

**PEMBANGKIT GRAFIK BERBASIS BAHASA ALAMI PADA
DATABASE MANAGEMENT SYSTEM
(DBMS)**

SKRIPSI

Oleh:

RIZQI LAILATUL MUTHMA'INNAH

NIM. 10650002



**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2014**

HALAMAN PENGAJUAN
PEMBANGKIT GRAFIK BERBASIS BAHASA ALAMI PADA
DATABASE MANAGEMENT SYSTEM
(DBMS)

SKRIPSI

Diajukan Kepada :
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk Menempuh Salah Satu Persyaratan Dalam
Memperoleh Gelar Sarjana Komputer

oleh :
RIZQI LAILATUL MUTHMA'INNAH
NIM. 10650002 / S-1

JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2014

HALAMAN PERSETUJUAN
PEMBANGKIT GRAFIK BERBASIS BAHASA ALAMI PADA
DATABASE MANAGEMENT SYSTEM
(DBMS)

SKRIPSI

Oleh :

Nama : Rizqi Lailatul M
NIM : 10650002
Jurusan : Teknik Informatika
Fakultas : Sains Dan Teknologi

Telah Disetujui, 1 September 2014

Pembimbing I,



Zainal Abidin, M.Kom
NIP. 19760613 200501 1 004

Pembimbing II,



Ririen Kusumawati, M.Kom
NIP. 19720309 200501 2 002

Mengetahui,
Ketua Jurusan Teknik Informatika

Dr. Cahyo Crysdian
NIP. 19740424 200901 1 008

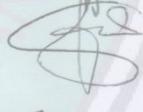
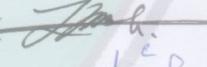
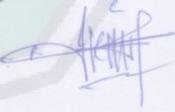
HALAMAN PENGESAHAN

**PEMBANGKIT GRAFIK BERBASIS BAHASA ALAMI PADA
DATABASE MANAGEMENT SYSTEM
(DBMS)**

SKRIPSI

Oleh :
Rizqi Lailatul Muthma'innah
NIM. 10650002

Telah Dipertahankan Di Depan Dewan Penguji Skripsi
Dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal ...9 SEPTEMBER 2014

| Susunan Dewan Penguji: | Tanda Tangan |
|--|---|
| 1. Penguji Utama : <u>Totok Chamidy, M.Kom</u> NIP. 19691222 200604 1 001 | () |
| 2. Ketua Penguji : <u>Dr.M.Amin Hariyadi, M.T</u> NIP. 19670118 200501 1 001 | () |
| 3. Sekretaris : <u>Zainal Abidin, M.Kom</u> NIP. 19760613 200501 1 004 | () |
| 4. Anggota Penguji : <u>Ririen Kusumawati, M.Kom</u> NIP. 19720309 200501 2 002 | () |

Mengetahui,

Ketua Jurusan Teknik Informatika

Dr. Cahyo Crysdian
NIP. 197404242009011008

HALAMAN PERSEMBAHAN

*Dengan segala kerendahan hati, kupersembahkan
sebuah karya kecil yang telah berhasil kuselesaikan ini
kepada :*

Allah SWT, Tuhan Pencipta Alam. Alhamdulillah. Terimakasih atas segala kemudahan dan kelancaran sehingga dapat menyelesaikan Skripsi ini tepat waktu. Shalawat serta salam selalu terlimpahkan teruntuk baginda Rasulullah Muhammad SAW.

Secara khusus skripsi ini kupersembahkan kepada kedua orang tuaku **Masykur Kasmad dan Khusbiyah**, terima kasih telah mendoakan dan memberikan motivasi dan dukungan baik moril maupun materiil. Untuk kakak ipar **Muhammad Rouf** terima kasih telah menjaga kakak tersayangku **Fahriyah Rizqi Pratama**. Adikku tersayang **Faisal Rizqi Hanggara** terima kasih sudah membantu ayah ibu di rumah. Semoga jadi engineer hebat kelak.

Spesial untuk **Cahya Fikri Muhammad**. Penyemangat dan Inspirator dalam penyelesaian skripsi ini. Terima kasih untuk semua waktumu yang tersita. Semoga Allah SWT mempertemukan kita di dunia-dunia selanjutnya. Aamiin...

Teman – temanku **Muiz Lidinillah, Ahmad Wahyu Rosyadi, Achmad An'im Fahmy**, dan yang terhormat **pak aziz** terimakasih telah membantu memberikan pencerahan dalam setiap kesulitan. Serta sahabat – sahabat ku **Rifa, Alif, Umi, Listya, Rizqi Izatul, Elis, Balqis, Listya, Rully** serta sahabat-sahabat yang tidak bisa disebutkan satu persatu, terimakasih telah menemani hari-hari yang melelahkan selama menuntut ilmu, mengerjakan tugas di kampus tercinta ini. Semoga Allah selalu membuka ridhonya yang luas untuk kesuksesan kita gaes.Amin.

Terakhir Semoga ilmu yang penulis terima selama ini barokah dan dapat memberikan manfaat bagi orang-orang disekelilingnya.
Aamiin, Aamiin, Aamiin yaa Robbal 'Aalamiin

HALAMAN MOTTO

لَا يُكَلِّفُ اللَّهُ نَفْسًا إِلَّا وُسْعَهَا لَهَا مَا كَسَبَتْ وَعَلَيْهَا مَا اكْتَسَبَتْ رَبَّنَا لَا تُؤَاخِذْنَا
إِن نَسِينَا أَوْ أَخْطَأْنَا ...

286. Allah tidak membebani seseorang melainkan sesuai dengan kesanggupannya. ia mendapat pahala (dari kebajikan) yang diusahakannya dan ia mendapat siksa (dari kejahatan) yang dikerjakannya.



**HALAMAN PERNYATAAN
ORISINALITAS PENELITIAN**

Saya yang bertandatangan di bawah ini:

Nama : Rizqi Lailatul Mutmainnah
NIM : 10650002
Fakultas/Jurusan : Sains Dan Teknologi / Teknik Informatika
Judul Penelitian : Pembangkit Grafik Berbasis Bahasa Alami pada
Database Management System (DBMS)

Menyatakan dengan sebenarnya bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka. Apabila di kemudian hari terbukti atau dapat dibuktikan skripsi ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut

Malang, 2 September 2014


Saya yang Membuat Pernyataan,

Rizqi Lailatul Mutmainnah
10650002

KATA PENGANTAR



Segala puji bagi Allah SWT yang telah melimpahkan rahmat serta karuniaNya kepada penulis sehingga bisa menyelesaikan skripsi dengan judul “Pembangkit Grafik Berbasis Bahasa Alami pada *Database Management System (DBMS)*” dengan baik.

Shalawat serta salam semoga tercurah kepada Nabi Agung Muhammad SAW yang telah membimbing umatnya dari gelapnya kekufuran menuju cahaya Islam yang terang benderang.

Tanpa menutup mata dan telinga penulisan skripsi ini tidak dapat terselesaikan tanpa dukungan moril, material maupun spiritual dari berbagai pihak. Beribu-ribu terima kasih penulis haturkan kepada:

1. Prof. Dr. H. Mudjia Rahardjo, M. Si selaku Rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Dr. Cahyo Crysdiyan, selaku Ketua Jurusan Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang, yang mendukung dan mengarahkan dalam pengerjaan skripsi ini.
3. Bapak Zainal Abidin, M.Kom selaku Dosen Pembimbing I, dan Ibu Ririen Kusumawati, M.Kom selaku Dosen Pembimbing II, terima kasih atas ilmu, revisi dan masukan yang telah diberikan.
4. Segenap Dosen Teknik Informatika yang telah memberikan bimbingan keilmuan kepada penulis selama masa studi.

5. Teman-teman Teknik Informatika 20104 tahun kita berbagi bangku perkuliahan bersama. Semoga ilmu yang kita peroleh bermanfaat dan membawa barokah.
6. Hanya ucapan terima kasih dan do'a tulus penulis berikan atas apa yang telah mereka berikan semoga apa yang telah mereka lakukan dapat memantulkan kebaikan kembali kepada mereka. *Aamiin yaa robbal 'alamiin.*

Malang, 2 September 2014

Penulis



DAFTAR ISI

| | |
|--|-------|
| HALAMAN JUDUL | i |
| HALAMAN PENGAJUAN | ii |
| HALAMAN PERSETUJUAN | iii |
| HALAMAN PENGESAHAN | iv |
| HALAMAN PERSEMBAHAN | v |
| HALAMAN MOTTO | vi |
| HALAMAN PERNYATAAN | vii |
| HALAMAN PENGANTAR | viii |
| DAFTAR ISI | x |
| DAFTAR GAMBAR | xiii |
| DAFTAR TABEL | xv |
| ABSTRAK | xvi |
| ABSTRACT | xvii |
| مستخلص البحث | xviii |
| BAB I | 1 |
| PENDAHULUAN | 1 |
| 1.1 Latar Belakang | 1 |
| 1.2 Rumusan Masalah | 4 |
| 1.3 Batasan Masalah | 4 |
| 1.4 Tujuan | 5 |
| 1.5 Manfaat | 5 |
| BAB II | 7 |
| LANDASAN TEORI | 7 |
| 2.1 Penelitian Sebelumnya | 7 |
| 2.1.1 Penelitian oleh Enikuomahin A. O. dan Okwufulueze D. O. | 7 |
| 2.1.2 Penelitian oleh Porfírio P. Filipe dan Nuno J. Mamede | 10 |
| 2.2 Kerangka Teori | 15 |
| 2.2.1 Kecerdasan Buatan (<i>Artificial Intelligence</i>) | 15 |

| | | |
|---------------------------------------|--|----|
| 2.2.2 | Bahasa Alami (<i>Natural Language</i>) | 17 |
| 2.2.4 | Basis Data | 19 |
| 2.2.5 | <i>Database Management system</i> | 20 |
| 2.2.6 | SQL (<i>Structured Query Language</i>) | 22 |
| 2.2.7 | <i>Grammar</i> | 23 |
| 2.2.9 | Parsing | 25 |
| BAB III | | 28 |
| ANALISIS DAN PERENCANAAN SISTEM | | 28 |
| 3.1 | Perancangan Sistem | 28 |
| 3.1.1 | Desain Proses | 31 |
| 3.1.2 | Flowchart | 40 |
| 3.1.3 | Percobaan Manual | 44 |
| 3.1.4 | Desain Interface | 49 |
| BAB IV | | 53 |
| HASIL DAN PEMBAHASAN | | 53 |
| 4.1 | Lingkungan Uji Coba | 53 |
| 4.2 | Data Uji Coba | 54 |
| 4.3 | Implementasi Sistem | 55 |
| 4.3.1 | Halaman Awal | 55 |
| 4.3.2 | SQL Conversion | 56 |
| 4.3.3 | SQL Translation | 59 |
| 4.3.4 | SQL Execution | 59 |
| 4.3.5 | Pembangkit grafik | 62 |
| 4.4 | Uji Coba Sistem dan Analisis Sistem | 65 |
| 4.4.1 | Analisis Pengujian pada Tabel Mahasiswa | 65 |
| 4.5 | Integrasi dengan Al-Quran | 78 |
| BAB V | | 80 |

| | |
|----------------------------|----|
| KESIMPULAN DAN SARAN | 80 |
| 5.1 Kesimpulan | 80 |
| 5.2 Saran | 80 |
| DAFTAR PUSTAKA | 82 |
| LAMPIRAN | 84 |



DAFTAR GAMBAR

| | |
|--|----|
| Gambar 2. 1 contoh format kalimat input query 1 | 9 |
| Gambar 2. 2 Hasil vontoh query 1 | 9 |
| Gambar 2. 3 contoh format kalimat input query 3 | 9 |
| Gambar 2. 4 Hasil contoh query 3 | 10 |
| Gambar 2. 5Arsitektur Natural Language untuk Database | 12 |
| Gambar 2. 6 sintak SELECT instruction (SQL language) | 13 |
| Gambar 2. 7 Pohon Sintak (Ewan Klein and Edward Loper, 2009) | 25 |
| Gambar 2. 8 Cara Kerja Top-down Parser | 26 |
| Gambar 2. 9 Cara Kerja <i>Bottom-up Parser</i> | 27 |
| Gambar 3. 1 Blok Diagram proses pembangkit grafik pada DBMS | 29 |
| Gambar 3. 2 Hubungan SQL Converter dengan database server | 29 |
| Gambar 3.3 Kelas kata dan Ontologi pembentuk kalimat inputan pada pembangkit grafik DBMS | 30 |
| Gambar 3. 4 Aturan produksi pembentuk kalimat inputan pada pembangkit grafik DBMS | 33 |
| Gambar 3.5 Pohon Turunan Parsing | 35 |
| Gambar 3. 6 Flowchart Proses penentuan tabel database | 40 |
| Gambar 3.7 Flowchart Proses penerjemah bahasa SQL | 41 |
| Gambar 3. 8 Flowchart Proses NLP | 42 |
| Gambar 3. 9 Flowchart Pembangkit grafik | 43 |
| Gambar 3. 10 Activity Diagram ekstraksi data grafik pada DBMS | 44 |
| Gambar 3. 11 Pembangkit Grafik dari tabel hasil | 49 |
| Gambar 3.12 Rancangan interface aplikasi Pembangkit grafik berbasis natural language | 50 |
| Gambar 3.13 Rancangan sub proses aplikasi Pembangkit grafik berbasis natural language | 51 |

| | |
|--|----|
| Gambar 3. 14 Rancangan form formasi kalimat inputan aplikasi Pembangkit grafik berbasis natural language | 51 |
| Gambar 3.15 Rancangan form pembangkit grafik aplikasi Pembangkit grafik berbasis natural language | 52 |
| Gambar 4. 1 Daftar nama tabel | 54 |
| Gambar 4. 2 daftar nama kolom | 54 |
| Gambar 4. 3 Halaman Awal | 55 |
| Gambar 4. 4 Contoh Inputan User | 56 |
| Gambar 4. 5 Proses parsing berdasarkan aturan produksi | 57 |
| Gambar 4. 6 Proses hasil SQL Execute | 60 |
| Gambar 4. 7 Pesan error ketika terjadi kesalahan pada inputan | 62 |
| Gambar 4. 8 hasil pencarian pada tabel dinamis | 63 |
| Gambar 4. 9 Hasil akhir pembangkit grafik berupa <i>bar chart</i> | 64 |

DAFTAR TABEL

| | |
|--|----|
| Tabel 3. 1 Tabel type query 1 | 36 |
| Tabel 3. 2 Tabel type query 3 | 37 |
| Tabel 3. 3 Tabel type query 2 | 37 |
| Tabel 3. 4 Tabel type query 4 | 38 |
| Tabel 3. 5 Tabel type query 5 | 38 |
| Tabel 3. 6 Tabel type query 6 | 39 |
| Tabel 3. 7 contoh tabel hasil | 44 |
| Tabel 4. 1 Tabel kebutuhan hardware dan kebutuhan software | 53 |
| Tabel 4. 2 Sampel kalimat uji coba yang bernilai benar karena sesuai dengan Format SQL | 65 |

ABSTRAK

Muthmainnah, Rizqi Lailatul. 2014. 10650002. **Pembangkit Grafik Berbasis Bahasa Alami Pada Database Management System (DBMS)** Tugas akhir / skripsi. Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing : (I) Zainal Abidin, M.Kom (II) Ririen Kusumawati, M.Kom

Kata Kunci: Bahasa alami (*Natural Language*), SQL (*Structured Query Language*), *Parsing*, *Database Management System (DBMS)*

Pengguna yang awam terhadap bahasa SQL tidak harus mengetahui format SQL jika sekedar ingin mengakses database untuk mengetahui informasi. Hanya menggunakan kalimat perintah berupa bahasa Indonesia sebagai inputan nya. Dimana data yang dihasilkan akan dieksekusi dan disajikan secara grafik, karena penyajian data dalam bentuk grafik atau gambar akan lebih mudah ditangkap daripada berupa baris atau tabel. Pada penelitian ini sistem dibangun menggunakan *Natural Language Processing* dengan Parser sebagai proses inti nya. Kategori frasa kata dibedakan berdasarkan kategori sintaksis nya. Kata yang mempunyai bentuk serta perilaku yang sama dimasukkan dalam satu kelompok untuk membentuk aturan produksi. Query yang digunakan adalah query *select projection* untuk mengakses jumlah yaitu *Select Count(*)*. Dari hasil penelitian yang dilakukan, struktur inputan yang berhasil dieksekusi oleh sistem sangat bergantung pada kosa kata dan type query yang dibentuk.

ABSTRACT

Muthmainnah, Rizqi Lailatul. 2014. 10650002. **Generating Graph Based On Natural Language Database Management System (DBMS)**. Final project / thesis. Department of Informatics, Faculty of Science and Technology of the State Islamic University of Maulana Malik Ibrahim Malang.

Promotor: (I) Zainal Abidin, M. Kom

(II) Ririen Kusumawati,

A lay user of the SQL language does not have to know the SQL format if just want to access the database to find out information. Just use a command sentence Indonesia language as its input. Where data generated will be executed and presented in graphs, because the presentation of the data in graphs or images will be captured more easily than a row or table (Graph Generator). In this research the system built using Natural Language Processing to the Parser as the core process. Category word phrases are distinguished by syntactical category. Words that have the same behavior as well as put in one group to form the rules of production. The Query used is the query select projection to access the number of IE Select Count (*). From the research results conducted, the input structure successfully executed by the system depends heavily on vocabulary and type the query in the forms.

Key words: Natural language (Natural Language), SQL (Structured Query Language), Parsing, Database Management System (DBMS)

ملخص البحث

رزقي ليلة المطمعنة، 2014، 10650002. مؤلّد رسم البيانة بأساس اللغة الطبيعيّة
 دتابس مينيجمن سيستم (د ب م س). البحث الجامعي. قسم الحاسوب، كلية
 العلمية والتكنولوجيّة، جامعة مولانا مالك إبراهيم الإسلامية الحكومية بمالانج.
 المشرف : الأستاذ زين العابدين الماجستير والأستاذة ريرين كسوماواتي الماجستير

الكلمات الأساسيات : اللغة الطبيعيّة (Natural Language)، س ق ل (Structured Query Language)، فرسينق، دتابس مينيجمن سيستم (د ب م س)
 المستخدم الذي يضع في اللغة س ق ل ما وجب على أن يعرف شكل س ق ل إذا يريد
 دتابس لمعرفة معلومات. فيستخدم الجملة القيادة فقط اللغة الإندونيسيّة كمدخلاتها.
 والبيانة التي تحصلها سيعدم ويظهر بالرسم البيانيّة. لأنّها تظهر البيانة بالرسم البيانيّة
 سأسهل لتعرف من الخط أو الجدول . في هذا البحث، شكل نظام تستخدم Natural
 Language Processing بفرسر كعملية أساسيّة. العبارة تختلف إلى نحويتها. المفردة التي
 تملك الشكل والعمل المتساوي دخلها في الفرقة الواحدة لتشكّل نظام الإنتاج. قوري
 الذي يستخدم له هو قوري select projection ليصل العدد، هو سلك قانون (Select
 Count). من نتيجة هذا البحث، الهيكل المدخليّ الذي نفذت بنجاح بنظام تعتمد جدّا
 إلى المفردة تيفي قوري المشكلة

BAB I

PENDAHULUAN

1.1 Latar Belakang

Organisasi pada lingkup teknologi informasi pasti mempunyai kebutuhan manajerial sistematis basis data yang kompleks. Pihak yang tidak berkompeten atau terkait langsung dengan informasi data tersebut tidak bisa dengan mudah mengakses data, karena sistem basis data hanya di handle oleh pekerja yang ahli dan mengerti bahasa database (Setyawan, 2009).

Berkaitan dengan sumber daya manusia, faktor *fleksibilitas* merupakan alasan bahwa setiap orang diharapkan untuk dapat menguasai hal yang bukan keahliannya (Mulyadi, 2001). Salah satunya adalah hak akses memperoleh informasi ke basis data.

Pengguna basis data dibedakan berdasarkan cara mereka berinteraksi dengan sistem basis data. Salah satunya adalah pengguna canggih yaitu pengguna basis data yang berinteraksi dengan sistem tanpa melalui program. Mereka mengakses data dengan bahasa query kemudian mengirim kalimat perintah yang ditulis dengan SQL ke *query prosesor* yang fungsi utamanya untuk memecah kalimat perintah menjadi instruksi yang dipahami sistem basis data (Nugroho, 2011). Oleh sebab itu untuk mengakses data atau informasi dalam database diperlukan pengetahuan tentang bahasa SQL yang merupakan bahasa yang

digunakan untuk mengakses data dalam basis data relationship dan merupakan bahasa standar yang digunakan dalam manajemen basis data relasional.

Permasalahan utama disini adalah tidak semua pengguna mengerti perintah SQL, hanya terbatas kalangan yang aktif dalam bidang Teknologi Informasi atau yang terlibat langsung dalam pengorganisasian database. Keterbatasan tersebut menurunkan aksesibilitas pengguna yang menggunakan Bahasa Indonesia dalam keperluan meningkatkan proses penerapan pencarian informasi.

Peneliti ingin membangun sistem dimana Database dalam komputer bisa diakses hanya dengan menggunakan bahasa indonesia sebagai inputannya dan membuat program tersebut mampu memberikan respon kepada pengguna dalam bentuk informasi yang disajikan dalam bentuk grafik dan tabel, sehingga pengguna yang awam terhadap bahasa SQL tidak perlu harus mengetahui format SQL jika hanya sekedar untuk mendapatkan informasi dalam suatu database.

Penyajian informasi dalam fokus kali ini, data disajikan secara sistematis sehingga memperjelas penyajian informasi yang diperoleh dari basis data. Penyajian data dalam bentuk gambar atau grafik akan lebih cepat bisa ditangkap daripada data dalam bentuk baris kata ataupun tabel. 90% dari informasi yang terkirim ke otak adalah *visual*. Visual diproses 60.000x lebih cepat di otak dibandingkan dengan teks. Hal ini artinya informasi apapun yang disampaikan lebih baik dikomunikasikan dalam bentuk gambar/grafik.

Penelitian yang membahas tentang aplikasi pengolah bahasa alami telah banyak dilakukan, salah satunya oleh Porfírio P. Filipe dengan judul *Databases and Natural Language Interfaces*. Penelitian tersebut berfokus pada tahap penerjemahan, model konseptual, database relasional yang mengatur semua data sehingga mendukung proses penerjemahan. Sistem tersebut mengalih bahasakan bahasa Portugis, Perancis, Inggris, dan Spanyol sehingga dapat diproses ke database sumber daya pariwisata. Untuk sistem yang peneliti kembangkan saat ini, sistem dapat mengekstraksi data grafik dari basis data akademik dengan cara mengakses database dalam bahasa Indonesia lalu mengeluarkan hasil output berupadadata tabel dan grafik sebagai bentuk informasinya.

Essensi proses konversi bahasa dari bahasa satu ke bahasa yang lain tercermin dalam peristiwa turunnya wahyu Allah kepada nabi Muhammad SAW. Seperti Hadist nabi dibawah ini.

