

Implementasi Graf dalam *Search Engine*

Gilang Julian Suherik - 13512045
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
gilang.9h@gmail.com

Abstrak—Graf adalah gabungan dari himpunan tidak kosong dari simpul(vertex) yang merepresentasikan sebuah objek dan himpunan sisi(edges) yang merepresentasikan hubungan antar dua buah simpul dengan relasi tertentu. Teori tentang graf dapat diaplikasikan untuk berbagai hal dalam kehidupan sehari-hari, misalnya untuk memilih jalur terpendek ke suatu tempat, memilih jalur yang paling efektif bagi pedagang, pembagian jadwal pelajaran bagi siswa, perancangan IC, dan lain-lain. Namun tahukah anda bahwa graf juga digunakan dalam *search engine*? Dalam makalah ini akan dibahas bagaimana implementasi graf dalam *search engine*, khususnya akan dibahas mengenai *search engine* yang digunakan oleh *Google* dan *Facebook*.

Index Terms—*Graph, search engine, google, facebook.*

I. PENDAHULUAN

Informasi merupakan salah satu kebutuhan yang tidak dapat terlepas dari kehidupan manusia. Setiap harinya orang-orang mendengarkan radio, membaca koran, dan menonton tv. Kegiatan-kegiatan tersebut dilakukan untuk memenuhi kebutuhan mereka akan informasi. Dengan perkembangan teknologi yang berjalan sangat cepat, media-media informasi seperti koran dan radio mulai ditinggalkan. Saat ini orang dewasa, bahkan anak-anak sekalipun bisa mendapatkan informasi dengan mudah menggunakan *gadget* yang mereka miliki.

Salah satu cara memperoleh informasi menggunakan *gadget* adalah dengan menggunakan *search engine*. Salah satu *search engine* yang banyak digunakan oleh masyarakat dunia saat ini adalah *google search*. Menurut *statista.com* tercatat pada bulan Desember 2012 sebanyak 1,17 milyar orang telah menggunakan *google search*, serta dari statistik *google.com* sebanyak 36.000 *search request* tercatat setiap detiknya.

Media lain yang digunakan oleh orang-orang untuk memperoleh informasi adalah jejaring sosial. Dengan jejaring sosial, setiap orang dapat mengetahui informasi-informasi serta aktivitas terbaru yang dilakukan oleh keluarga, saudara, serta teman-teman mereka. Jejaring sosial yang memiliki pengguna terbanyak adalah *Facebook*. Menurut *expandedramblings.com*, pengguna *Facebook* yang tercatat pada tanggal 30 Oktober 2013 berjumlah 1,19 milyar orang.

Walaupun *facebook* bukan merupakan *search engine*,

namun *Facebook* memiliki fitur *search* yang tentu saja mengimplementasikan kerja *search engine*.

Mengapa kita membutuhkan *search engine*? Karena dengan menggunakannya kita dapat memperoleh suatu informasi yang kita butuhkan dengan waktu yang sangat singkat. Cara kerja *search engine* pada awalnya hanyalah mencari suatu kata atau kalimat yang match dengan kata atau kalimat dalam dokumen-dokumen yang tersebar di internet. Namun saat ini, cara kerja *search engine* menjadi lebih baik, salah satunya bisa memberikan kita *suggestion* untuk pencarian lain yang mungkin awalnya tidak terlintas dalam benak kita untuk mencarinya, namun berhubungan dengan apa yang kita cari sebelumnya. Kecanggihan *search engine* ini dipengaruhi oleh salah satunya penggunaan graf di dalamnya.

II. DASAR TEORI

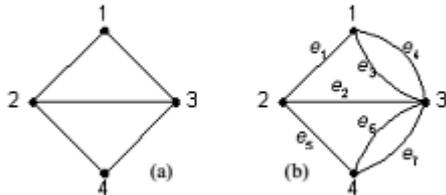
Graf adalah gabungan dari himpunan tak kosong dari simpul(vertex) yang merepresentasikan objek dengan himpunan sisi(edges) yang merepresentasikan hubungan antar dua buah simpul dengan relasi tertentu. Secara matematis, graf G didefinisikan sebagai pasangan himpunan (V,E) , dengan V adalah himpunan tidak kosong dari simpul (vertex/node) = $\{v_1, v_2, \dots\}$ dan E adalah himpunan sisi (edges/arcs) yang menghubungkan dua buah simpul = $\{e_1, e_2, \dots\}$. Graf G tersebut dapat dinotasikan dengan $G = (V,E)$. Sedangkan untuk sisi, misalkan e_1 adalah sisi yang menghubungkan antara simpul v_1 dan v_2 , maka e_1 dinotasikan dengan $e_1 = (v_1,v_2)$.

