Aplikasi *Backend*
 - Processor : *Prosesor Intel® Core™ i3-9300 Minimum 2.0 GHZ*
 - Memori : *Minimum 2048 MB*
 - Hardisk : *Minimum Kapasitas 20 GB*
2. Perangkat keras Aplikasi *Frontend*
 - Resolusi : 240 x 320 pixels
 - Memori : 100 MB
 - CPU : 300 MHZ

6. *Flowchart* Sistem

Flowchart merupakan bagan yang memperlihatkan urutan maupun hubungan antar proses beserta keterangannya yang dinyatakan dengan simbol. Dengan demikian setiap simbol menggambarkan proses tertentu. Sedangkan antara proses digambarkan dengan garis penghubung. Untuk melihat *Flowchart* dari sistem yang akan dibangun perhatikan Gambar 3.4 berikut:



Gambar 3.4. *Flowchart* proses *Brute Force*

7. Kesimpulan

Dari hasil penelitian yang telah dijelaskan pada bab sebelumnya maka dapat disimpulkan sebagai berikut:

1. Tingkat akurasi metode *Brute Force* dalam pencocokan *string* menghasilkan tingkat rata-rata akurasi dari beberapa percobaan adalah 95% dan sistem yang dibangun memberikan layanan terhadap pengunjung perpustakaan untuk melakukan pencarian terhadap katalog buku perpustakaan dalam waktu singkat.
2. Dari hasil pengukuran tingkat kecepatan pencocokan *string* diperoleh waktu rata-rata dalam pencarian senilai 0.000007529 *seconds* dan rata-rata tingkat akurasi mencapai 94%.

8. Saran

Untuk pengembang berikutnya adalah sebagai berikut:

1. Sistem dapat dikembangkan dengan berbasis *mobile* atau *android*.
Dapat dikembangkan dengan mengimplementasikan algoritma pencarian kata yang lain (*buyer moree*, *straightforward matching* dan *Finite automata*) untuk mendapatkan optimal waktu dalam pencarian *string*.

DAFTAR PUSTAKA

- Anhar. 2012. "Panduan Menguasai PHP da MySQL Secara Otodidak". Yogyakarta: Andi.
- Charras, C. & Lecroq, T. 2004. Handbook of Exact *String*-Matching Algorithms. London: King's College Publications.
- Fernando, Hary., 2009. "Perbandingan dan pengujian beberapa algoritma pencocokan *string*." Makalah IF2251, pp.1-7.
- Kadir, A. 2013. Pengertian PHP. Tersedia dalam : Buku Pintar Programmer. Pemula PHP. Yogyakarta. Mediakom.
- Kadir, A. 2013. Pemrograman Database MySQL untuk pemula. Yogyakarta. MediaKom.
- Madcoms. 2009. Aplikasi Program PHP + MySQL untuk membuat website interaktif. Yogyakarta: Andi.
- Madcoms. 2016. Notepad++. <http://eprints.polsri.ac.id/19-10-2019>.
- Mirza, P, S. 2013. "Tujuh Langkah Praktis Pembangunan Basis Data". Jakarta: PT. Elex Media Komputindo.
- Nurhayati, S., 2010, "Text Mining", Implementasi Text Mining Untuk Klasifikasi Kesenian Tradisional Dengan Metode Nbc (Naïve Bayes Classifier), Fakultas Teknik dan Ilmu Komputer Universitas Komputer Indonsria. Bandung.
- Sarno, R., dkk, 2012, Semantic Search Pencarian Berdasarkan Konten, Andi Yogyakarta.
- Silitonga, P.M., 2014, Statistika Teori Dan Aplikasi Dalam Penelitian Edisi Kedua, Fakultas Matematika dan Ilmu Pengetahuan Alam, UNIMED, Medan.
- Sulianta, Feri. 2010. Seacrh Engine Pilihan untuk Berbagai Kebutuhan. Jakarta: Elex Media Komputindo.
- Sutarman. 2012. "Buku Pengantar Teknologi Informasi". Jakarta: Bumi Aksara.
- Sutarno, NS, 2006, Perpustakaan dan Masyarakat, Bogor: Sagung Seto,
- Wayteg, T., 2013, " Kamus Internet ",1st ed, Surabaya: Liris.