عَنْ عَائِشَةَ أُمِّ الْمُؤْمِنِينَ - رَضِيَ اللَّهُ عَنْهَا - أَنَّ الْأَحَارِثَ بْنَ هِشَامٍ - رَضِيَ اللَّهُ عَنْهُ - سَأَلَ رَسُولَ اللَّهِ - صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ - فَقَالَ يَا رَسُولَ اللَّهِ كَيْفَ يَأْتِيكَ الْوَحْيُ فَقَالَ رَسُولُ اللَّهِ - صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ - « أَحِبَانًا يَأْتِينِي مِثْلَ صَلْصَلَةِ الْجَرَسِ - وَهُوَ أَشَدُّ عَلَيَّ - فَيُفْصِمُ عَنِّي وَقَدْ وَعَيْتُ عَنْهُ مَا قَالَ ، وَأَحِبَانًا يَتَمَثَّلُ لِي الْمَلَكُ رَجُلًا فَيَكَلِّمُنِي فَأَعِي مَا يَقُولُ » . قَالَتْ عَائِشَةُ رَضِيَ اللَّهُ عَنْهَا وَقَدْ رَأَيْتُهُ يَنْزِلُ عَلَيْهِ الْوَحْيُ فِي الْيَوْمِ الشَّدِيدِ الْبَرْدِ ، فَيُفْصِمُ عَنْهُ وَإِنَّ جَبِينَهُ لَيَتَفَصَّدُ عَرَقًا

Dari Aisyah Ummul Mukminin r.a. bahwa Harits bin Hisyam r.a. bertanya kepada Nabi Muhammad SAW, "Ya Rasulullah, bagaimana caranya wahyu turun kepada Anda?" Rasulullah menjawab, "kadang-kadang wahyu itu datang kepadaku seperti bunyi lonceng. Itulah yang sangat berat bagiku. Setelah bunyi itu berhenti, aku baru mengerti apa yang disampaikan. Kadang-kadang malaikat menjelma seperti seorang laki-laki menyampaikan kepadaku dan aku mengerti apa yang disampaikan," Aisyah berkata, "Aku pernah melihat Nabi

ketika turunnya wahyu kepadanya pada suatu hari yang amat dingin. Setelah wahyu itu berhenti turun, kelihatan dahi Nabi bersimpah peluh."

Hadits di atas adalah hadits ke-2 dalam Shahih Bukhari (صحيح البخارى), di bawah Kitab Bad'il Wahyi (كتاب بدء الوحي) (Permulaan Turunnya Wahyu). Meskipun Imam Bukhari tidak memberikan bab pada hadits kedua ini, dari matannya terlihat jelas bahwa ia memuat cara turunnya wahyu kepada Rasulullah SAW. Hal ini berhubungan dengan proses konversi bahasa Allah SWT kepada baginda nabi Muhammad SAW yang berisi wahyu melalui berbagai cara untuk disampaikan kepada umat islam. Sama halnya dengan sistem ini dirancang untuk bisa memahami dan memproses bahasa manusia kedalam bahasa buatan lalu di simpan ke data berupa grafik.

1.2 Rumusan Masalah

Rumusan masalah dalam penelitian ini yaitu bagaimana merancang dan membangun mesin pengolah bahasa alami dalam DBMS yang dapat mengekstraksi data dalam bentuk Grafik dengan cara menerjemahkan bahasa SQL melalui inputan user berupa bahasa Indonesia.

1.3 Batasan Masalah

- a. Aplikasi ini menerjemahkan inputan user yang menggunakan bahasa Indonesia menjadi bahasa SQL.
- b. Aplikasi ini menggunakan bank data (*database*) yang kemudian di koneksikan dengan java netbeans.

- c. Aplikasi ini menggunakan study kasus basis data akademis (data Mahasiswa dan Dosen).
- d. Penggunaan bahasa SQL hanya melibatkan query *select count* untuk mengeluarkan hasil berupa jumlah untuk selanjutnya diproses dalam grafik, serta penggunaan *where* untuk kalimat perintah menggunakan kondisi.
- e. Bahasa alami yang diproses hanya satu baris kalimat perintah bahasa Indonesia yang sesuai dengan *type query* yang telah ditentukan.

1.4 Tujuan

Tujuan dari penelitian ini adalah membangun mesin pengolah bahasa alami dalam DBMS yang dapat mengekstraksi data dalam bentuk Grafik dengan cara menerjemahkan bahasa SQL melalui inputan user berupa bahasa Indonesia. Serta mengaplikasikan komponen *Natural Language Processing* dalam penerapan sistem.

1.5 Manfaat

Manfaat yang dapat diambil dari penelitian tugas akhir ini adalah:

- a. Dengan adanya aplikasi ini dapat membantu orang yang awam untuk penyajian data ke bentuk grafik dalam mengakses informasi database, tanpa harus mempelajari struktur SQL .
- b. Mampu mengaplikasikan komponen *Natural Language Processing* dalam penerapan Sistem Pembangkit Grafik Berbasis Bahasa Alami pada DBMS.

1.6 Sistematika Penulisan

Penulisan skripsi ini tersusun dalam lima bab dengan sistematika penulisan sebagai berikut :

BAB I PENDAHULUAN

Pendahuluan adalah bab yang menjelaskan alasan dan tujuan dari penelitian ini, yang terdiri dari latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian

BAB II TINJAUAN PUSTAKA

Kajian pustaka adalah bab yang menjelaskan penelitian terkait dan teori-teori pendukung untuk melakukan penelitian ini

BAB III ANALISIS DAN PERANCANGAN SISTEM

Analisis dan perancangan sistem bab yang membahas tentang bagaimana desain sistem aplikasi yang akan di buat. Perancangan sistem didalam aplikasi ini terdiri dari desain proses, flowchart, percobaan manual dan desain interface.

BAB IV HASIL DAN PEMBAHASAN

Menjelaskan tentang hasil sistem dan pengujian aplikasi bahasa alami yang telah diterapkan serta integrasi agama dengan sistem.

BAB V KESIMPULAN DAN SARAN

Berisi kesimpulan dan saran

BAB II

LANDASAN TEORI

2.1 Penelitian Sebelumnya

2.1.1 Penelitian oleh Enikuomehin A. O. dan Okwufulueze D. O.

Penelitian tersebut berjudul *An Algorithm for Solving Natural Language Query Execution Problems on Relational Databases*. Penelitian tersebut dibuat untuk menjembatani kesenjangan yang ada antara kebutuhan informasi dan pengguna yang non-ahli interaksi dengan database. Untuk menghemat waktu eksekusi, peneliti menyajikan algoritma *executable* untuk mengambil hasil pengolahan bahasa alami berbasis query dari *database* relasional. Fitur tambahan dari sistem ini adalah penggunaan kamus pengetahuan seperti tabel di mana penyajian natural language oleh user dapat dijadikan asumsi dalam interpretasi pengetahuan.

Sistem melewati beberapa proses, query bahasa alami oleh user diterima sebagai masukan, lalu query bahasa alami di transformasikan ke query SQL formal. SQL query kemudian dikirim ke database relasional. Hasil query yang dihasilkan oleh database diterima dan diubah kembali menjadi ekspresi dalam bahasa alami sesuai permintaan pengguna. Hasil mengubah ini kemudian ditampilkan kepada pengguna sebagai output. Sistem ini dapat diintegrasikan ke modul sistem.

2.1.1.1 Alur Sistem

Langkah-langkah sistem diperlukan untuk operasi yang efisien dari NLIDB. Tujuan dari percobaan pelaksanaan algoritma, database departemen ilmu komputer digunakan sebagai studi kasus. Proses eksekusi dapat digolongkan dalam fase dan disajikan berupa Inputan untuk interface *Natural Language* berupa bahasa inggris dan proses Transformasi Natural Language Query ke Sql. Database yang digunakan dalam hal ini adalah *Relational Database*. Untuk berinteraksi dengan relasional *database*, bahasa yang akan digunakan adalah *Structured Query Language (SQL)*. Karena antarmuka bahasa alami mengumpulkan ekspresi bahasa alami sebagai input, masukan ini harus dikonversi ke ekspresi SQL yang sesuai sebelum Database bisa memahami permintaan dari pengguna. Oleh karena itu, harus dilakukan beberapa transformasi untuk mengubahnya untuk query SQL yang setara. Aplikasi ini harus dapat membagi permintaan bahasa alami ke token penyusunnya, dan melalui perbandingan dengan isi corpus kemudian memilih kata kunci dalam pernyataan itu. Kata kunci ini kemudian dibentuk menjadi perumusan query SQL yang sesuai lalu akan diteruskan ke *Database*.

2.1.1.2 Hasil Penelitian

Sistem ini berbasis web aplikasi. Dengan demikian bahasa yang digunakan termasuk html, css, javascript, php, sql, sedangkan database yang digunakan adalah MySQL seperti yang dinyatakan sebelumnya. Sistem ini menjawab sebagian besar pertanyaan diajukan kepadanya oleh pengguna

dalam bahasa alami. Sistem ini memungkinkan pengguna untuk mendapatkan informasi dengan mengetikkan teks dalam bentuk bahasa alami.



Give me the names of African presidents

Submit

Gambar 2. 1 contoh format kalimat input query 1

The result of your query is presented below :

User Query : Give me the names of African presidents

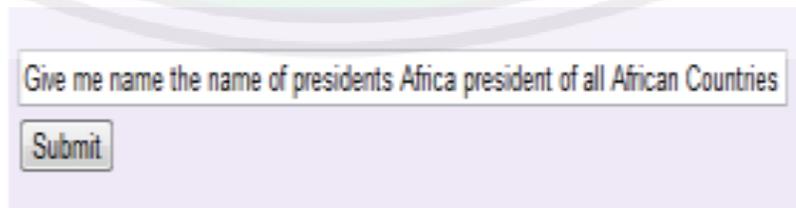
Relevant Words : Africa

PRESIDENT : Goodluck Ebele Jonathan

PRESIDENT : John Atta Mills

Gambar 2. 2 Hasil vontoh query 1

Kasus pertanyaan tata bahasa yang tidak benar dapat ditangani. kesalahan ini mungkin timbul ketika pengguna memiliki tata bahasa yang buruk tentang penggunaan Bahasa Inggris, atau pengguna lupa bahwa kata tertentu telah diketik sekali sebelumnya, dan kemudian diketik lagi.



Give me name the name of presidents Africa president of all African Countries

Submit

Gambar 2. 3 contoh format kalimat input query 3



Gambar 2. 4 Hasil contoh query 3

2.1.2 Penelitian oleh Porfirio P. Filipe dan Nuno J. Mamede

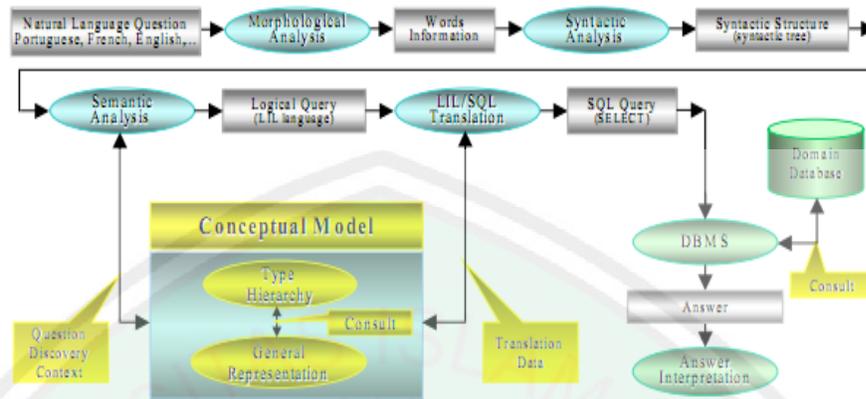
Penelitian tersebut berjudul *Databases and Natural Language Interfaces*. Penelitian Makalah ini berfokus pada tahap penerjemahan. Perhatian khusus ditujukan untuk model konseptual, *database* relasional yang mengatur semua data yang mendukung proses penerjemahan. Pertanyaan pengguna pertama kali diterjemahkan ke dalam bahasa logika dan kemudian menjadi *Structured Query Language* (SQL), yang diproses oleh sistem manajemen *database* untuk kembali jawabannya.

Evolusi teknologi telah menyebabkan perkembangan berkesinambungan dari NLIDBs, terutama di bidang pengolahan bahasa alami. Banyak masyarakat yang membutuhkan untuk mengakses informasi yang disimpan di database. Hampir semua sistem manajemen *database* relasional menggunakan SQL SELECT instruksi sebagai standar antarmuka untuk melakukan eksekusi. Bahasa ini rumit bila digunakan oleh pengguna awam.

Arsitektur sistem didasarkan pada *Intermediate Representation Language* (LIL), di mana pertanyaan diajukan dan berturut-turut berubah (analisis morfologi, sintaksis, dan semantik).dimana pertanyaan dalam bahasa alami berubah menjadi perantara permintaan logis sebelum terjemahan terakhir ke sebuah query SQL. Bahasa ini mengungkapkan arti kalimat dalam hal konsep tingkat tinggi, independen dari struktur database.

Arsitektur sistem terdiri dari dua modul besar. Modul pertama mengontrol pengolahan bahasa alami (komponen linguistik) di mana pertanyaan diajukan dan berturut-turut berubah (analisis morfologi, sintaksis, dan semantik). Satu atau lebih Ekspresi LIL diperoleh pada akhir proses ini. Ekspresi ini sesuai dengan kemungkinan interpretasi dari pertanyaan awal. Mengingat dimensi domain dan fleksibilitas bahasa alami ini, biasanya akan ada beberapa interpretasi untuk pertanyaan yang sama. Komponen kedua adalah yang bertanggung jawab atas hubungan dengan database, menerjemahkan ekspresi LIL ke ekspresi SQL dan mengirim Query ke sistem manajemen *database* untuk menghasilkan jawaban.

Terjemahan tersebut dibuat dengan tahap: bahasa alami menjadi pohon sintaksis, kemudian logis rumus, dan akhirnya menjadi sebuah instruksi SELECT. Komunikasi dengan database sistem manajemen yang menghasilkan jawaban, dilakukan melalui driver ODBC.



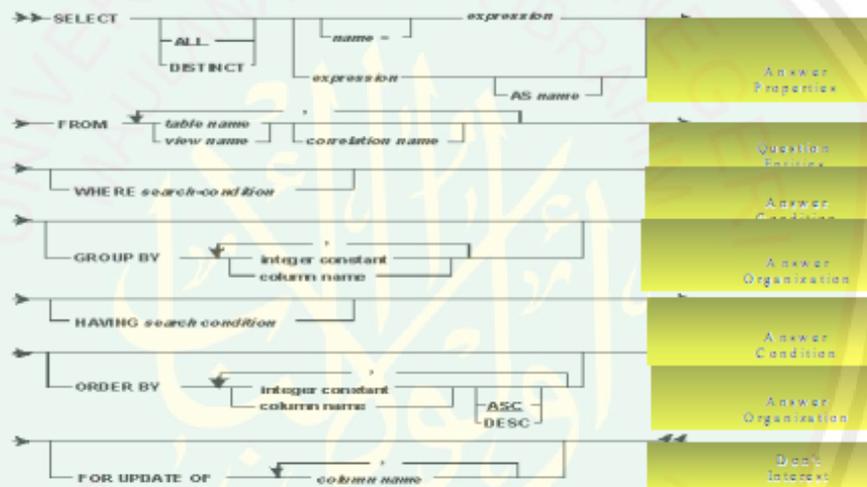
Gambar 2. 5 Arsitektur *Natural Language* untuk *Database*

2.1.2.1. Pengolah Bahasa Alami

Pertanyaan yang ditempatkan untuk sistem NLIDB berisi tiga jenis informasi yang digunakan untuk mengidentifikasi: (a) sifat-sifat yang relevan dengan jawabannya, (b) kondisi seleksi, dan (C) bagaimana jawabannya adalah akan diurutkan. Semua komponen ini adalah opsional, yaitu, mungkin tidak ditentukan. Misalnya, dalam pertanyaan “*which hotels have sauna?*”, Properti yang relevan adalah “*hotel name*” dan kondisi seleksi adalah “*have sauna*”. Perhatikan bahwa pertanyaan awal tidak mengidentifikasi properti yang relevan, yang diperlukan untuk menyimpulkan hal itu selama proses penerjemahan.

Terjemahan dari LIL ke SQL melibatkan identifikasi komponen di atas dan mengubah mereka ke dalam sintaks dari instruksi SELECT. Gambar dibawah menggambarkan sintaks dari instruksi SELECT, yang berkaitan dengan isi dari pertanyaan dalam bahasa alami: (a) nama-nama (kolom) atau

ekspresi (yang melibatkan kolom) yang mengikuti kata kunci SELECT untuk menentukan sifat yang relevan dengan jawabannya, (b) tabel yang mengikutikata kunci FROM menentukan entitas yang dimaksud dalam pertanyaan, (c) ekspresi logis yang diiikuti kata kunci WHERE dan HAVING harus memenuhi kondisi jawabannya (d) mengikuti ORDER BY dan GROUP BY kata kunci adalah definisi output penyortiran. Lihat **gambar 2.6**



Gambar 2. 6 sintak SELECT *instruction (SQL language)*

Komponen WHERE adalah tujuan kami yang paling penting. Mungkin diperlukan untuk menggunakan komponen HAVING sebagai alternatif untuk komponen WHERE saat ekspresi logika panggilan *SQL function*.

Proses penerjemahan menggunakan delapan aturan: satu untuk menggantikan variabel dengan jenisnya; dua sampai menerjemahkan formula

primitif, satu untuk menerjemahkan formula yang memiliki *connectives* logis, satu untuk menangani formula yang telah ditetapkan, dua untuk menjamin bahwa semua tabel yang direferensikan dan kolom milik DARI dan SELECT klausa, dan satu untuk menangani format.

2.1.2.2 Hasil Penelitian

Hasil pengujian penggunaan aplikasi natural language pada contoh inputan pertanyaan :

“What are the names of the hotels having five stars?”

Kami berasumsi bahwa "hotel" entitas diwakili oleh tabel **HOTEL**, yang memiliki kolom **QT_SAUNA** untuk menyimpan jumlah sauna. Fakta *"Hotel memiliki kolam-kolam renang"* diwakili oleh kata kunci 'S' pada kolom **POOL**.

LIL formula: $\text{hotel}(X), V(\text{have}(X,\text{pool}), \text{have}(X,\text{sauna}))$

Inisial SQL query: **SELECT ? FROM ?**

Terjemahan dari: **hotel (X)**:

karena panggilan ke fungsi **CLASS(hotel)** returns 'Table' ekspresi **EQUIVALENT(hotel)** dievaluasi dan nilainya (**hotel**) ditambahkan ke klausa FROM. Kemudian query SQL yang baru adalah: **SELECT ? FROM HOTEL**

Translasi dari $V(\text{have}(X,\text{pool}),\text{have}(X,\text{sauna}))$:

karena rumus LIL juga mengandung **hotel (X)** aturan substitusi menghasilkan $\text{have}(\text{hotel},\text{pool}) \rightarrow \text{TRANSLATION}(\text{HAVE}, \text{HOTEL}, \text{POOL})$

$\rightarrow \text{HOTEL.POOL}='S'$

have(hotel,sauna) → TRANSLATION(HAVE, HOTEL, SAUNA)
→ HOTEL.QT_SAUNA>0

nilai-nilai yang diperoleh ditambahkan ke klausa WHERE menggunakan kata kunci OR sebagai pemisah. Kemudian query SQL yang baru adalah:

```
SELECT ? FROM HOTEL WHERE HOTEL.POOL='S' OR  
HOTEL.QT_SAUNA > 0
```

Penentuan kolom dihilangkan

Ekspresi EQUIVALENT(HOTEL).DEFAULT(HOTEL) dievaluasi dan nilainya (HOTEL.NAME) ditambahkan ke klausa SELECT. Maka query SQL terakhir adalah: SELECT HOTEL.NAME FROM HOTEL WHERE HOTEL.POOL='S' OR HOTEL.QT_SAUNA > 0

Hasil evaluasi FORMATTER(NAME) dalam hal ini CHAR(60) adalah informasi yang memformat jawabannya.

2.2 Kerangka Teori

2.2.1 Kecerdasan Buatan (*Artificial Intelligence*)

Kecerdasan buatan merupakan proses belajar bagaimana membuat komputer dapat melakukan banyak hal, dan pada saat yang bersamaan manusia dapat mengerjakannya juga (Rich dan Knight, 1991). Kecerdasan buatan meliputi bagaimana mempelajari manusia dalam melakukan proses suatu gagasan atau pekerjaan dan bagaimana mempresentasikan proses atau gagasan tersebut kepada mesin seperti komputer. Kecerdasan buatan adalah dasar teori dari bagaimana

manusia melakukan suatu pekerjaan sesuai dengan ide atau gagasannya (Turban, 1992). Winston Mengemukakan tiga tujuan dari kecerdasan buatan, yaitu:

- Membuat mesin yang pintar.
- Memberi pemahaman atas intelegensia itu sendiri.
- Membuat mesin yang bermanfaat bagi manusia.

Menurut Kusumadewi (Kusumadewi, 2003) kecerdasan buatan mempunyai banyak bidang terapan yaitu:

1. *Expert System* (Sistem Pakar)

Komputer digunakan sebagai sarana untuk menyimpan pengetahuan para pakar. Dengan demikian komputer akan memiliki keahlian untuk menyelesaikan permasalahan dengan meniru keahlian yang dimiliki oleh pakar.

2. *Natural Language Processing* (Pengolah Bahasa Alami)

Dengan adanya pengolahan bahasa alami diharapkan user dapat berkomunikasi dengan komputer dengan menggunakan bahasa sehari-hari.

3. *Speech Recognition* (Pengenalan Ucapan)

Melalui pengenalan ucapan diharapkan manusia dapat berkomunikasi dengan komputer dengan menggunakan suara.

4. *Robotics and Sensory System* (Robotika dan Sistem Sensor)

5. *Computer Vision*

Computer Vision mencoba untuk dapat menginterpretasikan gambar atau objek-objek tampak melalui komputer..

2.2.2 Bahasa Alami (*Natural Language*)

Bahasa dapat dibedakan menjadi (1) Bahasa Alami, dan (2) Bahasa Buatan. Bahasa alami adalah bahasa yang biasa digunakan untuk berkomunikasi manusia dengan manusia, misalnya: bahasa Indonesia, Sunda, Jawa, Inggris, Jepang, dan sebagainya. Sedangkan Bahasa buatan adalah bahasa yang dibuat untuk memenuhi kebutuhan tertentu, misalnya bahasa pemodelan atau bahasa pemrograman komputer. “*Natural Language Processing*” atau Pengolahan Bahasa Alami merupakan salah satu bidang ilmu kecerdasan buatan (*Artificial Intelligence*) yang mempelajari komunikasi antara manusia dengan komputer”.

2.2.3 Komponen *Natural Language Processing* (NLP)

Menurut Rich & Knight (Rich & Knight, 1991) ada lima tahap dalam *natural language processing* yaitu :

1. *Morphological Analysis*

Kata - kata secara individu dianalisa berdasarkan komponen dan jenisnya, dan token yang tidak termasuk kata dipisahkan dari kata-kata tersebut.

2. *Syntax Analysis*

Hasil dari *morphologic analysis* yang berupa urutan kata-kata kemudian diproses lebih lanjut dalam tahap *syntax analysis* dengan mentransformasikan urutan kata-kata tersebut menjadi sebuah struktur yang menggambarkan hubungan antar kata-kata tersebut.

3. *Semantic analysis*

Struktur yang diciptakan dari *syntax analysis* akan dianalisa lebih lanjut untuk mendapatkan arti yang sebenarnya dari kalimat tersebut. Bila struktur tersebut tidak memenuhi persyaratan kaidah bahasa alami kalimat tersebut dianggap anomaly semantik.