Dengan definisi di atas, kita ketahui bahwa sebuah graf harus memiliki paling tidak sebuah simpul. Namun sebuah graf boleh tidak memiliki sisi, graf yang hanya memiliki simpul dan tidak memiliki sisi disebut graf kosong / *null graph*. Graf kosong menunjukkan ketiadaan hubungan antar objek-objek di dalamnya.

2.1 Jenis-jenis Graf

Graf terbagi atas beberapa jenis. Berdasarkan ada/tidaknya gelang atau sisi ganda dalam suatu graf, graf dibedakan menjadi dua jenis.

1. Graf sederhana (*simple graph*)
Graf sederhana adalah graf yang tidak memiliki gelang maupun sisi ganda.
2. Graf tak-sederhana (*unsimple graph*)
Graf tak-sederhana adalah graf yang mengandung gelang atau sisi ganda.

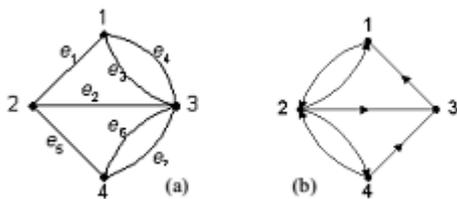


Gambar 2.1.1 Graf sederhana (a), dan Graf tak-sederhana (b)

[1] source: Munir, Rinaldi. 2008. "Diktat Kuliah IF 2091 Struktur Diskrit". Bandung:Program Studi Teknik Informatika STEI ITB.

Sedangkan berdasarkan orientasi arah pada sisi-sisinya, graf dibedakan menjadi dua jenis:

1. Graf tak-berarah (*undirected graph*)
Graf tak berarah adalah graf yang sisi-sisinya tidak memiliki orientasi arah.
2. Graf berarah (*directed graph/digraph*)
Graf berarah adalah graf yang sisinya memiliki orientasi arah.



Gambar 2.2.2 Graf tak-berarah (a), dan Graf berarah(b)

[1] source: Munir, Rinaldi. 2008. "Diktat Kuliah IF 2091 Struktur Diskrit". Bandung:Program Studi Teknik Informatika STEI ITB.

2.2 Terminologi Graf

1. Ketetanggaan (*Adjacent*)
Dua buah simpul dikatakan bertetangga jika keduanya terhubung oleh satu atau lebih sisi.
2. Bersisian (*Incidency*)
Sebuah sisi e dan simpul v dikatakan bersisian jika simpul v tersebut terhubung pada simpul lain melalui sisi e .
3. Simpul Terpencil (*Isolated Vertex*)
Simpul terpencil adalah simpul yang tidak bersisian dengan sisi manapun di dalam sebuah graf.

4. Graf Kosong (*Null Graph* atau *Empty Graph*)
Graf kosong adalah graf yang himpunan sisinya merupakan sebuah himpunan kosong.

5. Derajat (*Degree*)
Derajat suatu simpul adalah sama dengan jumlah sisi yang bersisian dengan simpul tersebut.

6. Lintasan (*Path*)
Lintasan adalah barisan berselang-seling antara simpul dan sisi yang berbentuk $v_0, e_1, v_1, e_2, v_2, \dots, e_n, v_n$ sedemikian sehingga $e_1 = (v_0, v_1)$, $e_2 = (v_1, v_2)$ dan seterusnya adalah sisi-sisi dari graf G .

7. Terhubung (*Connected*)
Dua buah simpul v_1 dan v_2 dikatakan terhubung jika ada lintasan dari v_1 menuju ke v_2 ataupun sebaliknya.

2.3 Graf Khusus

1. Graf Lengkap (*Complete Graph*)
Graf lengkap adalah graf sederhana yang setiap simpulnya memiliki sisi ke semua simpul lainnya. Jika G adalah graf dengan n buah simpul, maka setiap simpulnya bersisian dengan $n-1$ buah sisi.
2. Graf Lingkaran (*Circle Graph*)
Graf lingkaran adalah graf sederhana yang setiap simpulnya memiliki derajat dua (setiap simpulnya hanya bersisian dengan dua buah sisi).
3. Graf Teratur (*Regular Graph*)
Graf teratur adalah graf yang setiap simpulnya memiliki derajat yang sama.