4. *Discourse Integration*

Pada tahap *semantic analysis* mungkin saja ditemukan ada beberapa kata yang belum mempunyai arti yang jelas karena arti dari kata tersebut bergantung pada arti dari kata satu kalimat sebelumnya. Dalam tahap *discourse integration* ini arti dari kata-kata satu kalimat-kalimat tersebut akan dicari.

5. *Pragmatics Analysis*

Merupakan struktur yang merepresentasikan apa yang dikatakan diinterpretasikan lagi untuk menentukan apa yang dimaksud sebenarnya. Secara praktis, batasan kelima fase diatas seringkali berbaur. Fase-fase tersebut, kadang-kadang dilakukan berurutan, namun terkadang dilewati bahkan dilakukan sekaligus tanpa membedakan fase-fase yang seharusnya dilakukan secara berurutan. Meskipun proses pembagian dalam fase-fase tersebut seringkali berguna, namun fase-fase tersebut dapat juga saling berinteraksi satu sama lain sehingga pembagian tahapan tersebut tidak dimungkinkan lagi

2.2.4 Basis Data

Basis data merupakan kumpulan dari kelompok data (arsip) yang saling berhubungan dimana masing-masing kelompok data tersebut memiliki jenis dan fungsi tertentu. Biasanya suatu basis data diorganisir sedemikian rupa sehingga dapat dimanipulasi dengan cepat dan mudah. Istilah-istilah dalam database :

a) *Entity*

Entity adalah orang, tempat, kejadian atau konsep yang informasinya direkam. Misalnya pada sekolah terdapat *entity* : mahasiswa, matakuliah, dosen, nilai test.

b) *Attribute*

Setiap *entity* mempunyai atribut atau sebutan untuk mewakili suatu *entity*. Seorang mahasiswa dapat dilihat dari atributnya, misalnya : nim, nama, alamat, jenis kelamin. *Attribute* juga disebut sebagai data elemen, data *field*, data item.

c) *Data value* (nilai atau isi data)

Adalah data aktual atau informasi yang disimpan pada tiap data elemen atau atribut. Atribut nama mahasiswa menunjukkan tempat dimana informasi nama mahasiswa disimpan, sedang data *value* adalah Pita, Respati, merupakan isi data nama mahasiswa tersebut.

d) *Record* (tupel)

Yaitu kumpulan elemen-elemen yang saling berkaitan menginformasikan tentang suatu *entity* secara lengkap. Satu record mewakili satu data atau

informasi tentang seseorang misalnya : nomor induk mahasiswa, nama, alamat, jenis kelamin dan seterusnya.

e) *File*

Kumpulan *record-record* sejenis yang mempunyai panjang elemen yang sama, attribute yang sama, namun berbeda-beda data valuenya.

f) *Database*

Kumpulan *file-file* yang mempunyai kaitan antara satu *file* dengan *file* yang lain sehingga membentuk satu bangunan data untuk menginformasikan satu perusahaan, instansi dalam batasan tertentu.

g) *Database Management System (DBMS)*

Kumpulan file yang saling berkaitan bersama dengan program untuk pengelolaannya disebut DBMS. *Database* adalah kumpulan datanya, sedangkan program pengelolanya berdiri sendiri dalam satu paket program yang berfungsi untuk membaca data, mengisi data, menghapus data serta melaporkan data dalam *database*.

2.2.5 *Database Management system*

Database Management System (DBMS) adalah suatu sistem perangkat lunak yang memungkinkan pengguna untuk mendefinisikan, membuat, memelihara, dan mengatur akses ke basis data. Menurut Connolly, *Database Language* terbagi atas dua bagian, yaitu *Data Definition Language* dan *Data Manipulation Language* (Connolly, 2010)

Menurut indrajani (indrajani, 2009), lima komponen penyusun DBMS yaitu sebagai berikut:

- a) Perangkat keras (*Hardware*)
- b) Perangkat lunak (*Software*)
- c) Data

Data merupakan komponen terpenting dalam DBMS karena data merupakan penghubung antara computer dan manusia.

- d) Prosedur

Prosedur adalah instruksi yang menentukan perancangan dan pengaturan basis data dimana pengguna system dan basis data dan memerlukan dokumentasi ini untuk menggunakan system.

- e) *User*

Peranan *user* dapat dibedakan berdasarkan fungsinya. Yaitu sebagai berikut:

1. *Data dan Database Administrator*, yaitu sekelompok user yang bertanggung jawab pada manajemen dan pengendalian basis data.
2. *Database Designer*, dibedakan menjadi perancang basis data secara logika dan fisik.
3. *Application developer* atau *programmer*, berfungsi untuk mengembangkan program program aplikasi yang diperlukan dalam manajemen basis data.

4. *End user*, di antaranya adalah pemilik *system (enterprise)*, para manajer, *superfisor*, operator, pelanggan.

2.2.6 SQL (*Structured Query Language*)

SQL adalah sebagai bahasa standar yang digunakan untuk mengolah database. MySQL sebagai *software* dalam pengolah *database*, menggunakan SQL sebagai bahasanya.

Bunafit Nugroho dan Indah Indriyana (2007), perintah-perintah SQL dikelompokkan menjadi 5 macam yaitu:

1. DDL (*Data Definition Language*).

Adalah perintah SQL yang digunakan untuk mendefinisikan kerangka basis data, perintah-perintahnya adalah:

- a. *Create* : untuk membuat atau menciptakan objek basis data
- b. *Alter* : untuk memodifikasi atau mengubah objek basis data
- c. *Drop* : untuk menghapus objek basisdata
- d. Objek *database* yang dimaksud adalah basis data, tabel, *index*

2. DML (*Data Manipulation Language*)

Adalah perintah yang digunakan untuk mengoperasikan atau memanipulasi isi basis data, SQL menyediakan 4 perintah DML yaitu:

- a. *Select* : digunakan untuk mengambil data dari basis data
- b. *Delete* : digunakan untuk menghapus data pada basis data
- c. *Insert* : digunakan untuk menambahkan data kedalam tabel

d. *Update* : digunakan untuk memodifikasi data pada basis data

3. *Security*

Adalah perintah-perintah yang digunakan untuk menjamin keamanan data, perintahnya antara lain:

- a. *Grant* : digunakan untuk memberikan akses kepada *user* tertentu ke basis data
- b. *Revoke* : digunakan untuk mencabut hak akses dari *user*

4. *Integrity*

Adalah perintah-perintah yang digunakan untuk menjaga kesatuan data, seperti: *recover table* digunakan untuk memperbaiki tabel pada basis data.

5. *Auxilliary*

Adalah perintah-perintah pelengkap atau tambahan, seperti: *unload* dan *rename*.

2.2.7 *Grammar*

Grammar (Tata Bahasa) Menurut *Rich & Knight Grammar* adalah representasi yang menjelaskan fakta sintaksis mengenai suatu bahasa. Representasi paling umum dari *grammar* adalah sekumpulan aturan-aturan produksi. Meskipun hal - hal rinci mengenai bentuk - bentuk yang diperbolehkan dalam aturan tersebut sangat bervariasi (Rich & Knight, 1991). Ada empat komponen dalam *Context Free Grammar*.

1. *Terminal* (T) Merupakan simbol dasar dari suatu string yang terbentuk. Umumnya berupa token.

2. *Non Terminal Variabel* statistik yang merupakan kumpulan dari *string*.

3. *Start Symbol Nonterminal* yang berfungsi sebagai simbol awal.

4. *Production Rule*

Merupakan *grammar* yang menentukan aturan apakah terminal dan non terminal dapat digabungkan untuk membentuk suatu *string*. Contoh *grammar* untuk bahasa Indonesia :

S → Start Symbol

SB → Subyek (Non Terminal)

PR → Predikat (Non Terminal)

OB → Obyek (Non Terminal)

kb → Kata benda (Terminal)

kk → Kata kerja Production Rule : (Terminal)

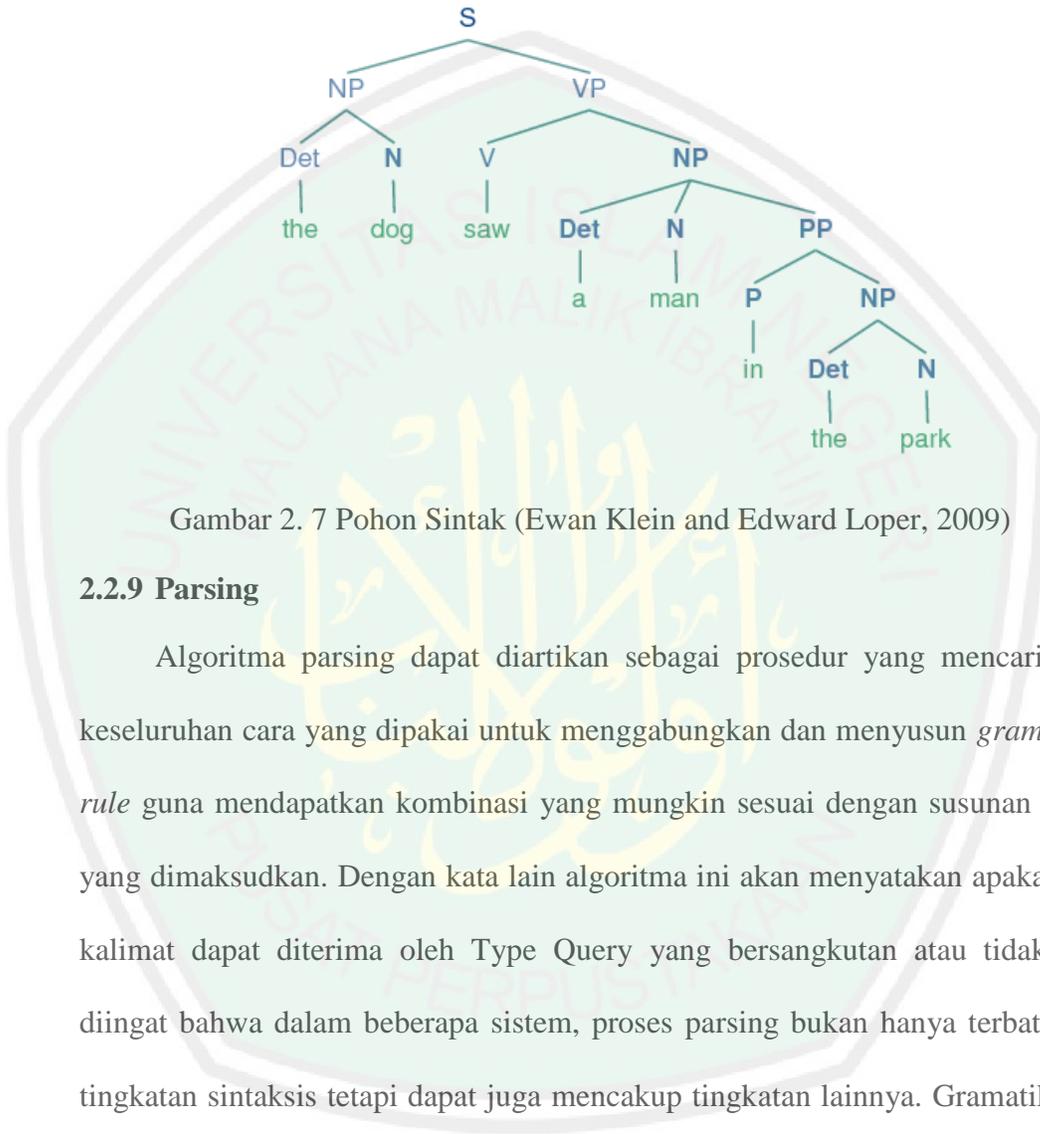
s =>SBPROB SB=>kb PR=>kk OB=>kb

Type Query ialah aturan-aturan pembentukan suatu kalimat dalam sebuah bahasa, atau yang biasa disebut tata bahasa. Dengan adanya *Type Query*, parsing dapat dilakukan secara cepat dengan hanya melakukan proses searching. *Parser* akan mencari aturan-aturan yang tepat untuk membentuk struktur suatu kalimat.

2.2.8 Pohon Sintak

Pohon sintak/pohon penurunan (*syntax tree/derivation tree/ parse tree*) berguna untuk menggambarkan bagaimana memperoleh suatu string dengan cara menurunkan simbol-simbol variabel menjadi simbol-simbol terminal. Setiap simbol variabel akan diturunkan menjadi terminal, sampai tidak ada yang belum

terganti. **Gambar 2.7** memberikan contoh sebuah *tree* yang menguraikan kalimat dalam bahasa Inggris.



Gambar 2. 7 Pohon Sintak (Ewan Klein and Edward Loper, 2009)

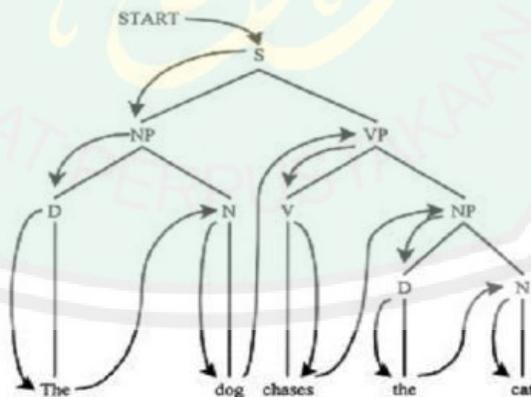
2.2.9 Parsing

Algoritma parsing dapat diartikan sebagai prosedur yang mencari secara keseluruhan cara yang dipakai untuk menggabungkan dan menyusun *grammatical rule* guna mendapatkan kombinasi yang mungkin sesuai dengan susunan kalimat yang dimaksudkan. Dengan kata lain algoritma ini akan menyatakan apakah suatu kalimat dapat diterima oleh Type Query yang bersangkutan atau tidak. Perlu diingat bahwa dalam beberapa sistem, proses parsing bukan hanya terbatas pada tingkatan sintaksis tetapi dapat juga mencakup tingkatan lainnya. Gramatika yang dipakai juga sangat menentukan proses parsing yang digunakan. Hal lain yang juga berkaitan erat dengan proses parsing adalah kamus atau *lexicon* yang digunakan. Dalam *lexicon* disimpan daftar kata yang dapat dikenali sebagai simbol terminal dalam Tipe Query dan informasi yang diperlukan untuk tiap kata tersebut

untuk proses parsing yang bersangkutan. Menurut James Suciadi jenis parser dibagi menjadi 2 antara lain:

a. *Top-down Parsing*

Top-down parsing bekerja dengan cara menguraikan sebuah kalimat mulai dari *constituent* yang terbesar yaitu sampai menjadi *constituent* yang terkecil. Hal ini dilakukan terus-menerus sampai semua komponen yang dihasilkan ialah *constituent* terkecil dalam kalimat, yaitu kata. Sebagai contoh, yang dapat dilakukan proses *top-down parsing* untuk kalimat “*The dog chased the cat*” yang ditunjukkan pada gambar dibawah. dari gambar ini terlihat bahwa *top-down parser* menelusuri setiap *node* pada *parse tree* secara *pre-order*. Beberapa metode parsing yang bekerja secara *top-down* ialah: *Top-down Parser* biasa, *Recursive-descent parser*, *Transition-network parser*, dan *Chart parser*.

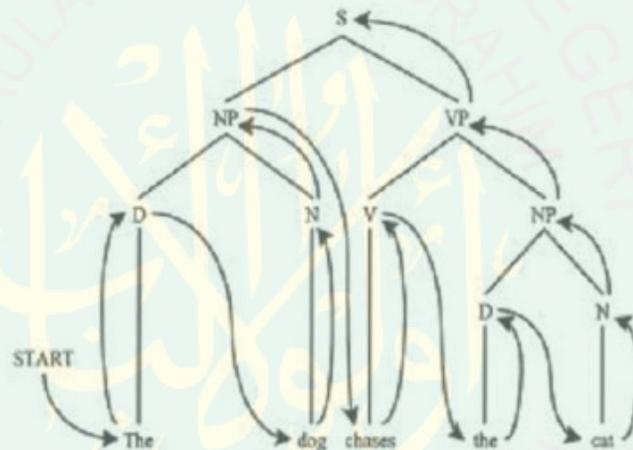


Gambar 2. 8 Cara Kerja *Top-down Parser*

[sumber : Suciadi, James. Studi Analisis Metode-Metode Parsing dan Interpretasi Semantik Pada Natural Language Processing. Journal of Artificial Intelligence Research]

b. Bottom-up Parsing

Bottom-up parser bekerja dengan cara mengambil satu demi satu kata dari kalimat yang diberikan, untuk dirangkaikan menjadi *constituent* yang lebih besar. Hal ini dilakukan secara terus-menerus sampai constituent yang terbentuk ialah *sentence* atau kalimat. Dengan demikian metode *bottom-up* bekerja dengan cara yang terbalik dari *top-down*. Cara kerja *bottom-up parser* ditunjukkan pada gambar dibawah.



Gambar 2. 9 Cara Kerja *Bottom-up Parser*

[sumber : Suciadi, James. Studi Analisis Metode-Metode Parsing dan Interpretasi Semantik Pada Natural Language Processing. Journal of Artificial Intelligence Research]

BAB III

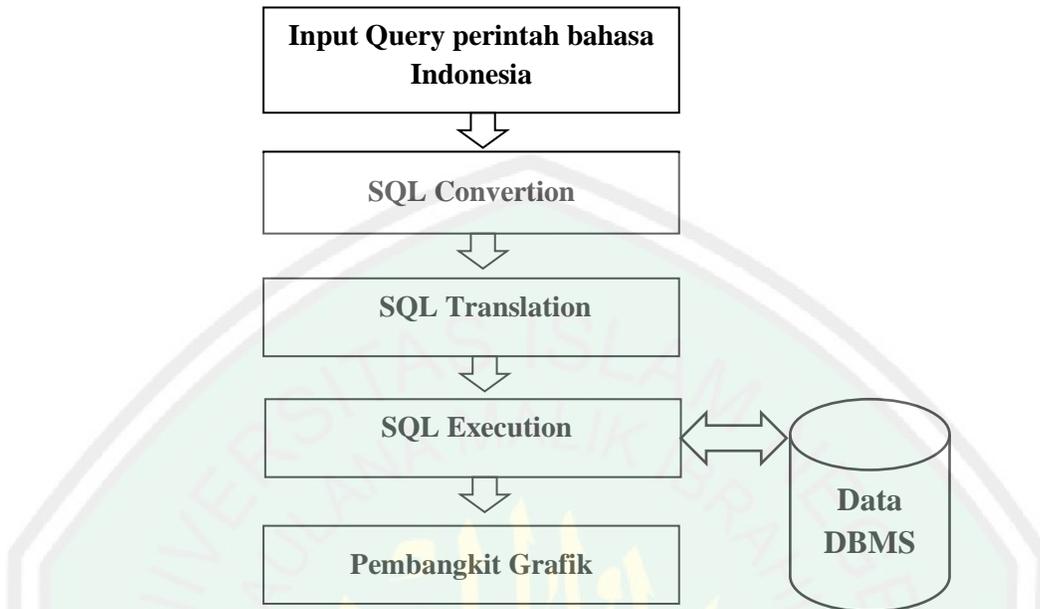
ANALISIS DAN PERANCANGAN SISTEM

3.1 Perancangan Sistem

Aplikasi yang akan dibangun adalah aplikasi pengolah format SQL berbasis bahasa alami kemudian diterapkan pada data grafik.

Proses dari sistem ini dimulai dari inputan user berupa text bahasa alami yakni perintah bahasa indonesia. Inputan tersebut diuraikan menjadi token-token yang kemudian dilakukan pengecekan dengan kata-kata penting yang terdapat didalam array dan memfilter kata kata yang *irrelevant*. Selanjutnya dicari rule/ Type Query bahasa indonesia yang sesuai dengan inputan kalimat (parsing). Parser melakukan pelacakan kalimat inputan untuk mendapatkan langkah pembentuknya. Selanjutnya query dieksekusi ke format SQL kemudian di-*generate* ke grafik berupa *bar chart*.

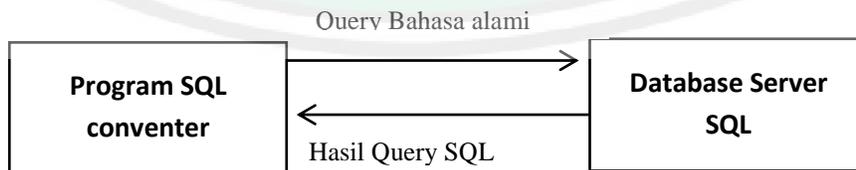
Output yang diharapkan dari aplikasi ini adalah data hasil eksekusi dari DBMS yang disajikan berupa data grafik yang di-*generate* melalui tabel dinamis sesuai dengan inputan bahasa alami dari user. Blok diagram proses dapat dilihat pada **gambar 3.1**.



Gambar 3. 1 Blok Diagram proses pembangkit grafik pada DBMS

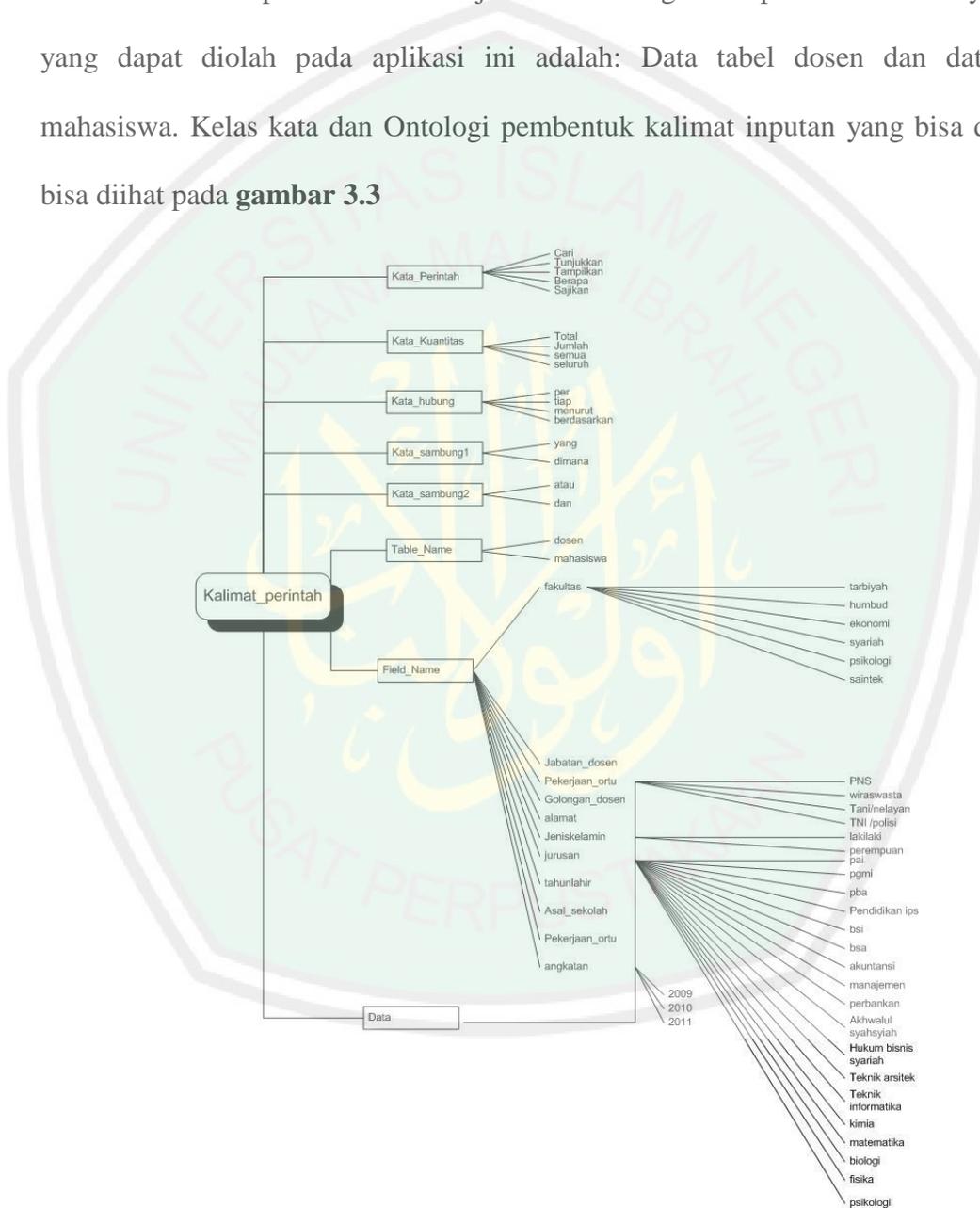
Query yang digunakan dalam penelitian ini yaitu *select projection*. *Select* termasuk bahasa untuk memanipulasi data/ DML (*Data Manipulation Language*). (Nugroho, 2002:32).

Aplikasi penerjemah ini bisa disebut juga dengan SQL Converter. Sistem menerjemahkan query kemudian Database server SQL memberikan hasil query sesuai perintah inputan. Selanjutnya SQL converter menyajikan hasilnya berupa grafik. lihat gambar 3.2



Gambar 3. 2 Hubungan SQL Converter dengan database server

Kategori frasa kata yang diinputkan dapat dibedakan berdasarkan kategori sintaksisnya. Kata yang mempunyai bentuk serta perilaku yang sama dimasukkan dalam satu kelompok berdasarkan jabatan dan kegunaan proses sintaksisnya. Data yang dapat diolah pada aplikasi ini adalah: Data tabel dosen dan data tabel mahasiswa. Kelas kata dan Ontologi pembentuk kalimat inputan yang bisa diproses bisa dilihat pada **gambar 3.3**



Gambar 3.3 Kelas kata dan Ontologi pembentuk kalimat inputan pada pembangkit grafik DBMS

3.1.1 Desain Proses

3.1.1.1 Membaca kalimat inputan

Inputan program aplikasi ini berupa kalimat perintah dalam bahasa Indonesia yang diketikkan kesuatu interface (form). Setiap kata yang diinputkan harus diberi tanda spasi (“ “) karena sistem memproses tiap kata berdasarkan spasi. Bila ada Field Name yang kiranya mempunyai lebih dari satu kata maka harap digabung (tidak menggunakan spasi)

3.1.1.2 SQL Conversion

Ada beberapa tahapan yang diterapkan dalam proses SQL Conversion. Tahapan dalam SQL Conversion antara lain:

a) Scanning

Tahapan awal yang dilakukan adalah proses scanning. Langkah - langkah dari proses scanning adalah:

- a. *Tokenizing*. Yaitu Memecah kalimat dalam bentuk list kata-kata tanpa memperhatikan keterkaitan antar kata yang dalam pemrograman java dilakukan dengan memanggil method split yang dapat diakses oleh tipe data string
- b. *Filtering*. Yaitu Membuang karakter yang *irrelevant*. Dalam proses ini dilakukan dengan cara wordlist (menyimpan kata penting). Pembuangan kata- kata yang irrelevant dilakukan dengan mengecek token dengan daftar kumpulan kata *stop word* yang terdapat dalam database. Jika kata tersebut sama dengan kata dalam *stop word* maka akan dihapus dari daftar.

b) Parsing

Langkah selanjutnya adalah *parsing*. Metode parsing yang digunakan dalam sistem ini adalah metode top-down parsing karena metode parser ini beorientasi pada output.

Parser melakukan pelacakan kalimat inputan untuk mendapatkan langkah pembentuknya. Token akan dikelompokkan menjadi struktur type query sesuai dengan aturan produksi yang telah disusun.

Aturan produksi dibuat untuk menentukan tata bahasa dan menjadi dasar bagi pembuatan aplikasi pemrosesan bahasa alami sebagai bahasa pengganti Query. Penyusunan aturan produksi dilakukan untuk membentuk type query yang mungkin.

Aturan produksi dapat ditentukan berdasarkan pola keteraturan pertanyaan untuk mengakses basis data. Pola keteraturan tersebut dapat diketahui dengan membuat daftar pertanyaan yang mungkin.

Contoh pertanyaan yang mungkin:

1. tampilkan jumlah dosen fakultas saintek
2. tampilkan jumlah mahasiswa tiap fakultas
3. tampilkan jumlah mahasiswa tiap fakultas yang angkatan 2010 dan alamat malang

Pada dasarnya kalimat perintah terdiri dari 3 bagian. Pertama adalah kata tanya, bagian kedua adalah bagian yang ditanyakan (frase atribut), dan bagian ketiga adalah bagian yang diketahui (frase kondisi).

Aturan produksi selengkapnya yang disesuaikan dengan aturan produksi andayani (Andayani, 2002) dapat dirumuskan pada **gambar 3.4** dibawah ini:

| | | |
|-------------------|---|---------|
| <query_pembentuk> | → <kata_perintah> <frase_atribut> <frase_kondisi> | .. (1) |
| <frase_atribut> | → <kata_kuantitas><TName>< frase_atribut >** | .. (2) |
| | <FName>< frase_atribut >** | .. (3) |
| | <kata_hubung><FName>< frase_atribut >** | ..(4) |
| | <kosong> | .. (5) |
| <frase_kondisi> | → <kata_sambung1><FName><data><frase_kondisi>** | .. (6) |
| | <kata_sambung2><FName><data>< frase_kondisi >** | .. (7) |
| | <data>** | .. (8) |
| | <kosong> | .. (9) |
| <kata_perintah> | → tampilkan berapakah tunjukkan berapa | .. (10) |
| <kata_kuantitas> | → jumlah total seluruh semua | .. (10) |
| <kata_hubung> | → tiap per menurut berdasarkan | .. (11) |
| <kata_sambung1> | → yang dimana | .. (12) |
| <kata_sambung2> | → dan atau | .. (13) |
| <TName> | → dosen mahasiswa | .. (14) |
| <FName> | → jabatan_dosen Golongan_dosen tahunlahir angkatan asal_sekolah pekerjaan_ortu jurusan fakultas Alamat angkatan | .. (15) |
| <kosong> | → | .. (16) |
| <data> | → tergantung dari data yang ada pada query. | .. (17) |

Gambar 3. 4 Aturan produksi pembentuk kalimat inputan pada pembangkit grafik DBMS

Notasi yang digunakan dalam aturan produksi tersebut berdasarkan pada aturan BNF (Backus Naur Form) :

- didefinisikan sebagai
- <> simbol nonterminal
- | atau
- ** simbol di depannya dapat diulang nol atau satu kali

Penerapan dari aturan produksi dapat disimulasikan seperti dibawah ini.

Misal Query : *tampilkan jumlah mahasiswa yang alamat malang;*

dapat diperoleh dari aturan produksi tersebut dengan langkah-langkah:

<query_pembentuk> →

<kata_perintah> <frase_atribut> <frase_kondisi>

[<tampilkan><kata_jumlah><TName><frase_atribut><frase_kondisi>]

[<tampilkan><jumlah><mahasiswa><kosong><kata_sambung1><FName>

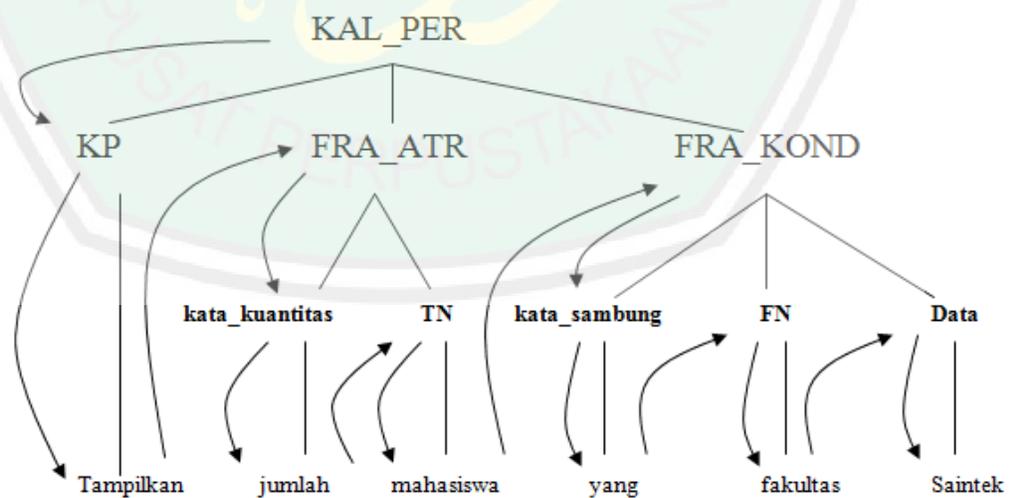
<data> <frase_kondisi>]

[<tampilkan><jumlah><mahasiswa><yang><alamat><malang><kosong>]

[<tampilkan><jumlah><mahasiswa><yang><alamat><malang>]

Dari contoh parsing yang dijelaskan, terbentuklah pohon turunan parsing dari bentuk *constituen global* yaitu kalimat perintah menjadi *constituen terkecil* yaitu kata-perkata. *Constituen global* disini disimbolkan dalam bentuk <KAL_PER> atau kalimat perintah. Dalam <KAL_PER> memiliki turunan *query pembentuk* yang terdiri dari <KP>, <FRA_ATR>, dan <FRA_KOND>.

Di mana <KP> merupakan kata perintah, kemudian <FRA_ATR> merupakan Frase Atribut, dan <FRA_KOND> merupakan Frase Kondisi. Pada **gambar 3.5** dijelaskan tentang proses parsing yang dilakukan secara *top-down parser* di mana proses awal adalah <KAL_PER> kemudian menuju turunan pertama yaitu KP kemudian dari KP turunan selanjutnya adalah kata “tampilkan”. Dari kata “tampilkan” proses selanjutnya adalah pada FRA_ATR di mana turunan dari FRA_ATR adalah kata_kuantitas dan terjadi proses penurunan lagi sehingga mendapatkan kata “jumlah”. Masih dalam FRA_ATR kemudian memproses pada Tname (nama tabel dari database). Turunan dari Tname terdapat kata “mahasiswa”. Dan selanjutnya hampir sama sehingga nanti di proses *SQL translation* akan mendapat suatu rule Type Query yang berfungsi untuk mengklarifikasi apakah kalimat yang input-kan user dapat diterima dan diproses oleh sistem.



Gambar 3.5 Pohon Turunan Parsing

3.1.1.3 SQL Translation

Tahap Translator ini berfungsi untuk mengubah pohon sintak hasil parsing ke formasi query yang sesuai. Memperhatikan kalimat perintah yang biasa digunakan untuk menanyakan jumlah atau frekuensi suatu data darbasis data akademik, maka formasi query dalam translator dirancang untuk mengubah pohon sintak yang susunan frase atribut dan kondisinya sesuai dengan pola pertanyaan yang menerangkan tentang banyaknya jumlah (frekuensi data).

Adapun formasi kalimat inputan berupa kalimat perintah yang bisa diproses oleh sistem sesuai dengan type query adalah sebagai berikut:

1. Kalimat perintah mengakses jumlah tanpa kata kata_hubung, kata_sambung / kata_pelengkap.

Dengan struktur type query:

TYPEQUERY1 → { **KP**, **kata_kuantitas**, **TN**, **FN**, **Data** }

Contoh inputan:

Tabel 3. 1 Tabel type query 1

| KP | kata_kuantitas | TN | FN | Data |
|-----------|----------------|-----------|----------|---------|
| Tampilkan | jumlah | Mahasiswa | Fakultas | saintek |
| Tampilkan | jumlah | Mahasiswa | Angkatan | 2010 |
| Tampilkan | jumlah | Mahasiswa | Alamat | malang |

2. Kalimat perintah mengakses jumlah dengan kata_hubung

Dengan struktur type query:

TYPEQUERY2→ { **KP, kata_kuantitas, TN, Kata_Hubung, FN** }

Contoh inputan:

Tabel 3. 2 Tabel type query 2

| KP | kata_kuantitas | TN | kata_hubung | FN |
|-----------|----------------|-----------|-------------|----------|
| Tampilkan | jumlah | Mahasiswa | Tiap | fakultas |
| Tampilkan | jumlah | Mahasiswa | per | angkatan |

3. Kalimat perintah mengakses jumlah dengan menggunakan kata_sambung1.

Dengan struktur type query:

TYPEQUERY3→ { **KP, kata_kuantitas, N, Kata_Sambung1, FN, Data** }

Contoh inputan:

Tabel 3. 3 Tabel type query 3

| KP | kata_kuantitas | TN |
|---------------|----------------|-----------|
| Tampilkan | jumlah | Mahasiswa |
| Tampilkan | jumlah | Mahasiswa |
| Tampilkan | jumlah | Mahasiswa |
| kata_sambung1 | FN | Data |
| Yang | fakultas | saintek |
| Yang | angkatan | 2010 |
| Yang | alamat | malang |

4. Kalimat perintah mengakses jumlah dengan kata_hubung dan satu

kata_sambung

Dengan struktur type query:

TYPEQUERY4→{ **KP, kata_kuantitas, TN, Kata_Hubung, FN, Data, Kata_Sambung1, FN, Data** }

Contoh inputan:

Tabel 3. 4 Tabel type query 4

| KP | kata_kuantitas | TN | kata_hubung | FN |
|---------------|----------------|-----------|-------------|----------|
| Tampilkan | jumlah | mahasiswa | Tiap | Fakultas |
| kata_sambung1 | FN | Data | | |
| yang | angkatan | 2010 | | |

5. Kalimat perintah mengakses jumlah dengan kata_hubung *dan* dua kata_sambung

Dengan struktur type query:

TYPEQUERY5→{ **KP, kata_kuantitas, TN, Kata_Hubung, FN, Kata_Sambung1, FN, Data, Kata_Sambung2, FN, Data** }

Contoh inputan:

Tabel 3. 5 Tabel type query 5

| KP | kata_kuantitas | TN | kata_hubung | FN |
|---------------|----------------|-----------|-------------|----------|
| Tampilkan | jumlah | mahasiswa | Tiap | Fakultas |
| kata_sambung1 | FN | Data | | |
| yang | angkatan | 2010 | | |
| kata_sambung2 | FN | Data | | |
| dan | alamat | Malang | | |

6. Kalimat perintah mengakses jumlah dengan dengan kata_hubung *dan* tiga kata_sambung

Dengan struktur type query:

TYPEQUERY6→{ **KP, kata_kuantitas, TN, Kata_Hubung, FN,**
Kata_Sambung1, FN, Data, Kata_Sambung2, FN, Data,
Kata_Sambung2, FN, Data }

Contoh inputan:

Tabel 3. 6 Tabel type query 6

| KP | kata_kuantitas | TN | kata_hubung | FN |
|---------------|----------------|-----------|-------------|----------|
| Tampilkan | jumlah | mahasiswa | Tiap | Fakultas |
| kata_sambung1 | FN | Data | | |
| yang | angkatan | 2010 | | |
| kata_sambung2 | FN | Data | | |
| dan | alamat | Malang | | |
| kata_sambung2 | FN | Data | | |
| dan | Tahun_lahir | 1992 | | |

3.1.1.4 SQL Execution

Tahap ini merupakan tahap eksekusi sistem terhadap inputan user, sehingga setiap inputan yang diterima akan diterjemahkan ke format SQL (*SQL language*) yang sesuai dengan tipe formasi query yang telah ditentukan sebelumnya.

3.1.1.5 Pembangkit grafik

Pembuatan grafik frekuensi merupakan kelanjutan dari pembuatan tabel distribusi frekuensi karena pembuatan grafik itu haruslah didasarkan pada tabel distribusi frekuensi. Gambar grafik frekuensi yang digunakan adalah grafik bar.

Grafik terbentuk dari SQL yang telah tereksekusi pada proses eksekusi SQL. Sumbu X dan sumbu Y dari grafik otomatis mengikuti tabel yang telah terbentuk sebelumnya di sistem.

3.1.2 Flowchart

3.1.2.1 Proses Penentuan Tabel Database

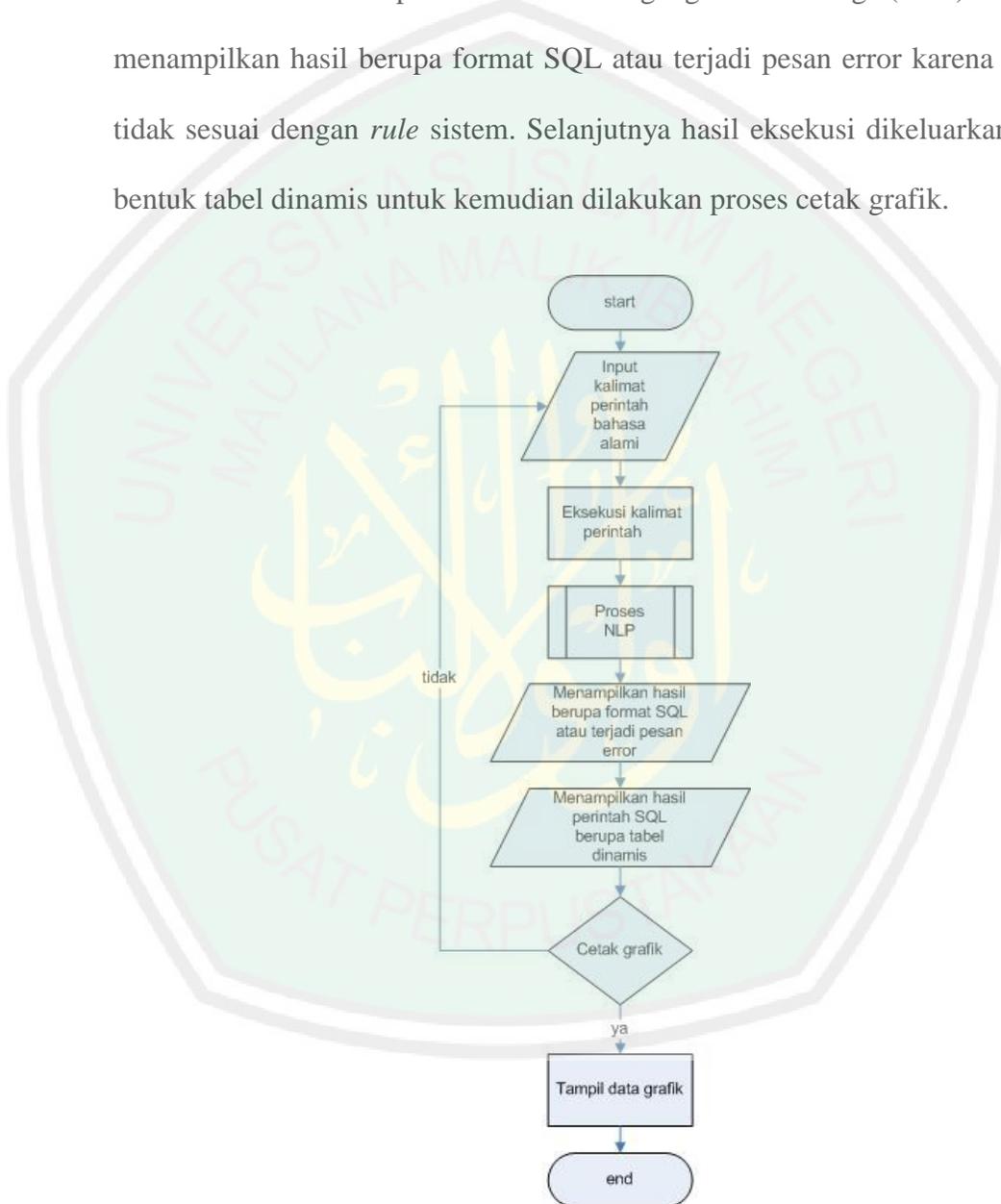
Pada **gambar 3.6** sistem melakukan pencarian bank data dengan mengecek koneksi database kemudian melakukan pencarian table database diteruskan dengan mencari field database.



Gambar 3. 6 Flowchart Proses penentuan tabel database

3.1.2.2 Proses penerjemah bahasa SQL

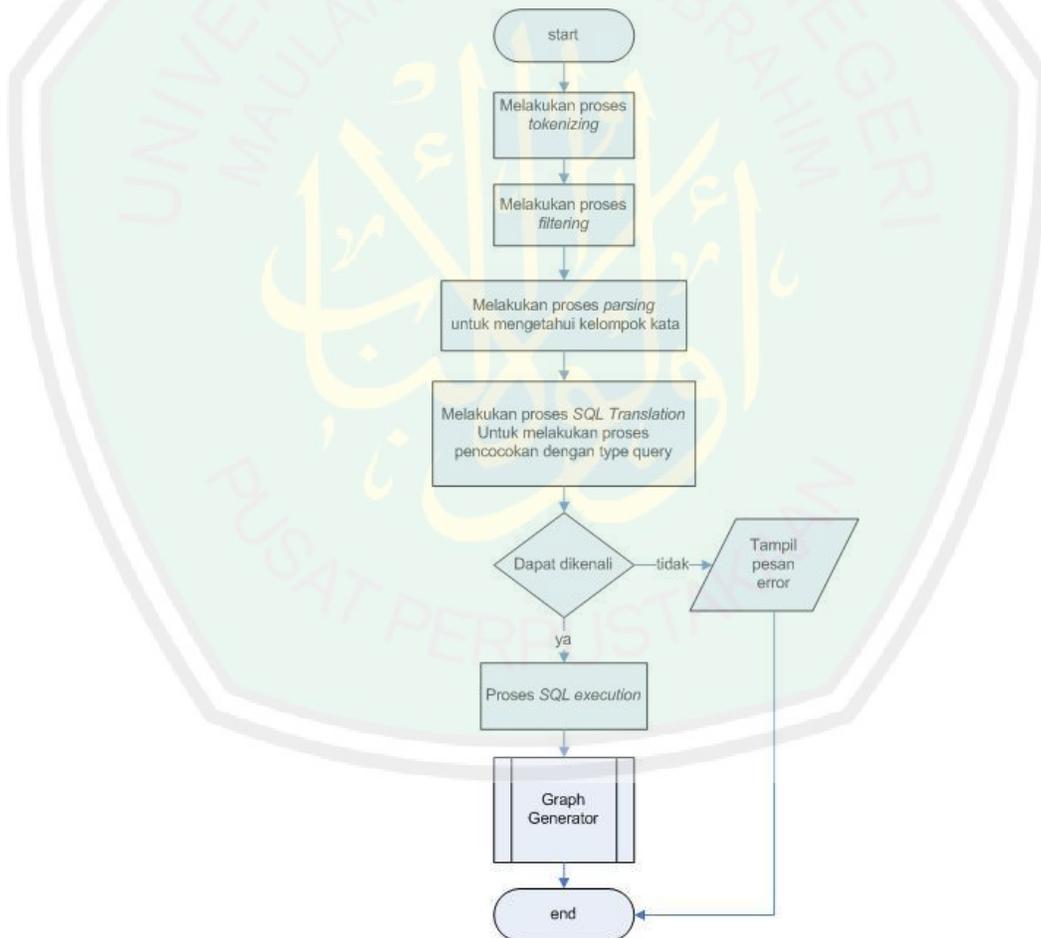
Pada **gambar 3.7** sistem menerima inputan berupa bahasa alami kemudian dilakukan proses Natural Language Processing (NLP) sehingga menampilkan hasil berupa format SQL atau terjadi pesan error karena inputan tidak sesuai dengan *rule* sistem. Selanjutnya hasil eksekusi dikeluarkan dalam bentuk tabel dinamis untuk kemudian dilakukan proses cetak grafik.