III. SEARCH ENGINE

Search Engine adalah perangkat lunak yang digunakan untuk mencari dokumen di internet yang relevan dengan kata kunci tertentu.

3.1 Sejarah *Search Engine*

Search engine pertama bernama Archie, dibuat oleh tiga orang mahasiswa dari McGill University pada tahun 1990. Cara kerja dari *search engine* ini cukup sederhana, mendata dokumen-dokumen yang ada dalam FTP-FTP di dunia, mengurutkannya secara alfabetikal, lalu mencocokkan query string yang dimasukkan pengguna dengan nama-nama dokumen yang telah didata tersebut. Kerja dari Archie masih sangat sederhana karena hanya mencocokkan antara string yang dimasukkan pengguna dengan nama dokumen seperti proses *searching* biasa, dan langsung menampilkan nama dokumen yang *match* dengan *query*.

Pada tahun 1993 *University of Nevada System Computing Services group* menciptakan Veronica. Cara kerja Veronica hampir sama persis dengan Archie, bedanya Veronica dibuat untuk melakukan *searching* terhadap Gopher, bukan FTP.

Masih di tahun 1993, Matthew Gray berhasil menemukan World Wide Web Wanderer, sebuah robot pertama yang dapat mengikuti perkembangan sebuah server web dan mencatat URL yang ada di dalamnya serta memasukkannya ke sebuah *database* yang bernama Wandex.

Selanjutnya, ide robot ini berkembang menjadi *spider/web crawler*, robot yang menjelajahi seluruh link yang terdapat dalam sebuah alamat *starter* dan menampilkan judul serta URL halaman yang berhasil dijelajahnya. Kelemahan dari *spider* pada masa itu, yaitu menampilkan URL dan judul dari halaman yang dijelajahnya berdasarkan urutan halaman pertama yang berhasil dijelajahi, sehingga banyak hasil penjelajahan yang ditampilkan tidak relevan dengan *query* yang diberikan.

3.2 Search Engine Masa Kini

Seiring dengan berkembangnya algoritma pada *search engine*, hasil yang ditampilkan oleh *search engine* tidak semena-mena halaman web yang memiliki judul yang sama dengan *query* atau halaman web yang berhasil dikunjungi *spider* terlebih dahulu.

Jika kita melihat proses *searching* yang dilakukan oleh google, proses *searching* diawali dengan metode yang sama dengan yang digunakan di masa lalu, yaitu dengan proses *crawling*. Google melakukan *crawling* dari satu halaman ke halaman lainnya, lalu menyimpan halaman-halaman tersebut berdasarkan isi dengan faktor-faktor tertentu dalam sebuah *database* raksasa yang berukuran lebih dari 100.000.000 gigabytes.

Selanjutnya saat user mulai mengetikkan *keyword*, di sinilah proses yang sebelumnya tidak diimplementasikan mulai dilakukan. Pertama, google akan melakukan pengecekan terhadap ejaan dari *query* yang diberikan dan melakukan *suggestion* atau *auto correction*. Proses pencarian tidak hanya berhenti di kata ataupun kalimat yang diberikan, google akan mencari juga kata yang menjadi sinonim dari *query*. Setelah *query* beserta sinonimnya didapat, *search engine* akan mulai mencari dan mengambil data-data ataupun dokumen dan arsip yang memiliki hubungan dengan *query* yang diberikan.

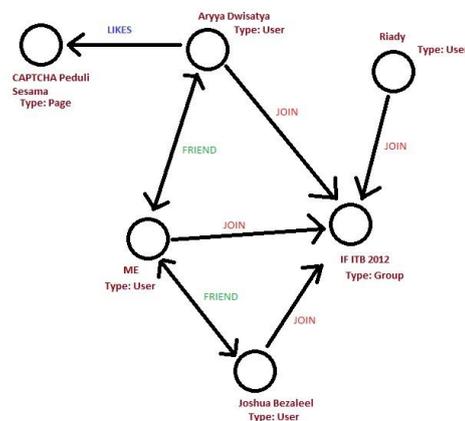
Terakhir, google akan melakukan *sorting* terhadap hasil yang telah didapat berdasarkan banyak sekali faktor, factor-faktor tersebut meliputi bagaimana kualitas suatu halaman web, bagaimana ratingnya bagi pengguna, bagaimana sifat informasi yang ada di dalamnya (baru atau sudah lama), hingga tingkat kesesuaian bagi pengguna (akan berbeda hasil pencarian oleh anak-anak dan orang tua).