Gambar 3.7 Flowchart Proses penerjemah bahasa SQL

3.1.2.3 Proses NLP

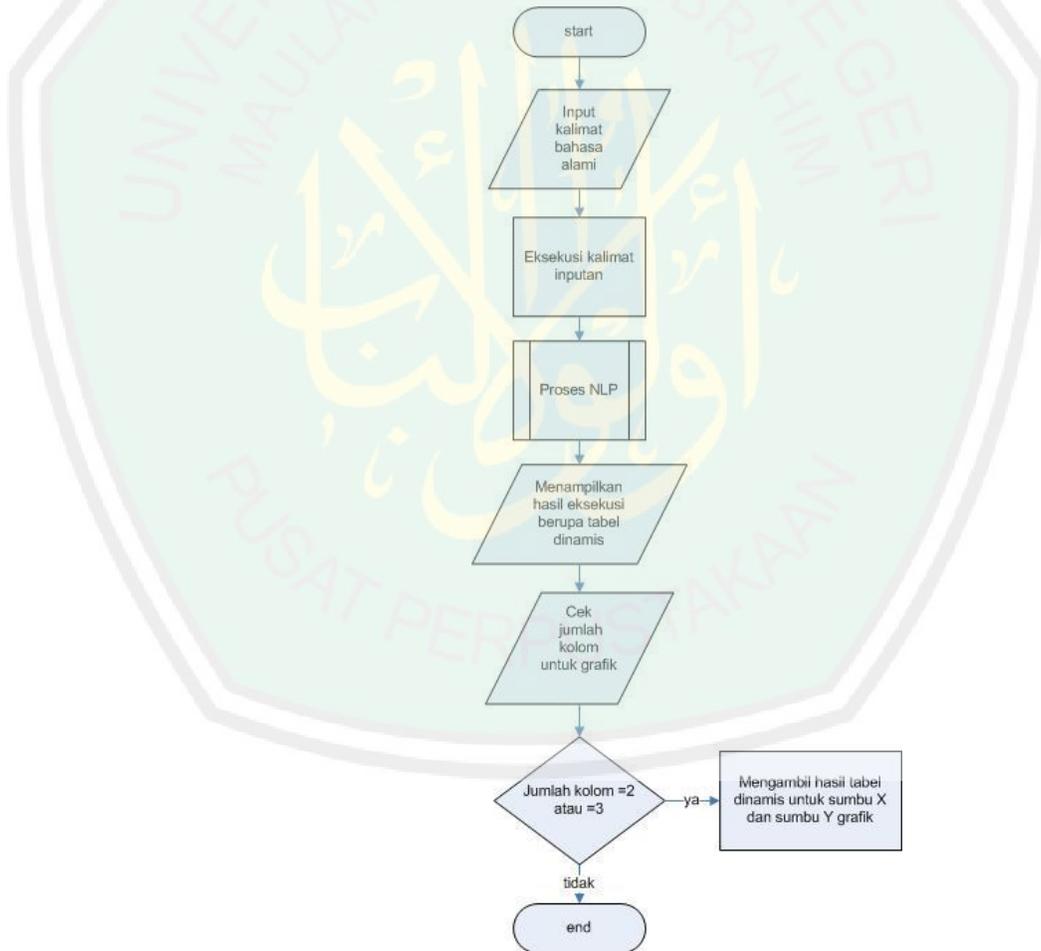
Pada **gambar 3.8** sistem menerapkan langkah Natural Language Processing sesuai blok diagram yang telah diulas diatas. Pada proses *SQL execution* sistem menerjemahkan array indonesia ke SQL kemudian melakukan *Query Transformation* di mana sistem akan menganalisis bagaimana dari arti kata – arti kata tersebut membentuk query yang utuh.



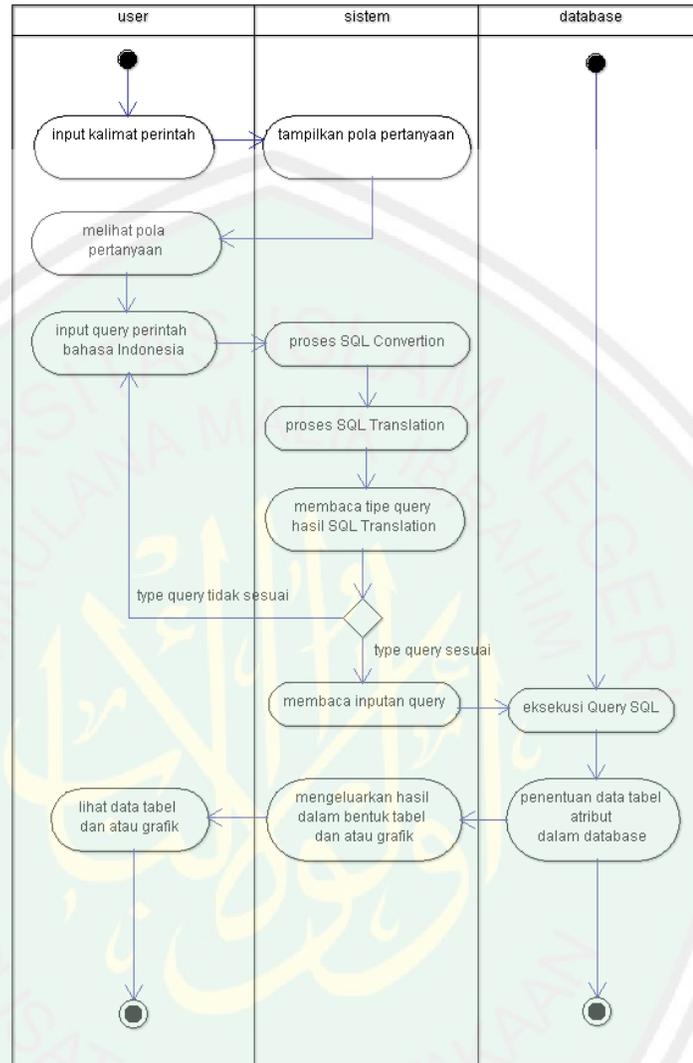
Gambar 3. 8 Flowchart Proses NLP

3.1.2.4 Proses Pembangkit grafik

Pada **gambar 3.9** sistem akan memproses tabel dinamis yang telah ter eksekusi menjadi grafik batang. Sistem akan mengecek jumlah kolom untuk grafik kemudian mengambil hasil tabel dinamis untuk sumbu X dan sumbu Y. Untuk tahapan sistem secara global dapat dilihat pada **gambar 3.10**



Gambar 3. 9 Flowchart Pembangkit grafik



Gambar 3. 10 Activity Diagram ekstraksi data grafik pada DBMS

3.1.3 Percobaan Manual

a) Kasus

Sebagai contoh dilakukan percobaan manual memproses kalimat perintah yang kemudian dieksekusi kedalam sistem natural language

pembangkit grafik. Inputan yang ada yaitu A1, A2 dan A3. Berikut proses manualnya:

A1 : tampilkan jumlah dosen fakultas saintek

A2 : tunjukkan jumlah mahasiswa yang fakultasnya bernama saintek

A3 : berapa total mahasiswa tiap jurusan yang angkatan 2010 dan alamatnya di jombang

b) Proses *SQL Conversion*

Masing-masing dari A1, A2, dan A3 diproses sesuai dengan urutan tahapan *SQL Conversion*. Berikut ini merupakan contoh hasil pemrosesan dari tiap tahap dimana tahap 1). *tokenizing*, (2). *filtering*, (3) *parsing*.

A1 : tampilkan jumlah dosen fakultas saintek

1. tampilkan||jumlah||dosen||fakultas||saintek
2. tampilkan||jumlah||dosen||fakultas||saintek
3. tampilkan → KP
jumlah → Kata_Kuantitas
dosen → TN
fakultas → FN
saintek → Data

A2 : tunjukkan jumlah mahasiswa yang fakultasnya bernama saintek

1. tunjukkan||jumlah||mahasiswa||yang||fakultasnya||bernama||saintek

2. tunjukkan||jumlah||mahasiswa||yang||fakultas nya||bernama||saintek
3. tunjukkan → KP
jumlah → Kata_Kuantitas
mahasiswa → TN
yang → Kata_Sambung1
fakultas → FN
saintek → Data

A3 : berapa totalnya mahasiswa tiap jurusan yang angkatan 2010 dan alamatnya di jombang

1. berapa||totalnya||mahasiswa||tiap||jurusan||yang||angkatan||2010||dan
||alamatnya||di||jombang
2. berapa||totalnya||mahasiswa||tiap||jurusan||yang||angkatan||2010||dan
||alamat nya||berada||di||jombang
3. berapa → KP
total → Kata_Kuantitas
mahasiswa → TN
tiap → Kata_Hubung
jurusan → FN
yang → Kata_Sambung1
angkatan → FN
2010 → Data
Dan → Kata_Sambung2
Alamat → FN
Jombang → Data

c) Proses *SQL Translation*

Dari proses sebelumnya, yaitu *SQL Conversion* yang berisi *tokenizing*, *filtering*, dan *parsing*. Dari proses parsing maka diperoleh

term-term yang dalam tahap ini merupakan proses penentuan *type query* tiap tiap inputan.

A1

tampilkan → KP
jumlah → Kata_Kuantitas
dosen → TN
fakultas → FN
saintek → Data

TYPEQUERY1 → { KP, kata_kuantitas, TN, FN, Data }

A2

tunjukkan → KP
jumlah → Kata_Kuantitas
mahasiswa → TN
yang → Kata_Sambung1
fakultas → FN
saintek → Data

TYPEQUERY3 → { KP, kata_kuantitas, TN, Kata_Sambung1, FN, Data }

A3

berapa → KP
total → Kata_Kuantitas
mahasiswa → TN
tiap → Kata_Hubung
jurusan → FN
yang → Kata_Sambung1
angkatan → FN
2010 → Data
Dan → Kata_Sambung2
Alamat → FN
Jombang → Data

TYPEQUERY5 → { KP, kata_kuantitas, TN, Kata_Hubung, FN, Kata_Sambung1, FN, Data, Kata_Sambung2, FN, Data }

d) Proses *SQL Execution*

Proses selanjutnya adalah *SQL Execution* dimana term yang terbentuk akan diterjemahkan kedalam format SQL.

A1

TYPEQUERY1 → { tampilkan, jumlah, dosen, fakultas, saintek }

TYPEQUERY1 → { select count(*), fakultas FROM mahasiswa WHERE
fakultas = 'saintek' GROUP BY fakultas }

A2

TYPEQUERY3 → { tunjukkan || jumlah || mahasiswa || yang || fakultas ||
saintek }

TYPEQUERY3 → { select count(*), fakultas FROM mahasiswa WHERE
fakultas = 'saintek' GROUP BY fakultas }

A3

TYPEQUERY5 → { berapa || total || mahasiswa || tiap || jurusan || yang ||
angkatan || 2010 || dan || alamat || jombang }

TYPEQUERY5 → { select count(*), fakultas , angkatan FROM mahasiswa
WHERE angkatan = '2010' and alamat = 'jombang'
'GROUP BY fakultas }

e) Proses Pembangkit grafik

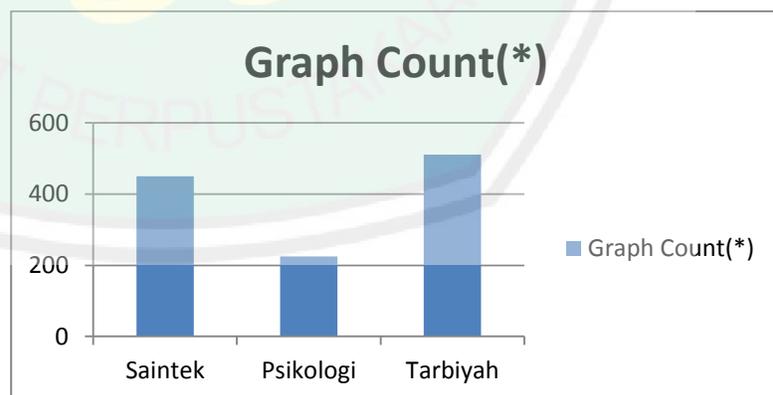
Proses Pembangkit grafik adalah proses terakhir di mana grafik yang disajikan terbentuk dari tabel dinamis hasil eksekusi query pada kolom database. Jika jumlah kolom pada tabel sama dengan 2 atau 3, maka sistem akan mengambil sumbu X dan sumbu Y.

Contoh tabel yang terbentuk:

Tabel 3. 7 contoh tabel hasil

| fakultas | Count(*) |
|-----------|----------|
| Saintek | 450 |
| Psikologi | 225 |
| Tarbiyah | 510 |

Maka grafik yang membentuk sumbu X dan sumbu Y yang diambil dari tabel hasil.



Gambar 3. 11 Pembangkit Grafik dari tabel hasil

3.1.4 Desain Interface

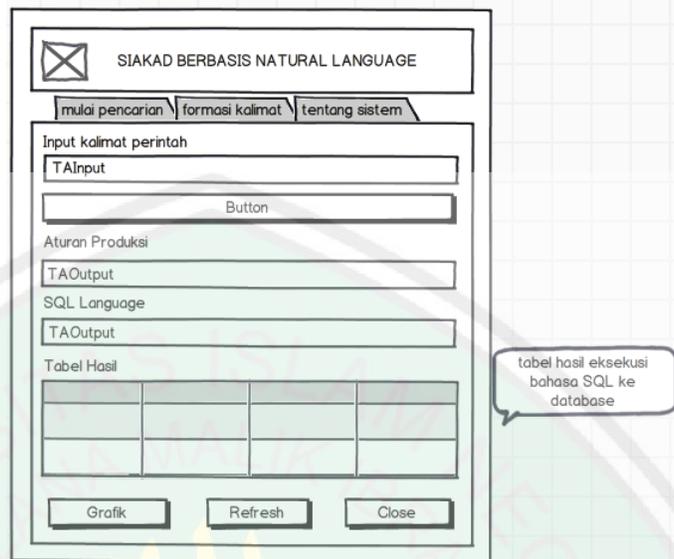
Aplikasi natural language processing ini dibangun dengan bahasa pemrograman java menggunakan IDE netbeans 7.2 dan disambungkan menggunakan MySQL untuk koneksi databasenya. Pada aplikasi ini terdapat satu form tampilan dengan beberapa sub proses sebagaimana berikut :

a. Form tampilan interface

Form ini merupakan tampilan awal ketika user mulai menjalankan aplikasi. Terdapat beberapa text area untuk menginputkan serta menampilkan hasil. Rancangan tampilan awal dapat dilihat pada **gambar 3.11** dan **gambar 3.12**

The screenshot shows a window titled "SIKAD BERBASIS NATURAL LANGUAGE". At the top, there is a menu bar with three items: "mulai pencarian", "formasi kalimat", and "tentang". Below the menu bar, there are three text input fields labeled "Input kalimat perintah", "Aturan Produksi", and "SQL Language". A "Button" is positioned below the "Input kalimat perintah" field. Below the input fields is a table labeled "Tabel Hasil" with 4 columns and 2 rows. At the bottom of the window, there are three buttons: "Grafik", "Refresh", and "Close".

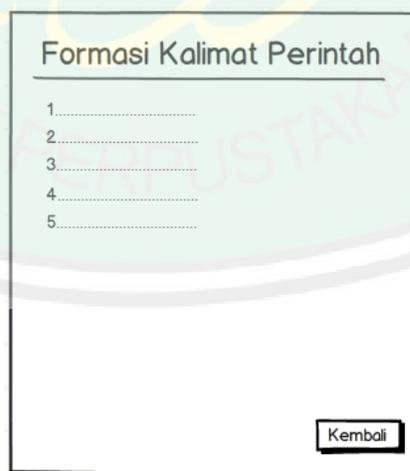
Gambar 3.12 Rancangan interface aplikasi Pembangkit grafik berbasis natural language



Gambar 3.13 Rancangan sub proses aplikasi Pembangkit grafik berbasis natural language

b. Form formasi kalimat inputan

Halaman ini berisi beberapa formasi kalimat inputan yang dapat diproses oleh sistem. Rancangan tampilan halamannya dapat dilihat pada **gambar 3.14**

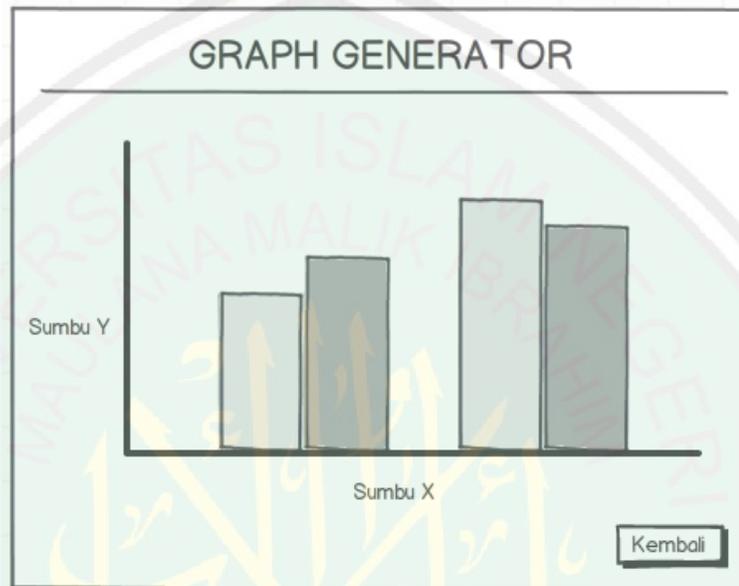


Gambar 3. 14 Rancangan form formasi kalimat inputan aplikasi Pembangkit grafik berbasis natural language

c. Form Pembangkit grafik

Halaman ini adalah halaman hasil akhir dari sistem yaitu pembangkit grafik.

Rancangan tampilan halamannya dapat dilihat pada **gambar 3.15**



Gambar 3.15 Rancangan form pembangkit grafik aplikasi Pembangkit grafik berbasis natural language

BAB IV

HASIL DAN PEMBAHASAN

Langkah selanjutnya setelah perancangan sebuah sistem adalah mengimplementasikan sistem yang telah dibuat. Langkah terakhir adalah pengujian rancangan sistem secara keseluruhan, langkah ini dilakukan karena sangat dibutuhkan untuk memastikan apakah hasil akhir sistem sesuai dengan rancangan sistem.

4.1 Lingkungan Uji Coba

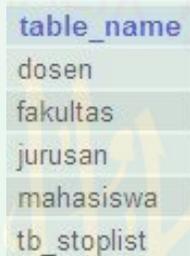
Pada tabel 4.1 dipaparkan spesifikasi lingkungan yang digunakan untuk menguji jalannya aplikasi yang diimplementasikan. Untuk mengimplementasikan sistem yang dibuat ada beberapa komponen yang digunakan yaitu komponen perangkat lunak (*software*) dan perangkat keras (*hardware*). Spesifikasi kebutuhan Perangkat keras dan kebutuhan perangkat lunak yang di gunakan di dalam penelitian ini sebagai berikut :

Tabel 4. 1 Tabel kebutuhan hardware dan kebutuhan software

| | |
|--------------------|---|
| Kebutuhan hardware | <ul style="list-style-type: none"> a. Laptop Processor Core 2 Duo 2.10 GHz b. Memory 2 GB c. Hardisk 160 GB |
| Kebutuhan Software | <ul style="list-style-type: none"> a. Microsoft Windows 7 b. Xampp 1.7.3 c. Netbeans IDE 6.9.1 d. Microsoft Office Visio 2003 e. Google Chrome f. Library java <i>jfreechart-1.0.14</i> g. Library java <i>mysql-connector-java-5.1.18-bin</i> |

4.2 Data Uji Coba

Data yang digunakan dalam sistem ini merupakan database yang dikoneksikan dengan Xampp 1.7.3. Penulis menggunakan database akademis sampling yang berisi tabel mahasiswa dan dosen. Untuk penjabaran tabel dan kolom apa saja yang digunakan dapat dilihat pada **gambar 4.1** dan **gambar 4.2**



| table_name |
|-------------|
| dosen |
| fakultas |
| jurusan |
| mahasiswa |
| tb_stoplist |

Gambar 4. 1 Daftar nama tabel



| column_name |
|----------------|
| id_dosen |
| nama |
| golongan |
| jabatan |
| fakultas |
| id_fakultas |
| nama_fakultas |
| id_jurusan |
| nama_jurusan |
| id |
| nim |
| jenis_kelamin |
| anggota_htq |
| alamat |
| tahun_lahir |
| jalan |
| asal_sekolah |
| telepon |
| jurusan |
| angkatan |
| pekerjaan_ortu |
| id_stoplist |
| stoplist |

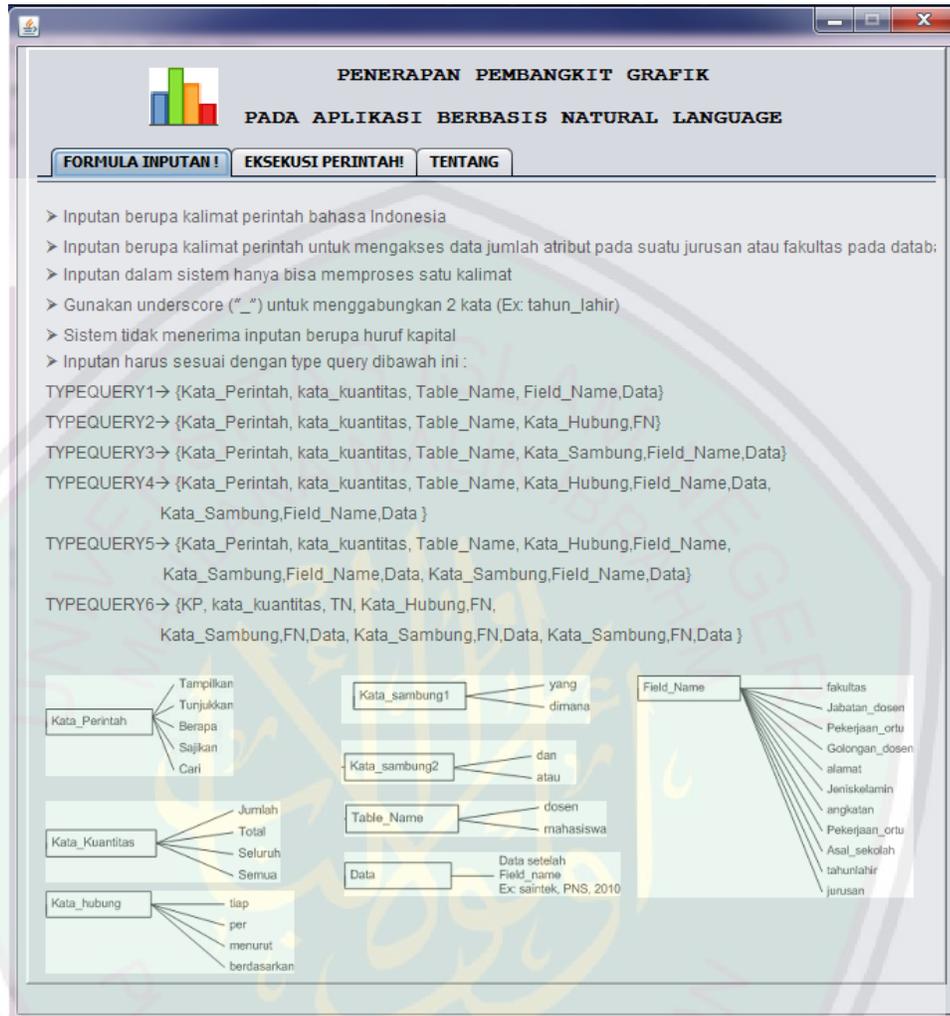
Gambar 4. 2 daftar nama kolom

4.3 Implementasi Sistem

Implementasi sistem *natural language processing* tersusun atas beberapa proses yang telah dirumuskan.

4.3.1 Halaman Awal

Halaman awal berisi penjelasan sistem dan Formasi inputan yang bisa diterima dan diproses oleh sistem. Halaman ini juga berisi penjelasan type query yang terbentuk dari kata-kata yang dikelompokkan sesuai jenisnya.

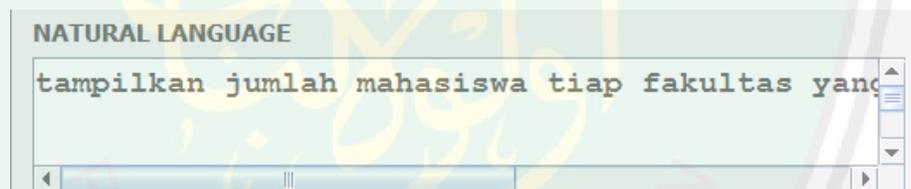


Gambar 4. 3 Halaman Awal

4.3.2 SQL Conversion

Kalimat perintah yang diinput oleh user akan melalui proses *SQL Conversion*, di mana dalam proses ini kalimat akan di *scanning* terlebih dahulu guna membaca kalimat inputan. Masih dalam tahap *SQL Conversion*, sistem kemudian memecah kalimat berdasarkan spasi dalam bentuk token-token tanpa memperhatikan keterkaitan antar kata, yang dalam pemrograman java dilakukan

dengan memanggil method `split`. Lihat **gambar 4.3** Selanjutnya dalam proses filtering yaitu membuang karakter yang *irrelevant*. Dalam proses ini dilakukan dengan cara wordlist (menyimpan kata penting). Pembuangan kata- kata yang *irrelevant* dilakukan dengan mengecek token dengan daftar kumpulan kata stopword yang terdapat dalam database. Jika kata tersebut sama dengan kata dalam stopword maka akan dihapus dari daftar. Langkah selanjutnya dalam *SQL Conversion* adalah *parsing*. Parser adalah proses inti dari NLP. Proses parsing dilakukan untuk melakukan pengenalan terhadap KAL_PER atau kalimat perintah. Parser melakukan pelacakan kalimat inputan untuk mendapatkan langkah pembentuknya. Token akan dikelompokkan menjadi struktur type query sesuai dengan aturan produksi yang telah disusun. Lihat **gambar 4.4**.



Gambar 4. 4 Contoh Inputan User

Berikut potongan *source code tokenizing*

```
proses = input.split(" ");
String nat1[] = nat.replaceAll("\n", "").split(" ");

for (int a = 0; a < nat1.length; a++) {
    Sistem.out.println(nat1[a] + " ");
}
```

Berikut potongan *source code filtering*

```
private void stop() {
    nat = taInput.getText();
    try {
        Statement stat = k.logon().createStatement();
        String sql6 = "select stoplist from tb_stoplist";
        ResultSet rs = stat.executeQuery(sql6);
        while (rs.next()) {
            nat = nat.replace("" + rs.getString(1) + "", "");
        }
        nat = nat.replace("", " ");
        Sistem.out.println(nat);
    } catch (Exception ex) {
        Sistem.out.println(ex.getMessage());
    }
}

tbstop();
```

ATURAN PRODUKSI

```
====>Perintah [tampilkan=KP]

====>Atribut [jumlah=kata_jumlah]
[mahasiswa=TName]
```

Gambar 4. 5 Proses parsing berdasarkan aturan produksi

Berikut potongan *source code parsing* aturan produksi

```
String aturanbiasa[][] = {
    {"KP", "tampilkan"},
    {"KP", "tunjukkan"},
    {"kata_kuantitas", "jumlah"},
    {"kata_kuantitas", "total"},
    {"kata_kuantitas", "seluruh"},
    {"KHub", "tiap"},
    {"KHub", "per"},
    {"KHub", "menurut"},
    {"KHub", "berdasarkan"},
    {"KS", "yang"},
    {"KS", "dan"}
};
```

```

void setAtributKondisi() {
fra_atr = "";
fra_kond = "";
perintah = "";
for (int k = 0; k < proses.length; k++) {
//ketemu = ndetect boolean ada tidaknya data di dlm "array
aturan"
boolean ketemu = false;
String string = proses[k];
for (int i = 0; i < aturan.length; i++) {
String strings = aturan[i][1];
if (string.equalsIgnoreCase(strings)) {
//k = data inputan "tampilkan jumlah dst"
if (k == 0) {
perintah = perintah + " [" + aturan[i][1] + "=" + aturan[i][0] +
"]\n\n";
ketemu = true;
} else {
ketemu = true;
String strings2 = aturan[i][0];
if (strings2.equalsIgnoreCase("KS")) {
kond = true;
}
if (!kond) {
fra_atr = fra_atr + " [" + aturan[i][1] + "=" + aturan[i][0] +
"]\n";
} else {
fra_kond = fra_kond + " [" + aturan[i][1] + "=" + aturan[i][0] +
"]\n";}}}}
if (!ketemu) {
kond = true;
fra_kond = fra_kond + " [" + proses[k] + "]\n";}}
taaturan.setText("====>Perintah");
taaturan.append(perintah);
taaturan.append("====>Atribut");
taaturan.append(fra_atr);
taaturan.append("\n====>Kondisi");
taaturan.append(fra_kond);
taaturan.append("=====");}

```

4.3.3 SQL Translation

Tahap Translator ini berfungsi untuk mengubah pohon sintak hasil parsing ke typequery yang telah ditetapkan pada sistem.

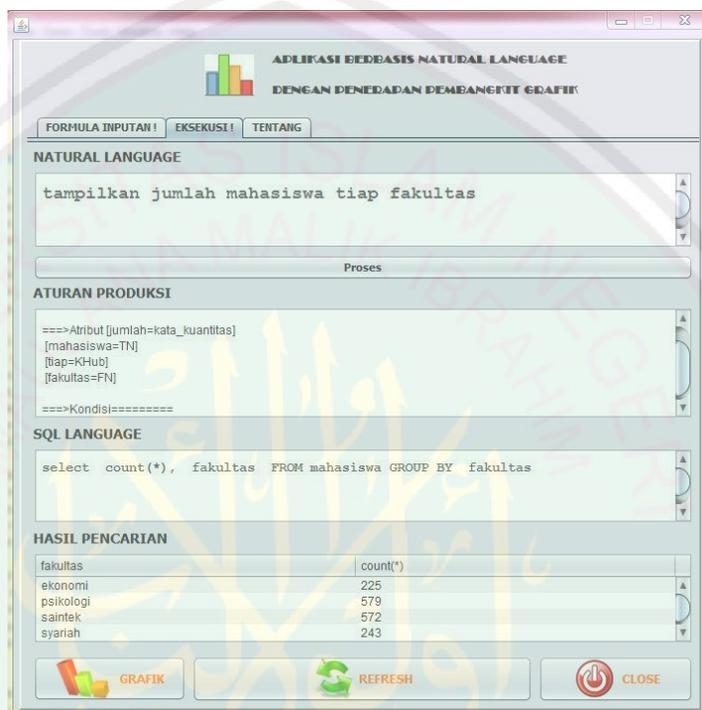
Berikut potongan *source code SQL Translation*

```
String typequery1[] = {"KP", "kata_kuantitas", "TN", "FN"};
String typequery2[] = {"KP", "kata_kuantitas", "TN", "KHub", "FN"};
String typequery3[] = {"KP", "kata_kuantitas", "TN", "KS", "FN"};
String typequery4[] = {"KP", "kata_kuantitas", "TN", "KHub", "FN", "KS",
"FN"};
String typequery5[] = {"KP", "kata_kuantitas", "TN", "KHub", "FN", "KS",
"FN", "KS", "FN"};
String typequery6[] = {"KP", "kata_kuantitas", "TN", "KHub", "FN", "KS",
"FN", "KS", "FN", "KS", "FN"};
if (Arrays.equals(typequery1, d.toArray())) {
Sistem.out.println("typequery1");
typequery1(nat1);
} else if (Arrays.equals(typequery2, d.toArray())) {
Sistem.out.println("typequery2");
typequery2(nat1);
} else if (Arrays.equals(typequery3, d.toArray())) {
Sistem.out.println("typequery3");
typequery3(nat1);
} else if (Arrays.equals(typequery4, d.toArray())) {
Sistem.out.println("typequery4");
typequery4(nat1);
} else if (Arrays.equals(typequery5, d.toArray())) {
Sistem.out.println("typequery5");
typequery5(nat1);
} else if (Arrays.equals(typequery6, d.toArray())) {
Sistem.out.println("typequery6");
typequery6(nat1);
}
}
```

4.3.4 SQL Execution

Tahap ini merupakan tahap eksekusi sistem terhadap inputan user, sehingga setiap inputan yang diterima akan diterjemahkan ke format SQL (SQL language) yang sesuai dengan tipe formasi query yang telah ditentukan sebelumnya. Pada proses *SQL execution* ini sistem membentuk kondisi-kondisi sehingga keluar hasil output berupa perintah SQL dengan type query yang dipanggil. Jika dari hasil

tersebut terdapat kecocokan maka proses akan berlanjut akan tetapi jika hasil Type Query tidak ditemukan maka terjadi pesan error. Lihat **gambar 4. 5** dan **gambar 4. 6**



Gambar 4. 6 Proses hasil SQL Execute

Berikut potongan *source code* SQL Execution

```
private void typequery1(String[] d) {
    setAtributKondisi();
    for (int i = 0; i < indonesia.length; i++) {
        for (int j = 0; j < proses.length; j++) {

            if (indonesia[i].equals(proses[j])) {
                proses[j] = sql[i];
            }
        }
    }

    String sqlini = "" + proses[0] + " " + proses[1] + " " + proses[3] + " FROM " +
        proses[2] + " WHERE " + proses[3] + "=" + proses[4] + " GROUP BY " +
        proses[3] + """;
    taOutput.setText(sqlini);
}
```

```

}

private void typequery2(String[] d) {
    setAtributKondisi();
    for (int i = 0; i < indonesia.length; i++) {
        for (int j = 0; j < proses.length; j++) {

            if (indonesia[i].equals(proses[j])) {
                proses[j] = sql[i];
            }
        }
    }
    String sqlini = ""+proses[0]+" "+proses[1]+" "+ proses[4] + " FROM " +
    proses[2] + " GROUP BY " + proses[4] + """;
    taOutput.setText(sqlini);
}

private void typequery3(String[] d) {
    setAtributKondisi();
    for (int i = 0; i < indonesia.length; i++) {
        for (int j = 0; j < proses.length; j++) {

            if (indonesia[i].equals(proses[j])) {
                proses[j] = sql[i];
            }
        }
    }
    String sqlini = ""+proses[0]+" "+proses[1]+" " + proses[4] + " FROM " +
    proses[2] + " WHERE " + proses[4] + "=" + proses[5] + " GROUP BY " +
    proses[4] + """;
    taOutput.setText(sqlini);
}

private void typequery4(String[] d) {
    setAtributKondisi();
    for (int i = 0; i < indonesia.length; i++) {
        for (int j = 0; j < proses.length; j++) {

            if (indonesia[i].equals(proses[j])) {
                proses[j] = sql[i];
            }
        }
    }
    String sqlini = ""+proses[0]+" "+proses[1]+" " + proses[4] + "," + proses[6] +
    " FROM " + proses[2] + " WHERE " + proses[6] + "=" + proses[7] + "
    GROUP BY " + proses[4] + """;
    taOutput.setText(sqlini);
}

```

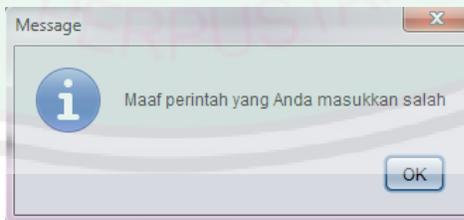
```

}
private void typequery5(String[] d) {
setAtributKondisi();
for (int i = 0; i < indonesia.length; i++) {
for (int j = 0; j < proses.length; j++) {

if (indonesia[i].equals(proses[j])) {
proses[j] = sql[i];
}
}
}
String sqlini = ""+proses[0]+" "+proses[1]+" " + proses[4] + "," + proses[6] +
" FROM " + proses[2] + " WHERE " + proses[6] + "=" + proses[7] + " and "
+ proses[9] + " = "" + proses[10]+""GROUP BY " + proses[4] + """;
taOutput.setText(sqlini);
}
private void typequery6(String[] d) {
setAtributKondisi();
for (int i = 0; i < indonesia.length; i++) {
for (int j = 0; j < proses.length; j++) {

if (indonesia[i].equals(proses[j])) {
proses[j] = sql[i];
}
}
}
String sqlini = ""+proses[0]+" "+proses[1]+" " + proses[4] + ","+proses[6]+
FROM " + proses[2] + " WHERE " + proses[6] + " = " + proses[7] + " and
"+proses[9]+
= ""+proses[10]+"" and ""+proses[12]+
""+proses[13]+""GROUP BY " + proses[4] + """;
taOutput.setText(sqlini); }

```



Gambar 4. 7 Pesan error ketika terjadi kesalahan pada inputan

4.3.5 Pembangkit grafik

Pembuatan grafik frekuensi terbentuk dari SQL yang telah tereksekusi pada proses *SQL execution*. Sumbu X dan sumbu Y dari grafik otomatis mengikuti tabel yang telah terbentuk sebelumnya di sistem. Lihat **gambar 4.7** Gambar grafik frekuensi yang digunakan adalah grafik bar. Lihat **gambar.4.8**

| HASIL PENCARIAN | |
|-----------------|----------|
| fakultas | count(*) |
| ekonomi | 225 |
| psikologi | 579 |
| saintek | 572 |
| syariah | 243 |

Gambar 4. 8 hasil pencarian pada tabel dinamis

Berikut potongan *source code* pembangkit grafik

```

ArrayList<String> xy2;
ArrayList<String> nama;

//mengambil grafik untuk kolom
void selekNamaXY(ArrayList<String> a) {
Sistem.out.println("size" + a.size());
xy2 = new ArrayList<String>();
nama = new ArrayList<String>();

//jika jumlah kolom 2
if (tabel.getColumnCount() == 2) {
for (int i = 0; i < a.size(); i++) {
if (i % 2 != 0) {
xy2.add(a.get(i));
} else {
nama.add(a.get(i));
}
}
gantiArray();

//jika jumlah kolom 3
} else if (tabel.getColumnCount() == 3) {
for (int i = 0; i < a.size(); i++) {
if (i % 3 == 0) {
} else if (i % 3 == 1) {
nama.add(a.get(i));
}
}
}

```

```

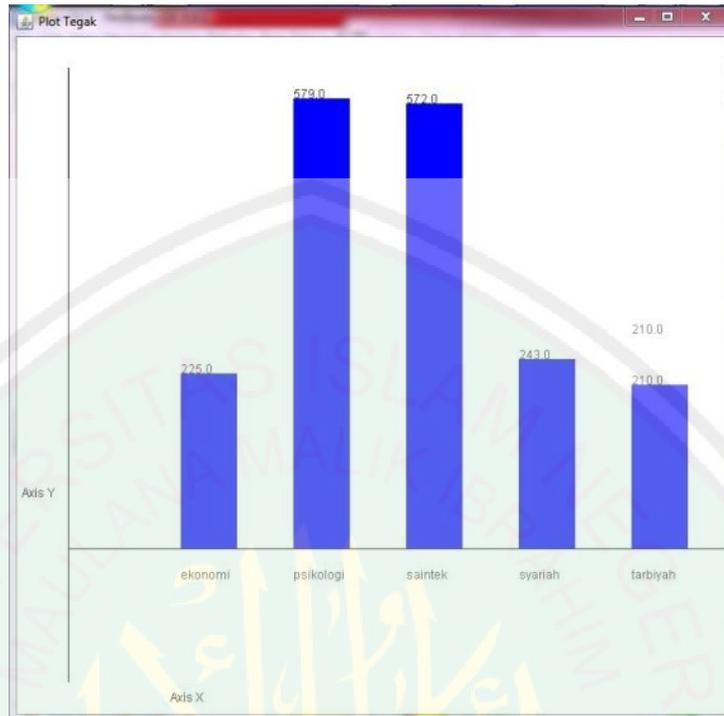
} else {
xy2.add(a.get(i));
}}
gantiArray();
}

Sistem.out.println("");
}
double[] dxy;
double[] dyy;

void gantiArray() {
int s = xy2.size();
dxy = new double[s];
for (int i = 0; i < s; i++) {
dxy[i] = Double.parseDouble(xy2.get(i));
}}

void tampilGrafik() {
int bx = 1300;
int by = 700;
int l = dxy.length;
double[] x = new double[l];
plottegak2 p = new plottegak2();
//p.frame( dxy, bx, by);
int nix = p.getNix() * 4;
double midX = ((bx - nix) / l) / 2;
double lebar = midX / 2;
for (int i = 0; i < l; i++) {
//      if(i==0){x[i]=10;}else{
x[i] = midX * (i + 1);
}
}
//int penetral = p.getPenetralY();
p.frame(x, dxy, bx, by, lebar, nama);
}

```



Gambar 4. 9 Hasil akhir pembangkit grafik berupa *bar chart*

4.4 Uji Coba Sistem dan Analisis Sistem

4.4.1 Analisis Pengujian pada Tabel Mahasiswa dan Tabel Dosen

Pada pengujian tabel mahasiswa dan tabel dosen, peneliti mencoba menginputkan kalimat perintah dan meminta sistem untuk mengeluarkan jumlah suatu atribut pada tabel yang sesuai dengan kalimat yang diperintahkan, kemudian dilakukan pengecekan apakah sudah sesuai dengan format query SQL dan hasil tabel pada DBMS.

Tabel 4. 2 Sampel kalimat uji coba yang bernilai benar karena sesuai dengan Format SQL

| No. | Kalimat yang diuji dan Format SQL yang didapat | Format SQL yang diharapkan | Eksekusi aplikasi | Eksekusi Tabel DBMS |
|-----|---|--|-------------------------------|-------------------------------|
| 1. | tampilkan jumlah mahasiswa fakultas saintek Select count(*), fakultas FROM mahasiswa WHERE fakultas = 'saintek' GROUP BY fakultas | Select count(*), fakultas FROM mahasiswa WHERE fakultas = 'saintek' GROUP BY fakultas | Hasil lihat lampiran gambar 1 | Hasil lihat lampiran gambar 1 |
| 2. | berapa totalnya mahasiswa tiap jenis_kelamin yang alamat pasuruan dan pekerjaan_ortu tani/nelayan select count(*), jenis_kelamin,alamat FROM mahasiswa where alamat='pasuruan' and pekerjaan_ortu = 'tani/nelayan'GROUP BY jenis_kelamin | select count(*), jenis_kelamin,alamat FROM mahasiswa where alamat='pasuruan' and pekerjaan_ortu = 'tani/nelayan'GROUP BY jenis_kelamin | Hasil lihat lampiran gambar 2 | Hasil lihat lampiran gambar 2 |
| 3. | Sajikan semua mahasiswa yang beralamatkan pasuruan select count(*), alamat FROM mahasiswa where alamat='pasuruan' GROUP BY alamat | select count(*), alamat FROM mahasiswa where alamat='pasuruan' GROUP BY alamat | Hasil lihat lampiran gambar 3 | Hasil lihat lampiran gambar 3 |
| 4. | cari jumlah dosen dimana memiliki jabatan_dosen lektor | select count(*), jabatan_dosen FROM dosen where | Hasil lihat lampiran | Hasil lihat lampiran |

| | | | | |
|----|---|--|---|--|
| | select count(*), jabatan_dosen FROM dosen where jabatan_dosen= 'lektor' GROUP BY jabatan_dosen | jabatan_dosen= 'lektor' GROUP BY jabatan_dosen | gambar 4 | gambar 4 |
| 5. | cari total setiap mahasiswa dimana tahun_lahirnya 1990 select count(*), tahun_lahir FROM mahasiswa where tahun_lahir= '1990' GROUP BY tahun_lahir | select count(*), tahun_lahir FROM mahasiswa where tahun_lahir= '1990' GROUP BY tahun_lahir | Hasil lihat lampiran gambar 5 | Hasil lihat lampiran gambar 5 |
| 6. | berapa semua mahasiswa per jurusan yang alamat malang dan angkatan 2010 dan pekerjaan_orstu PNS select count(*), jurusan ,alamat FROM mahasiswa where alamat = 'malang' and angkatan = '2010' and pekerjaan_orstu = 'PNS' GROUP BY jurusan | select count(*), jurusan ,alamat FROM mahasiswa where alamat = 'malang' and angkatan = '2010' and pekerjaan_orstu = 'PNS' GROUP BY jurusan | Hasil lihat lampiran gambar 6 | Hasil lihat lampiran gambar 6 |
| 7. | berapa total mahasiswa angkatan 2010 Select count(*), angkatan FROM mahasiswa WHERE angkatan = '2010' GROUP BY angkatan | Select count(*), angkatan FROM mahasiswa WHERE angkatan = '2010' GROUP BY angkatan | Hasil lihat lampiran gambar 7 | Hasil lihat lampiran gambar 7 |
| 8. | berapa jumlahnya mahasiswa menurut jurusan dimana angkatan 2011 dan alamat jombang atau alamat pasuruan | select count(*), jurusan ,angkatan FROM mahasiswa where angkatan = '2011' and alamat = 'jombang' or alamat = | Hasil lihat lampiran gambar 8 | Hasil lihat lampiran gambar 8 |