IV. PENGGUNAAN GRAF PADA SEARCH ENGINE

Di samping proses *searching* dan *sorting* yang dilakukan *search engine* untuk membuat pencarian menjadi lebih efektif dan sesuai keinginan, implementasi graf dalam *search engine* menghadirkan fitur lain dalam proses pencarian. Dalam bahasan kali ini akan diperlihatkan bagaimana implementasi graf dalam *search engine* yang digunakan Google dan Facebook.

4.1 Graph Search pada Facebook

Facebook melakukan penyimpanan datanya mungkin tidak *se-simple* yang orang kira. Di balik penggunaannya yang mudah dan penyajian data yang *user-friendly*, ternyata Facebook mengimplementasikan teori graf dalam *database* nya. Penggunaan graf ini, selain digunakan untuk memilih informasi dalam proses pencarian, juga digunakan untuk memilih informasi mana saja yang sekiranya relevan untuk ditampilkan di *wall* pengguna karena Facebook tidak akan menampilkan informasi seperti status, foto, dan kegiatan yang dilakukan pengguna asing yang tidak memiliki hubungan apapun dengan kita.



Gambar 4.1.1 Representasi Graf dalam Facebook

Implementasi graf dalam Facebook memungkinkan kita untuk melakukan *searching* terhadap suatu objek yang memenuhi kriteria atau relasi tertentu terhadap objek lain. Misalnya saja dalam contoh di atas, kita dapat melakukan *query* untuk mencari tahu siapa saja teman kita di Facebook yang tergabung dalam grup “IF ITB 2012”, atau siapa saja teman kita yang menyukai halaman “CAPTCHA Peduli Sesama”.



Gambar 4.1.2 Implementasi Graf Search dalam Facebook
source: <http://facebook.com>



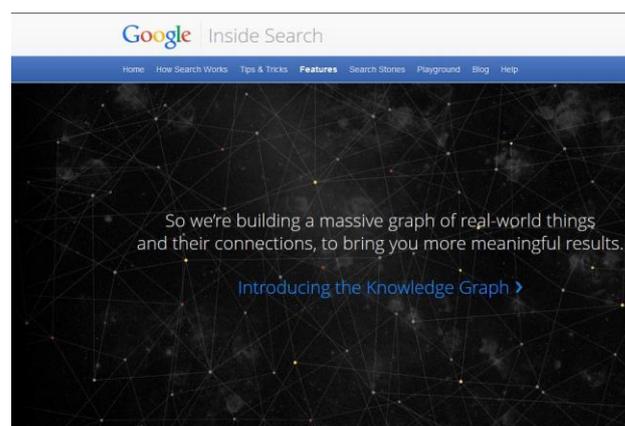
Gambar 4.1.3 Implementasi Graf Search dalam Facebook
source: <http://facebook.com>

Dari contoh di atas, saat kita memasukkan *query* “My Friends who belong to IF ITB 2012”, Facebook akan melakukan penelusuran terlebih dahulu dengan *keyword* “My friends” lalu membuat list siapa saja *user* yang merupakan teman kita di Facebook. Hal tersebut dilakukan dengan cara melihat sisi (edges) “Friend” yang terhubung dengan simpul (vertex) “ME” dalam graf yang ada di *database*. Kemudian setelah didapat sebuah list yang berisi siapa saja teman kita, Facebook akan melanjutkan pencarian ke *keyword* selanjutnya yaitu “who belong to IF ITB 2012”. *Keyword* ini akan mem-filter list yang sebelumnya telah dibuat dengan melihat siapa saja *user* yang memiliki panah keluar menuju ke simpul “IF ITB 2012” (simpul *user* yang bertetangga dengan simpul IF ITB 2012) dan membuat list baru yang memenuhi kriteria tersebut. Setelah melalui proses-proses di atas, akhirnya ditampilkan simpul-simpul yang memenuhi kriteria.

Hal serupa juga dilakukan saat kita memberikan *query* “My Friends who likes Captcha peduli sesama”. Facebook akan mencari terlebih dahulu simpul-simpul yang memenuhi kriteria “My Friends” sama seperti langkah pada *query* sebelumnya. Setelah ditemukan list *user* yang memenuhi kriteria “My Friends”, Facebook akan mencari simpul mana saja yang terdapat dalam list tersebut yang bertetangga dengan simpul halaman “CAPTCHA peduli sesama”.

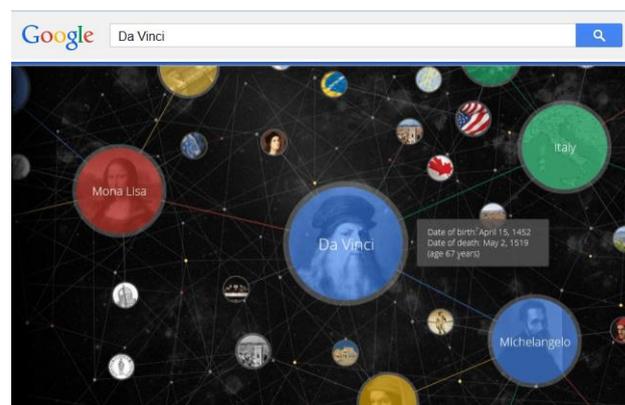
4.2 Knowledge Graph pada Google

Tidak hanya Facebook, Google juga memiliki fitur pencarian yang mengimplementasikan teori graf di dalamnya. Jika Facebook menggunakan graf sebagai representasi *user*, grup, dan halaman yang ada dalam *database*-nya, Google tidak mau kalah dengan mengatakan mereka merepresentasikan seluruh objek yang ada di dunia dengan graf yang sangat luas. Sehingga saat seseorang memberikan suatu *query*, yang diharapkan bukanlah suatu permintaan kata yang dibalas dengan kata itu sendiri. Melainkan sebuah kata yang dibalas dengan sebuah objek yang memiliki makna, objek yang sama dengan objek aslinya yang memiliki relasi terhadap objek lain.



Gambar 4.2.1 Google Knowledge Graph
source: <http://google.com/insidesearch>

Misalnya kita memberikan permintaan terhadap pencarian “Da Vinci”, yang diharapkan bukanlah kita mendapatkan dokumen ataupun halaman yang hanya mengandung nama Da Vinci di dalamnya, melainkan kita mendapatkan gambaran tentang siapa itu Da Vinci, apa perannya bagi dunia, dan apa yang pernah dilakukannya. Lihat gambar di bawah ini:



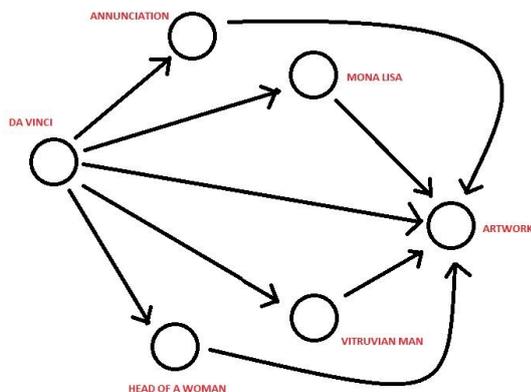
Gambar 4.2.2 Google Knowledge Graph
source: <http://google.com/insidesearch>

Ketika permintaan *search* tentang Da Vinci dilakukan, Google akan mencari simpul yang memiliki info Da Vinci dalam datanya. Kemudian pencarian tidak akan berhenti hanya sampai di situ, Google akan mulai mencari sisi-sisi

yang bersisian dengan simpul Da Vinci serta melakukan pendataan mengenai simpul-simpul yang bersisian dengan sisi tersebut. Misalnya saja, Da Vinci merupakan seorang seniman, sehingga ia pasti akan memiliki sisi(edges) yang terhubung dengan karya seni yang dihasilkannya. Selain itu akan ada juga sisi yang terhubung dengan simpul seniman yang selanjutnya akan mengarahkan pencarian kepada seniman lain.

Sehingga saat ini, jika kita memberikan *query* “Da Vinci” pada *Google Search*, akan ditampilkan informasi mengenai siapa itu Da Vinci, apa saja karya seni yang dihasilkannya, siapa saja orang-orang yang mungkin memiliki hubungan dengannya, kapan ia lahir, di mana ia tinggal, dan lain-lain.