| | | | | |
|-----|--|---|--|--|
| | select count(*), jurusan ,angkatan FROM mahasiswa where angkatan = '2011' and alamat = 'jombang' or alamat = 'pasuruan'GROUP BY jurusan | 'pasuruan'GROUP BY jurusan | | |
| 9. | tunjukkan seluruh mahasiswa fakultas tarbiyah | Select count(*), fakultas FROM mahasiswa WHERE fakultas = 'tarbiyah' GROUP BY fakultas | Hasil lihat lampiran gambar 9 | Hasil lihat lampiran gambar 9 |
| | Select count(*), fakultas FROM mahasiswa WHERE fakultas = 'tarbiyah' GROUP BY fakultas | | | |
| 10. | sajikan semua mahasiswa pekerjaan_ortu PNS | select count(*), pekerjaan_ortu FROM mahasiswa WHERE pekerjaan_ortu= 'PNS' GROUP BY pekerjaan_ortu | Hasil lihat lampiran gambar 10 | Hasil lihat lampiran gambar 10 |
| | select count(*), pekerjaan_ortu FROM mahasiswa WHERE pekerjaan_ortu= 'PNS' GROUP BY pekerjaan_ortu | | | |
| 11. | berapa total mahasiswa dimana fakultasnya bernama psikologi | select count(*), fakultas FROM mahasiswa where fakultas = 'psikologi' GROUP BY fakultas | Hasil lihat lampiran gambar 11 | Hasil lihat lampiran gambar 11 |
| | select count(*), fakultas FROM mahasiswa where fakultas = 'psikologi' GROUP BY fakultas | | | |
| 12. | tampilkan semua mahasiswa per fakultas | select count(*), fakultas FROM mahasiswa GROUP BY fakultas | Hasil lihat lampiran gambar 12 | Hasil lihat lampiran gambar 12 |
| | select count(*), fakultas FROM mahasiswa GROUP BY fakultas | | | |

| | | | | |
|-----|--|--|--------------------------------|--------------------------------|
| 13. | tampilkan semua mahasiswa yang tahun_lahirnya 1992 | Select count(*), tahun_lahir FROM mahasiswa WHERE tahun_lahir = '1992' GROUP BY tahun_lahir | Hasil lihat lampiran gambar 13 | Hasil lihat lampiran gambar 13 |
| | Select count(*), tahun_lahir FROM mahasiswa WHERE tahun_lahir = '1992' GROUP BY tahun_lahir | | | |
| 14. | tampilkan semua dosen dimana golongan_dosen III/c | select count(*), golongan_dosen FROM dosen where golongan_dosen= 'III/c' GROUP BY golongan_dosen | Hasil lihat lampiran gambar 14 | Hasil lihat lampiran gambar 14 |
| | select count(*), golongan_dosen FROM dosen where golongan_dosen= 'III/c' GROUP BY golongan_dosen | | | |
| 15. | sajikan total dosen yang jabatan_dosennya sebagai lektor | select count(*), jabatan_dosen FROM dosen where jabatan_dosen= 'lektor' GROUP BY jabatan_dosen | Hasil lihat lampiran gambar 15 | Hasil lihat lampiran gambar 15 |
| | select count(*), jabatan_dosen FROM dosen where jabatan_dosen= 'lektor' GROUP BY jabatan_dosen | | | |
| 16. | tampilkan semua dosen yang golongan_dosennya III/c | select count(*), golongan_dosen FROM dosen where golongan_dosen= 'III/c' GROUP BY golongan_dosen | Hasil lihat lampiran gambar 16 | Hasil lihat lampiran gambar 16 |
| | select count(*), golongan_dosen FROM dosen where golongan_dosen= 'III/c' GROUP BY golongan_dosen | | | |
| 17. | Berapa totalnya mahasiswa per jurusan yang angkatannya 2010 dan | select count(*), jurusan ,angkatan FROM mahasiswa where angkatan = | Hasil lihat lampiran gambar | Hasil lihat lampiran gambar |

| | | | | |
|-----|---|--|--|--|
| | pekerjaan_orstu PNS atau pekerjaan_orstu TNI/polisi | '2010' and pekerjaan_orstu = 'PNS' or pekerjaan_orstu = 'TNI/polisi'GROUP BY jurusan | 17 | 17 |
| | select count(*), jurusan ,angkatan FROM mahasiswa where angkatan = '2010' and pekerjaan_orstu = 'PNS' or pekerjaan_orstu = 'TNI/polisi'GROUP BY jurusan | | | |
| 18. | tampilkan jumlah mahasiswa menurut fakultas yang angkatan 2011 | select count(*), fakultas ,angkatan FROM mahasiswa where angkatan= '2011' GROUP BY fakultas | Hasil lihat lampiran gambar 18 | Hasil lihat lampiran gambar 18 |
| | select count(*), fakultas ,angkatan FROM mahasiswa where angkatan= '2011' GROUP BY fakultas | | | |
| 19. | tampilkan jumlah mahasiswa tiap jurusan yang tahun_lahir 1992 | select count(*), jurusan ,tahun_lahir FROM mahasiswa where tahun_lahir= '1992' GROUP BY jurusan | Hasil lihat lampiran gambar 19 | Hasil lihat lampiran gambar 19 |
| | select count(*), jurusan ,tahun_lahir FROM mahasiswa where tahun_lahir= '1992' GROUP BY jurusan | | | |
| 20. | tampilkan semua mahasiswa berdasarkan alamat dimana tahun_lahirnya 1992 | select count(*), alamat ,tahun_lahir FROM mahasiswa where tahun_lahir= '1992' GROUP BY jurusan | Hasil lihat lampiran gambar 20 | Hasil lihat lampiran gambar 20 |
| | select count(*), alamat ,tahun_lahir FROM mahasiswa where tahun_lahir= '1992' GROUP BY jurusan | | | |

| | | | | |
|-----|---|--|--|--|
| 21. | sajikan semua mahasiswa menurut jurusan dimana angkatan 2011 | select count(*), jurusan ,angkatan FROM mahasiswa where angkatan= '2011' GROUP BY jurusan | Hasil lihat lampiran gambar 21 | Hasil lihat lampiran gambar 21 |
| | select count(*), jurusan ,angkatan FROM mahasiswa where angkatan= '2011' GROUP BY jurusan | | | |
| 22. | sajikan semua mahasiswa tiap fakultas dimana jenis_kelamin perempuan | select count(*), fakultas ,jenis_kelamin FROM mahasiswa where jenis_kelamin= 'perempuan' GROUP BY fakultas | Hasil lihat lampiran gambar 22 | Hasil lihat lampiran gambar 22 |
| | select count(*), fakultas ,jenis_kelamin FROM mahasiswa where jenis_kelamin= 'perempuan' GROUP BY fakultas | | | |
| 23. | sajikan total mahasiswa tiap fakultasnya dimana jenis_kelamin laki-laki | select count(*), fakultas ,jenis_kelamin FROM mahasiswa where jenis_kelamin= 'laki-laki' GROUP BY fakultas | Hasil lihat lampiran gambar 23 | Hasil lihat lampiran gambar 23 |
| | select count(*), fakultas ,jenis_kelamin FROM mahasiswa where jenis_kelamin= 'laki-laki' GROUP BY fakultas | | | |
| 24 | berapa total mahasiswa tiap jurusan yang fakultas syariah dan jenis_kelamin perempuan | select count(*), jurusan , fakultas FROM mahasiswa where fakultas = 'syariah' and jenis_kelamin = 'perempuan'GROUP BY jurusan | Hasil lihat lampiran gambar 24 | Hasil lihat lampiran gambar 24 |
| | select count(*), jurusan , fakultas FROM mahasiswa where fakultas = 'syariah' and jenis_kelamin = | | | |

| | | | | |
|-----|--|--|--------------------------------|--------------------------------|
| | 'perempuan'GROUP BY jurusan | | | |
| 25. | sajikan semua mahasiswa tiap angkatan dimana alamat malang | select count(*), angkatan,alamat FROM mahasiswa where alamat='malang' GROUP BY angkatan | Hasil lihat lampiran gambar 25 | Hasil lihat lampiran gambar 25 |
| | select count(*), angkatan,alamat FROM mahasiswa where alamat='malang' GROUP BY angkatan | | | |
| 26. | tunjukkan semua mahasiswa tiap jurusan dimana pekerjaan_ortu wiraswasta | select count(*), jurusan ,pekerjaan_ortu FROM mahasiswa where pekerjaan_ortu='wiraswasta' GROUP BY jurusan | Hasil lihat lampiran gambar 26 | Hasil lihat lampiran gambar 26 |
| | select count(*), jurusan ,pekerjaan_ortu FROM mahasiswa where pekerjaan_ortu='wiraswasta' GROUP BY jurusan | | | |
| 27. | sajikan semua mahasiswa menurut tahun_lahir dimana alamat banyuwangi | select count(*), tahun_lahir,alamat FROM mahasiswa where alamat='banyuwangi' GROUP BY tahun_lahir | Hasil lihat lampiran gambar 27 | Hasil lihat lampiran gambar 27 |
| | select count(*), tahun_lahir,alamat FROM mahasiswa where alamat='banyuwangi' GROUP BY tahun_lahir | | | |
| 28 | cari jumlah mahasiswa tiap angkatan yang fakultas ekonomi | select count(*), angkatan, fakultas FROM mahasiswa where fakultas = 'ekonomi' GROUP BY angkatan | Hasil lihat lampiran gambar 28 | Hasil lihat lampiran gambar 28 |
| | select count(*), angkatan, fakultas FROM mahasiswa where fakultas = 'ekonomi' GROUP BY | | | |

| | | | | |
|-----|--|--|--------------------------------|--------------------------------|
| | angkatan | | | |
| 29. | tampilkan total mahasiswa tiap fakultas yang angkatan 2010 dan alamat malang | select count(*), fakultas ,angkatan FROM mahasiswa where angkatan='2010' and alamat = 'malang'GROUP BY fakultas | Hasil lihat lampiran gambar 29 | Hasil lihat lampiran gambar 29 |
| | select count(*), fakultas ,angkatan FROM mahasiswa where angkatan='2010' and alamat = 'malang'GROUP BY fakultas | | | |
| 30. | tunjukkan seluruh mahasiswa menurut fakultas yang alamat surabaya dan berjenis_kelamin perempuan | select count(*), fakultas ,alamat FROM mahasiswa where alamat= 'surabaya' and jenis_kelamin = 'perempuan'GROUP BY fakultas | Hasil lihat lampiran gambar 30 | Hasil lihat lampiran gambar 30 |
| | select count(*), fakultas ,alamat FROM mahasiswa where alamat= 'surabaya' and jenis_kelamin = 'perempuan'GROUP BY fakultas | | | |
| 31. | cari jumlah mahasiswa berdasarkan fakultas yang alamat bojonegoro dan angkatan 2010 | select count(*), fakultas ,alamat FROM mahasiswa where alamat= 'bojonegoro' and angkatan = '2010'GROUP BY fakultas | Hasil lihat lampiran gambar 31 | Hasil lihat lampiran gambar 31 |
| | select count(*), fakultas ,alamat FROM mahasiswa where alamat= 'bojonegoro' and angkatan = '2010'GROUP BY fakultas | | | |

| | | | | |
|-----|---|---|--|--|
| 32. | berapa total mahasiswa berdasarkan jurusan yang pekerjaan_ortu wiraswasta atau alamat malang | select count(*), jurusan ,pekerjaan_ortu FROM mahasiswa where pekerjaan_ortu= 'wiraswasta' or alamat = 'malang'GROUP BY jurusan | Hasil lihat lampiran gambar 32 | Hasil lihat lampiran gambar 32 |
| | select count(*), jurusan ,pekerjaan_ortu FROM mahasiswa where pekerjaan_ortu= 'wiraswasta' or alamat = 'malang'GROUP BY jurusan | | | |
| 33. | cari total mahasiswa berdasarkan jurusan dimana jenis_kelamin laki-laki atau alamat kediri | select count(*), jurusan ,jenis_kelamin FROM mahasiswa where jenis_kelamin= 'laki-laki' or alamat = 'kediri'GROUP BY jurusan | Hasil lihat lampiran gambar 33 | Hasil lihat lampiran gambar 33 |
| | select count(*), jurusan ,jenis_kelamin FROM mahasiswa where jenis_kelamin= 'laki-laki' or alamat = 'kediri'GROUP BY jurusan | | | |
| 34. | cari total mahasiswa per jurusan yang pekerjaan_ortu tni/polisi dan tahun_lahir 1992 | select count(*), jurusan ,pekerjaan_ortu FROM mahasiswa where pekerjaan_ortu= 'tni/polisi' and tahun_lahir = '1992'GROUP BY jurusan | Hasil lihat lampiran gambar 34 | Hasil lihat lampiran gambar 34 |
| | select count(*), jurusan ,pekerjaan_ortu FROM mahasiswa where pekerjaan_ortu= 'tni/polisi' and tahun_lahir = '1992'GROUP BY jurusan | | | |
| 35. | berapa total mahasiswa yang jurusan fisika | select count(*), jurusan FROM | Hasil lihat | Hasil lihat |

| | | | | |
|-----|---|--|--|--|
| | select count(*), jurusan FROM mahasiswa where jurusan = 'fisika' GROUP BY jurusan | mahasiswa where jurusan = 'fisika' GROUP BY jurusan | lampiran gambar 35 | lampiran gambar 35 |
| 36. | berapa total mahasiswa per fakultas yang jenis_kelamin perempuan dan alamat nganjuk select count(*), fakultas ,jenis_kelamin FROM mahasiswa where jenis_kelamin= 'perempuan' and alamat = 'nganjuk'GROUP BY fakultas | select count(*), fakultas ,jenis_kelamin FROM mahasiswa where jenis_kelamin= 'perempuan' and alamat = 'nganjuk'GROUP BY fakultas | Hasil lihat lampiran gambar 36 | Hasil lihat lampiran gambar 36 |
| 37. | berapa jumlah mahasiswa berdasarkan fakultas yang alamat malang dan pekerjaan_orbu PNS select count(*), fakultas ,alamat FROM mahasiswa where alamat= 'malang' and pekerjaan_orbu = 'PNS'GROUP BY fakultas | select count(*), fakultas ,alamat FROM mahasiswa where alamat= 'malang' and pekerjaan_orbu = 'PNS'GROUP BY fakultas | Hasil lihat lampiran gambar 37 | Hasil lihat lampiran gambar 37 |
| 38. | cari jumlah dosen tiap fakultas yang jabatan_dosennya lektor select count(*), fakultas ,jabatan_dosen FROM dosen where jabatan_dosen= 'lektor' GROUP BY fakultas | select count(*), fakultas ,jabatan_dosen FROM dosen where jabatan_dosen= 'lektor' GROUP BY fakultas | Hasil lihat lampiran gambar 38 | Hasil lihat lampiran gambar 38 |
| 39. | berapa jumlah dosen tiap fakultas dimana golongan_dosennya III/d | select count(*), fakultas ,golongan_dosen FROM dosen where | Hasil lihat lampiran gambar | Hasil lihat lampiran gambar |

| | | | | |
|-----|--|--|--|--|
| | select count(*), fakultas ,golongan_dosen FROM dosen where golongan_dosen= 'III/d' GROUP BY fakultas | golongan_dosen= 'III/d' GROUP BY fakultas | 39 | 39 |
| 40. | tunjukkan semua mahasiswa tiap jurusan dimana angkatan 2011 select count(*), jurusan ,angkatan FROM mahasiswa where angkatan= '2011' GROUP BY jurusan | select count(*), jurusan ,angkatan FROM mahasiswa where angkatan= '2011' GROUP BY jurusan | Hasil lihat lampiran gambar 40 | Hasil lihat lampiran gambar 40 |
| 41. | berapa jumlah mahasiswa menurut fakultas dimana alamatnya kediri dan pekerjaan_ortu PNS select count(*), fakultas ,alamat FROM mahasiswa where alamat= 'kediri' and pekerjaan_ortu = 'PNS'GROUP BY fakultas | select count(*), fakultas ,alamat FROM mahasiswa where alamat= 'kediri' and pekerjaan_ortu = 'PNS'GROUP BY fakultas | Hasil lihat lampiran gambar 41 | Hasil lihat lampiran gambar 41 |
| 42. | berapa seluruh mahasiswa berdasarkan fakultas dimana jenis_kelamin perempuan dan pekerjaan_ortu wiraswasta atau pekerjaan_ortu tani/nelayan select count(*), fakultas ,jenis_kelamin FROM mahasiswa where jenis_kelamin = 'perempuan' and | select count(*), fakultas ,jenis_kelamin FROM mahasiswa where jenis_kelamin = 'perempuan' and pekerjaan_ortu = 'wiraswasta' or pekerjaan_ortu = 'tani/nelayan'GROUP BY fakultas | Hasil lihat lampiran gambar 42 | Hasil lihat lampiran gambar 42 |

| | | | | |
|-----|--|---|--|--|
| | pekerjaan_ortu = 'wiraswasta' or pekerjaan_ortu = 'tani/nelayan'GROUP BY fakultas | | | |
| 43. | cari semua mahasiswa tiap tahun_lahir yang alamatnya jombang atau alamatnya malang | select count(*), tahun_lahir,alamat FROM mahasiswa where alamat= 'jombang' or alamat = 'malang'GROUP BY tahun_lahir | Hasil lihat lampiran gambar 43 | Hasil lihat lampiran gambar 43 |
| | select count(*), tahun_lahir,alamat FROM mahasiswa where alamat= 'jombang' or alamat = 'malang'GROUP BY tahun_lahir | | | |
| 44. | tunjukkan semua mahasiswa tiap angkatan yang jenis_kelamin perempuan atau pekerjaan_ortu wiraswasta | select count(*), angkatan,jenis_kelamin FROM mahasiswa where jenis_kelamin= 'perempuan' or pekerjaan_ortu = 'wiraswasta'GROUP BY angkatan | Hasil lihat lampiran gambar 44 | Hasil lihat lampiran gambar 44 |
| | select count(*), angkatan,jenis_kelamin FROM mahasiswa where jenis_kelamin= 'perempuan' or pekerjaan_ortu = 'wiraswasta'GROUP BY angkatan | | | |
| 45. | berapa semua mahasiswa per angkatan dimana pekerjaan_ortu wiraswasta dan tahun_lahir 1991 | select count(*), angkatan,jenis_kelamin FROM mahasiswa where jenis_kelamin= 'perempuan' and pekerjaan_ortu = 'wiraswasta'GROUP BY angkatan | Hasil lihat lampiran gambar 45 | Hasil lihat lampiran gambar 45 |
| | select count(*), angkatan,jenis_kelamin FROM mahasiswa where jenis_kelamin= 'perempuan' and | | | |

| | | | | |
|-----|--|--|--|--|
| | pekerjaan_ortu = 'wiraswasta'GROUP BY angkatan | | | |
| 46. | berapa total dosen menurut fakultas dimana golongan_dosen III/d dan jabatan_dosennya lektor select count(*), fakultas ,golongan_dosen FROM dosen where golongan_dosen= 'III/d' and jabatan_dosen = 'lektor'GROUP BY fakultas | select count(*), fakultas ,golongan_dosen FROM dosen where golongan_dosen= 'III/d' and jabatan_dosen = 'lektor'GROUP BY fakultas | Hasil lihat lampiran gambar 46 | Hasil lihat lampiran gambar 46 |
| 47. | berapa total dosen berdasarkan fakultas yang golongan_dosen IV/a dan jabatan_dosennya lektor select count(*), fakultas ,golongan_dosen FROM dosen where golongan_dosen= 'IV/a' and jabatan_dosen = 'lektor'GROUP BY fakultas | select count(*), fakultas ,golongan_dosen FROM dosen where golongan_dosen= 'IV/a' and jabatan_dosen = 'lektor'GROUP BY fakultas | Hasil lihat lampiran gambar 47 | Hasil lihat lampiran gambar 47 |
| 48. | sajikan total dosen per fakultas yang golongan_dosen IV/a atau jabatan_dosennya lektor select count(*), fakultas ,golongan_dosen FROM dosen where | select count(*), fakultas ,golongan_dosen FROM dosen where golongan_dosen= 'IV/a' or jabatan_dosen = 'lektor'GROUP BY fakultas | Hasil lihat lampiran gambar 48 | Hasil lihat lampiran gambar 48 |

| | | | | |
|-----|---|---|--|--|
| | golongan_dosen= 'IV/a' or jabatan_dosen = 'lektor'GROUP BY fakultas | | | |
| 49. | cari jumlah mahasiswa tiap fakultas yang alamat Mojokerto dan angkatan 2010 dan tahun_lahir 1991 select count(*), fakultas ,alamat FROM mahasiswa where alamat = 'mojokerto' and angkatan = '2010' and tahun_lahir = '1991'GROUP BY fakultas | select count(*), fakultas ,alamat FROM mahasiswa where alamat = 'mojokerto' and angkatan = '2010' and tahun_lahir = '1991'GROUP BY fakultas | Hasil lihat lampiran gambar 49 | Hasil lihat lampiran gambar 49 |
| 50. | sajikan jumlahnya mahasiswa berdasarkan angkatan dimana jenis_kelamin laki-laki dan pekerjaan_orbu PNS atau pekerjaan_orbu lain- lain select count(*), angkatan,jenis_kelamin FROM mahasiswa where jenis_kelamin = 'laki-laki' and pekerjaan_orbu = 'PNS' or alamat = 'pasuruan'GROUP BY angkatan | select count(*), angkatan,jenis_kelamin FROM mahasiswa where jenis_kelamin = 'laki-laki' and pekerjaan_orbu = 'PNS' or alamat = 'pasuruan'GROUP BY angkatan | Hasil lihat lampiran gambar 50 | Hasil lihat lampiran gambar 50 |

Dari tabel 4.2 didapatkan suatu analisa sebagai berikut:

- 1) Dari 50 sampel uji coba yang dilakukan bernilai benar dikarenakan dari kosakata yang digunakan dapat dikenali oleh sistem dan sesuai dengan data jumlah pada DBMS
- 2) Struktur yang dihasilkan oleh sistem sangat bergantung pada type query yang digunakan. Dari tabel 4.2 menerangkan bahwa penggunaan type query yang digunakan sesuai dengan type query yang terdapat pada sistem.

4.5 Integrasi dengan Al-Quran

Bahasa merupakan media yang paling *essensi* yang digunakan oleh suatu kelompok untuk berinteraksi dan berkomunikasi. Melalui bahasa kita dapat menguasai ilmu pengetahuan (Elviana, 2013). Adanya bahasa sangat membantu untuk memudahkan proses akses informasi secara leluasa karena perbedaan bahasa tidak membatasi seseorang untuk memenuhi kebutuhan informasinya.

Sebagaimana Allah SWT telah berfirman dalam surat Ibrahim ayat 4 :

وَمَا أَرْسَلْنَا مِنْ رَّسُولٍ إِلَّا بِلِسَانٍ قَوْمِهِ لِيُبَيِّنَ لَهُمْ فَيُضِلُّ اللَّهُ مَنْ يَشَاءُ وَيَهْدِي مَنْ يَشَاءُ وَهُوَ الْعَزِيزُ الْحَكِيمُ

Artinya :

“Kami tidak mengutus seorang rasulpun, melainkan dengan bahasa kaumnya[779], supaya ia dapat memberi penjelasan dengan terang kepada mereka. Maka Allah menyesatkan[780] siapa yang Dia kehendaki, dan memberi petunjuk kepada siapa yang Dia kehendaki. dan Dia-lah Tuhan yang Maha Kuasa lagi Maha Bijaksana.”(QS, 14:4)

Menurut kajian tafsir Ibnu Katsir dari ayat tersebut, Hal ini termasuk kasih sayang Allah kepada makhluk-Nya, bahwa Allah mengutus para Rasul dari kalangan mereka dan dengan bahasa mereka supaya mereka memahami apa yang dikehendaki dari mereka. Seperti yang diriwayatkan oleh Imam Ahmad dari Abu Dzarr ra. berkata: Rasulullah saw. bersabda: “Allah tidak mengutus seorang Nabi pun melainkan dengan bahasa kaumnya”. Al Quran diturunkan dalam bahasa Arab itu, bukanlah berarti bahwa Al Qur'an untuk bangsa Arab saja tetapi untuk seluruh manusia. Dari ayat tersebut dapat diambil pelajaran bahwa pentingnya mengembangkan fungsi bahasa termasuk bahasa Indonesia yang digunakan untuk memudahkan akses ke sistem. Dengan begitu informasi akan dengan mudah diakses seseorang tanpa terbatas dengan kemampuan bahasanya sekalipun.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dalam pengerjaan skripsi ini bisa diambil kesimpulan bahwa sistem pengolah bahasa alami yang dibangun dapat mengeluarkan hasil ekstraksi data dalam bentuk grafik. User yang awam terhadap format SQL dapat menginputkan kalimat perintah untuk mengakses data jumlah berupa kalimat bahasa Indonesia, kemudian sistem menerjemahkan kedalam bahasa sesuai format SQL sesuai perintah inputan untuk selanjutnya sistem menyajikan hasilnya berupa grafik. Selain itu komponen *Natural Language Processing* dapat diaplikasikan dalam penerapan sistem pembangkit grafik berbasis bahasa alami pada DBMS.