Jika kita melakukan sedikit modifikasi terhadap *query* yang kita berikan, akan ditampilkan hasil yang berbeda sesuai yang kita inginkan. Misalnya jika kita memberikan *query* “Da Vinci Artwork”. Pertama-tama kata kunci tersebut akan membawa kita kepada simpul Da Vinci. Lalu Da Vinci yang pasti memiliki relasi terhadap simpul “Artwork”, di simpul Artwork ini selanjutnya dicari, mana saja karya seni yang merupakan hasil karya dari Da Vinci. Kurang lebih proses yang dilakukan di sini hampir sama seperti yang telah dijelaskan dalam proses *Graph Search* yang dilakukan oleh *Facebook*.



Gambar 4.2.3 Google Knowledge Graph

Penggunaan *Knowledge Graph* ini juga dapat berlaku sebaliknya. Misalkan kita memberikan *query* “Mona Lisa”, maka akan dilihat simpul mana saja yang berhubungan dengan simpul “Mona Lisa”. Dari gambar 4.2.2 sebelumnya, dapat kita lihat bahwa Mona Lisa terhubung dengan simpul Da Vinci dan simpul *Artwork*, sehingga ketika diberikan *query* Mona Lisa, akan diberikan *suggestion* untuk pencarian Da Vinci *Artwork*.

Selain contoh-contoh *query* yang telah diberikan di atas, masih ada lagi hal yang bisa kita eksplorasi dari *Knowledge Graph* ini. Misalnya kita tidak mengetahui siapa presiden dari Indonesia, kita cukup memberikan *query* “Indonesian President”, maka hasil dari pencarian menggunakan graf akan menunjukkan kita pada Presiden Indonesia saat ini yaitu Susilo Bambang Yudhoyono. Tidak hanya itu, kita juga akan melihat orang-orang yang

berhubungan dengannya di hasil pencarian. Orang-orang yang muncul dalam hasil pencarian bisa merupakan orang yang pernah memiliki jabatan yang sama dengan Pak SBY, ataupun kerabat-kerabatnya. Jika kita tengok ke belakang, sekitar tahun 2010 saya pernah mencoba untuk mencari nama presiden suatu Negara, namun hasilnya tidak semudah saat ini. Sebelum digunakannya fitur *searching* yang mengimplementasikan graf, kita harus mencari terlebih dahulu artikel yang memuat tentang presiden tersebut dan kita baru bisa mengetahui namanya jika kita menemukan ada nama yang tertulis dalam artikel tersebut.

Menarik bukan? Dari sebuah algoritma *searching* biasa, dengan mengimplementasikan graf di dalamnya kita dapat mendapat informasi yang tidak terfikirkan oleh kita, namun sangat bermanfaat jika kita mengetahuinya. Bayangkan saat *database* untuk pencarian semacam ini sudah sangat lengkap, kita bisa mengetahui apapun yang berkaitan dengan apapun yang kita inginkan. Namun yang pasti penggunaannya harus disertai dengan hukum yang jelas dan tegas agar privasi setiap orang tetap dapat terjaga.

V. KESIMPULAN

Sebagian besar penggunaan Graf selama ini berhubungan dengan meningkatkan efektifitas penyelesaian masalah dalam suatu pekerjaan, namun ternyata graf juga dapat digunakan untuk memudahkan pengguna *search engine* mencari suatu informasi. Implementasi graf dalam *search engine* dapat menjadikan *search engine* sebagai perangkat lunak yang tidak hanya dapat mencari informasi, namun juga dapat menjadi media untuk mengeksplorasi dan mengenal dunia.

REFERENSI

- [2] Munir, Rinaldi. 2008. "Diktat Kuliah IF 2091 Struktur Diskrit". Bandung:Program Studi Teknik Informatika STEI ITB.
- [3] <http://www.statista.com/topics/1001/google/chart/899/unique-users-of-search-engines-in-december-2012/>, 15 Desember 2013.
- [4] <http://expandedramblings.com/index.php/resource-how-many-people-use-the-top-social-media/>, 15 Desember 2013.
- [5] <http://www.wordstream.com/articles/internet-search-engines-history>, 15 Desember 2013.
- [6] <http://www.wiley.com/legacy/compbooks/sonnenreich/history.htm>, 15 Desember 2013.
- [7] <http://www.google.com/insidesearch/>, 15 Desember 2013.

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 27 November 2013

A handwritten signature in black ink, consisting of a stylized, cursive script that is difficult to decipher but appears to be the name of the author.

Gilang Julian S. (13512045)