Dari hasil uji coba yang dilakukan, 50 sampel yang diuji bernilai benar dikarenakan dari kosakata yang digunakan dapat dikenali oleh sistem dan sesuai dengan data jumlah pada DBMS.

5.2 Saran

Sistem yang dihasilkan pada skripsi ini masih terdapat banyak kekurangan yang dapat disempurnakan. Beberapa saran untuk pengembangan sistem ini lebih lanjut adalah :

1. Pengembangan type query dengan mencoba memperluas bentuk-bentuk kalimat yang dapat diterima oleh sistem. Sehingga dapat beraneka kalimat yang dapat digunakan oleh user.

2. Memperbanyak variasi perintah-perintah SQL yang dihasilkan dengan memperluas penerjemahan dari sistem sehingga dapat mencakup semua perintah-perintah SQL.
3. Sistem ini mampu memberi pesan *error* ketika ada ketidak sesuai-an bahasa alami yang diberikan, namun sebaiknya memunculkan kalimat alternatif yang berhubungan dengan kesalahan tersebut.



DAFTAR PUSTAKA

- Adriani, Mirna and Ruli Manurung. 2008. *A survey of bahasa Indonesia NLP research conducted at the University of Indonesia*.
- Andayani,Sri .2006. *Penyusunan Aturan Produksi Bahasa Pengganti Query Untuk Basis Data Matakuliah Program Studi Matematika FMIPA UNY*. Jurusan Pendidikan Matematika FMIPA UNY.
- Bunafit Nugroho dan Indah Indriyana, 2007, “*Membuat Aplikasi Database SQL Server dengan Visual Basic 6.0*”, GavaMedia,Yogyakarta.
- Connolly, Thomas, (2010). *Database systems*, Yogyakarta : penerbit Andi
- Danuri, *Natural Language Processing Untuk Structured Query Language Pada Dbms Mysql* .Teknik Informatika , Politeknik Negeri Bengkalis, Bengkalis.
- Diskominfo. Setiap orang berhak untuk berkomunikasi dan memperoleh informasi.2013.
http://www.magelangkab.go.id/index.php?option=com_content&view=article&id=1099:setiap-orang-berhak-untuk-berkomunikasi-dan-memperoleh-informasi&catid=203:berita-sosial. Diakses tanggal 18 januari 2014 .
- Enikuomihin A.O. dan Okwufulueze D.O.*An Algorithm for Solving Natural Language Query Execution Problems on Relational Databases*.
- Handoko, Agus Purwo. 2009. *Aplikasi Pengolahan Bahasa Alami Untuk Operasi Query Database*. Volume 02. N0 02. Januari 2009
- Indrajani, S.Kom (2004), *Pemrograman Berorientasi Objek dengan Java*, PT. Elex Media Komputindo, Jakarta,
- Jadhav Sneha, Raut Shubhangi, A.S.Zore, 2014. *Natural Language to Database Interface*. International Journal of Advanced Research in Computer and Communication Engineering Vol. 3
- Kadir, Abdul. 2002. *Penuntun Praktis Belajar SQL*. Penerbit Andi. Yogyakarta.
- Kusumadewi, Sri. 2003. *Artificial Intelligence: Teknik dan Aplikasinya*. Graha Ilmu, Yogyakarta.

- Kuspriyanto, 2005. *Perancangan Translator Bahasa Alami Ke Dalam Format Sql (Structured Query Language)*. Departemen Teknik Elektro, Institut Teknologi Bandung, Bandung.
- Mulyadi. 2001. *Akuntansi Manajemen: Konsep, Manfaat dan Rekayasa*. Edisi ketiga. Salemba Empat. Jakarta.
- Nugroho, Adi. 2011. *Perancangan dan Implementasi Sistem basis Data*. Penerbit Andi. Yogyakarta.
- Porfírio P. Filipe. Nuno J. Mamede .2012. *Databases and Natural Language Interfaces*. Lisboa. Portugal.
- Rich, E. dan Knight, K. (1991). *Artificial Intelligence, second edition*. McGraw-Hill, New York
- Ramasubramanian P and Kannan A, *Temporal Event Matching Approach based Natural Language Query Processing in Temporal Databases*, International Journal of Information Technology Vol.10 No.1. India.
- Setyawan, wahyudi. 2009. Model Ontologi perusahaan IT .
<http://wahyudisetiawan.com/2009/08/30/model-ontologi-perusahaan-it/>.
Diakses tanggal 06 Februari 2014.
- Shahih Bukhari (صحيح البخارى), di bawah Kitab Bad'ul Wahyi (كتاب بدء الوحي) (Permulaan Turunnya Wahyu)
- Siswanto .2010. *Kecerdasan tiruan* .Edisi Kedua.Cetakan Pertama.Graha Ilmu. Yogyakarta.
- Suciadi, James. *Studi Analisis Metode -Metode Parsing dan Interpretasi Semantik pada Natural Language Processing*. Journal of Artificial Intelligence Research
- Turban, E. dan Frenzel L.E. (1992). *Expert Systems and Applied Artificial Intelligence, third edition*. Macmillan, New York
- Wibisono, Setyawan. 2013. *Aplikasi Pengolah Bahasa Alami untuk Query Basis data Akademik dengan Format Data Xml*. Universitas Stikubank
- Winston, Patric Henry (1992). *Artificial Intelligence. Second Edition*. Addison-Wesley, Canada

LAMPIRAN

Daftar Kosa kata

| Kata | Jenis_Kata |
|-------------|----------------|
| tampilkan | KP |
| tunjukkan | KP |
| berapa | KP |
| sajikan | KP |
| cari | KP |
| jumlah | Kata_kuantitas |
| seluruh | Kata_kuantitas |
| total | Kata_kuantitas |
| semua | Kata_kuantitas |
| tiap | kata_hub |
| per | kata_hub |
| menurut | kata_hub |
| berdasarkan | kata_hub |
| yang | KS1 |
| dimana | KS1 |
| dan | KS2 |
| atau | KS2 |

Daftar Stoplist

| No | Stoplist | No. | Stoplist |
|----|--------------|-----|--------------|
| 1 | mampu | 64 | mana |
| 2 | tentang | 65 | sementara |
| 3 | di | 66 | mengapa |
| 4 | setelah | 67 | akan |
| 5 | hampir | 68 | dengan |
| 6 | juga | 69 | belum |
| 7 | antara | 70 | anda |
| 8 | ada | 71 | sebuah |
| 9 | seperti | 72 | atas |
| 10 | jadi | 73 | sesuai |
| 11 | sudah | 74 | seberang |
| 12 | oleh | 75 | sebenarnya |
| 13 | bisa | 76 | berdasarkan |
| 14 | tidak | 77 | lagi |
| 15 | sayang | 78 | terhadap |
| 16 | melakukannya | 79 | memungkinkan |
| 17 | lakukan | 80 | sendirian |
| 18 | memang | 81 | bersama |
| 19 | baik | 82 | meskipun |
| 20 | lain | 83 | selalu |
| 21 | pernah | 84 | apapun |
| 22 | setiap | 85 | siapapun |
| 23 | untuk | 86 | saja |
| 24 | mendapatkan | 87 | selain |
| 25 | punya | 88 | muncul |
| 26 | telah | 89 | menghargai |
| 27 | memiliki | 90 | tepat |
| 28 | dia | 91 | sekitar |
| 29 | miliknya | 92 | samping |
| 30 | bagaimana | 93 | bertanya |
| 31 | bagaimanapun | 94 | terkait |
| 32 | jika | 95 | tersedia |
| 33 | ke | 96 | jauh |
| 34 | dalam | 97 | sangat |
| 35 | itu | 98 | menjadi |
| 36 | sama | 99 | sebab |
| 37 | paling | 100 | sebelum |

| | | | |
|----|-----------|-----|------------------|
| 38 | biarkan | 101 | sebelumnya |
| 39 | mungkin | 102 | belakang |
| 40 | aku | 103 | percaya |
| 41 | sebagian | 104 | bawah |
| 42 | besar | 105 | terbaik |
| 43 | harus | 106 | lebih |
| 44 | saya | 107 | luar |
| 45 | sering | 108 | keduanya |
| 46 | pada | 109 | singkat |
| 47 | hanya | 110 | namun |
| 48 | kita | 111 | ayo |
| 49 | sendiri | 112 | saat |
| 50 | agak | 113 | pasti |
| 51 | kata | 114 | dijelaskan |
| 52 | begitu | 115 | berbeda |
| 53 | beberapa | 116 | melakukan |
| 54 | mereka | 117 | dilakukan |
| 55 | kemudian | 118 | selama |
| 56 | sana | 119 | akibatnya |
| 57 | ini | 120 | mempertimbangkan |
| 58 | sungguh | 121 | berisi |
| 59 | keinginan | 122 | mengandung |
| 60 | adalah | 123 | halo |
| 61 | kami | 124 | membantu |
| 62 | apa | 125 | maka |
| 63 | kapan | 126 | tentu |

Printscreen Hasil Uji Coba

Gambar 1

| fakultas | count(*) |
|----------|----------|
| saintek | 598 |

| count(*) | fakultas |
|----------|----------|
| 598 | saintek |

Gambar 2

| alamat | jenis_kelamin | count(*) |
|----------|---------------|----------|
| PASURUAN | laki-laki | 3 |
| PASURUAN | perempuan | 9 |

| count(*) | jenis_kelamin | alamat |
|----------|---------------|----------|
| 3 | laki-laki | PASURUAN |
| 9 | perempuan | PASURUAN |

Gambar 3

| alamat | count(*) |
|----------|----------|
| PASURUAN | 100 |

| count(*) | alamat |
|----------|----------|
| 100 | PASURUAN |

Gambar 4

| jabatan_dosen | count(*) |
|---------------|----------|
| Lektor | 112 |

| count(*) | jabatan_dosen |
|----------|---------------|
| 112 | Lektor |

Gambar 5

| tahun_lahir | count(*) |
|-------------|----------|
| 1990 | 849 |

| count(*) | tahun_lahir |
|----------|-------------|
| 849 | 1990 |

Gambar 6

| alamat | jurusan | count(*) |
|--------|--------------------|----------|
| malang | akuntansi | 4 |
| MALANG | fisika | 6 |
| Malang | kimia | 1 |
| Malang | perbankan | 1 |
| malang | teknik informatika | 4 |

| count(*) | jurusan | alamat |
|----------|--------------------|--------|
| 4 | akuntansi | malang |
| 6 | fisika | MALANG |
| 1 | kimia | Malang |
| 1 | perbankan | Malang |
| 4 | teknik informatika | malang |

Gambar 7

| angkatan | count(*) |
|----------|----------|
| 2010 | 638 |

| count(*) | angkatan |
|----------|----------|
| 638 | 2010 |

Gambar 8

| angkatan | jurusan | count(*) |
|----------|--------------------|----------|
| 2011 | akuntansi | 5 |
| 2011 | AS | 11 |
| 2010 | biologi | 8 |
| 2010 | fisika | 6 |
| 2011 | HBS | 7 |
| 2010 | kimia | 12 |
| 2011 | manajemen | 4 |
| 2010 | matematika | 2 |
| 2011 | PAI | 13 |
| 2011 | PBA | 3 |
| 2011 | pendidikan IPS | 2 |
| 2010 | perbankan | 15 |
| 2011 | PGMI | 4 |
| 2011 | psikologi | 24 |
| 2011 | teknik arsitek | 9 |
| 2010 | teknik informatika | 7 |

| count(*) | jurusan | angkatan |
|----------|--------------------|----------|
| 5 | akuntansi | 2011 |
| 11 | AS | 2011 |
| 8 | biologi | 2010 |
| 6 | fisika | 2010 |
| 7 | HBS | 2011 |
| 12 | kimia | 2010 |
| 4 | manajemen | 2011 |
| 2 | matematika | 2010 |
| 13 | PAI | 2011 |
| 3 | PBA | 2011 |
| 2 | pendidikan IPS | 2011 |
| 15 | perbankan | 2010 |
| 4 | PGMI | 2011 |
| 24 | psikologi | 2011 |
| 9 | teknik arsitek | 2011 |
| 7 | teknik informatika | 2010 |

Gambar 9

| fakultas | count(*) |
|----------|----------|
| tarbiyah | 338 |

| count(*) | fakultas |
|----------|----------|
| 338 | tarbiyah |

Gambar 10

| pekerjaan_orstu | count(*) |
|-----------------|----------|
| PNS | 312 |

| count(*) | pekerjaan_orstu |
|----------|-----------------|
| 312 | PNS |

Gambar 11

| fakultas | count(*) |
|-----------|----------|
| psikologi | 311 |

| count(*) | fakultas |
|----------|-----------|
| 311 | psikologi |

Gambar 12

| fakultas | count(*) |
|-----------|----------|
| ekonomi | 388 |
| psikologi | 311 |
| saintek | 598 |
| syariah | 194 |
| tarbiyah | 338 |

| count(*) | fakultas |
|----------|-----------|
| 388 | ekonomi |
| 311 | psikologi |
| 598 | saintek |
| 194 | syariah |
| 338 | tarbiyah |

Gambar 13

| tahun_lahir | count(*) |
|-------------|----------|
| 1992 | 553 |

| count(*) | tahun_lahir |
|----------|-------------|
| 553 | 1992 |

Gambar 14

| golongan_dosen | count(*) |
|----------------|----------|
| III/c | 70 |

| count(*) | golongan_dosen |
|----------|----------------|
| 70 | III/c |

Gambar 15

| jabatan_dosen | count(*) |
|---------------|----------|
| Lektor | 112 |

| count(*) | jabatan_dosen |
|----------|---------------|
| 112 | Lektor |

Gambar 16

| golongan_dosen | count(*) |
|----------------|----------|
| III/c | 70 |

| count(*) | golongan_dosen |
|----------|----------------|
| 70 | III/c |

Gambar 17

| angkatan | jurusan | count(*) |
|----------|------------|----------|
| 2011 | akuntansi | 23 |
| 2011 | AS | 5 |
| 2010 | biologi | 7 |
| 2010 | fisika | 24 |
| 2011 | HBS | 3 |
| 2010 | kimia | 22 |
| 2011 | manajemen | 1 |
| 2010 | matematika | 13 |
| 2011 | PAI | 3 |
| 2011 | PBA | 1 |
| 2010 | perbankan | 19 |
| 2011 | PGMI | 1 |
| 2011 | psikologi | 1 |

| count(*) | jurusan | angkatan |
|----------|--------------------|----------|
| 23 | akuntansi | 2011 |
| 5 | AS | 2011 |
| 7 | biologi | 2010 |
| 24 | fisika | 2010 |
| 3 | HBS | 2011 |
| 22 | kimia | 2010 |
| 1 | manajemen | 2011 |
| 13 | matematika | 2010 |
| 3 | PAI | 2011 |
| 1 | PBA | 2011 |
| 19 | perbankan | 2010 |
| 1 | PGMI | 2011 |
| 1 | psikologi | 2011 |
| 19 | teknik informatika | 2010 |

Gambar 18

| angkatan | fakultas | count(*) |
|----------|-----------|----------|
| 2011 | ekonomi | 137 |
| 2011 | psikologi | 311 |
| 2011 | saintek | 211 |
| 2011 | syariah | 194 |
| 2011 | tarbiyah | 338 |

| count(*) | fakultas | angkatan |
|----------|-----------|----------|
| 137 | ekonomi | 2011 |
| 311 | psikologi | 2011 |
| 211 | saintek | 2011 |
| 194 | syariah | 2011 |
| 338 | tarbiyah | 2011 |

Gambar 19

| tahun_lahir | jurusan | count(*) |
|-------------|--------------------|----------|
| 1992 | akuntansi | 45 |
| 1992 | AS | 33 |
| 1992 | biologi | 21 |
| 1992 | fisika | 31 |
| 1992 | HBS | 27 |
| 1992 | kimia | 35 |
| 1992 | manajemen | 23 |
| 1992 | matematika | 19 |
| 1992 | PAI | 58 |
| 1992 | PBA | 13 |
| 1992 | pendidikan IPS | 10 |
| 1992 | perbankan | 45 |
| 1992 | PGMI | 20 |
| 1992 | psikologi | 100 |
| 1992 | teknik arsitek | 36 |
| 1992 | teknik informatika | 37 |

| count(*) | jurusan | tahun_lahir |
|----------|--------------------|-------------|
| 45 | akuntansi | 1992 |
| 33 | AS | 1992 |
| 21 | biologi | 1992 |
| 31 | fisika | 1992 |
| 27 | HBS | 1992 |
| 35 | kimia | 1992 |
| 23 | manajemen | 1992 |
| 19 | matematika | 1992 |
| 58 | PAI | 1992 |
| 13 | PBA | 1992 |
| 10 | pendidikan IPS | 1992 |
| 45 | perbankan | 1992 |
| 20 | PGMI | 1992 |
| 100 | psikologi | 1992 |
| 36 | teknik arsitek | 1992 |
| 37 | teknik informatika | 1992 |

Gambar 20

| tahun_lahir | alamat | count(*) |
|-------------|----------------|----------|
| 1992 | BANDAR LAMPUNG | 1 |
| 1992 | BANGKALAN | 2 |
| 1992 | BANYUWANGI | 4 |
| 1992 | BATU | 2 |
| 1992 | BLITAR | 15 |
| 1992 | BLORA | 1 |
| 1992 | BOGOR | 1 |
| 1992 | BOJONEGORO | 1 |
| 1992 | BONDOWOSO | 1 |
| 1992 | BUNGO | 1 |
| 1992 | GORONTALO | 1 |
| 1992 | GRESIK | 6 |
| 1992 | JEMBER | 6 |

| count(*) | alamat | tahun_lahir |
|----------|----------------|-------------|
| 45 | BLORA | 1992 |
| 33 | Mojokerto | 1992 |
| 21 | JOMBANG | 1992 |
| 31 | Pati | 1992 |
| 27 | PAMEKASAN | 1992 |
| 35 | PASURUAN | 1992 |
| 23 | MALANG | 1992 |
| 19 | PALEMBANG | 1992 |
| 58 | PASURUAN | 1992 |
| 13 | LAMONGAN | 1992 |
| 10 | kediri | 1992 |
| 45 | MADIUN | 1992 |
| 20 | Pasuruan | 1992 |
| 100 | NGANJUK | 1992 |
| 36 | GRESIK | 1992 |
| 37 | BANDAR LAMPUNG | 1992 |

Gambar 21

| angkatan | jurusan | count(*) |
|----------|--------------------|----------|
| 2011 | akuntansi | 42 |
| 2011 | AS | 113 |
| 2011 | biologi | 54 |
| 2011 | HBS | 81 |
| 2011 | manajemen | 85 |
| 2011 | PAI | 198 |
| 2011 | PBA | 35 |
| 2011 | pendidikan IPS | 36 |
| 2011 | perbankan | 10 |
| 2011 | PGMI | 69 |
| 2011 | psikologi | 311 |
| 2011 | teknik arsitek | 123 |
| 2011 | teknik informatika | 34 |

| count(*) | jurusan | angkatan |
|----------|--------------------|----------|
| 42 | akuntansi | 2011 |
| 113 | AS | 2011 |
| 54 | biologi | 2011 |
| 81 | HBS | 2011 |
| 85 | manajemen | 2011 |
| 198 | PAI | 2011 |
| 35 | PBA | 2011 |
| 36 | pendidikan IPS | 2011 |
| 10 | perbankan | 2011 |
| 69 | PGMI | 2011 |
| 311 | psikologi | 2011 |
| 123 | teknik arsitek | 2011 |
| 34 | teknik informatika | 2011 |

Gambar 22

| jenis_kelamin | fakultas | count(*) | count(*) | fakultas | jenis_kelamin |
|---------------|-----------|----------|----------|-----------|---------------|
| perempuan | ekonomi | 255 | 255 | ekonomi | perempuan |
| perempuan | psikologi | 142 | 142 | psikologi | perempuan |
| perempuan | saintek | 338 | 338 | saintek | perempuan |
| perempuan | syariah | 131 | 131 | syariah | perempuan |
| perempuan | tarbiyah | 200 | 200 | tarbiyah | perempuan |

Gambar 23

| jenis_kelamin | fakultas | count(*) | count(*) | fakultas | jenis_kelamin |
|---------------|-----------|----------|----------|-----------|---------------|
| laki-laki | ekonomi | 133 | 133 | ekonomi | laki-laki |
| laki-laki | psikologi | 169 | 169 | psikologi | laki-laki |
| laki-laki | saintek | 260 | 260 | saintek | laki-laki |
| laki-laki | syariah | 63 | 63 | syariah | laki-laki |
| laki-laki | tarbiyah | 138 | 138 | tarbiyah | laki-laki |

Gambar 24

| fakultas | jurusan | count(*) | count(*) | jurusan | fakultas |
|----------|---------|----------|----------|---------|----------|
| syariah | AS | 83 | 83 | AS | syariah |
| syariah | HBS | 48 | 48 | HBS | syariah |

Gambar 25

| alamat | angkatan | count(*) | count(*) | angkatan | alamat |
|--------|----------|----------|----------|----------|--------|
| malang | 2010 | 80 | 80 | 2010 | malang |
| Malang | 2011 | 117 | 117 | 2011 | Malang |

Gambar 26

| pekerjaan_ortu | jurusan | count(*) | count(*) | jurusan | pekerjaan_ortu |
|----------------|--------------------|----------|----------|--------------------|----------------|
| WIRASWASTA | akuntansi | 62 | 62 | akuntansi | WIRASWASTA |
| WIRASWASTA | AS | 48 | 48 | AS | WIRASWASTA |
| WIRASWASTA | biologi | 34 | 34 | biologi | WIRASWASTA |
| WIRASWASTA | fisika | 45 | 45 | fisika | WIRASWASTA |
| WIRASWASTA | HBS | 36 | 36 | HBS | WIRASWASTA |
| WIRASWASTA | kimia | 50 | 50 | kimia | WIRASWASTA |
| WIRASWASTA | manajemen | 40 | 40 | manajemen | WIRASWASTA |
| WIRASWASTA | matematika | 29 | 29 | matematika | WIRASWASTA |
| WIRASWASTA | PAI | 76 | 76 | PAI | WIRASWASTA |
| WIRASWASTA | PBA | 18 | 18 | PBA | WIRASWASTA |
| WIRASWASTA | pendidikan IPS | 19 | 19 | pendidikan IPS | WIRASWASTA |
| WIRASWASTA | perbankan | 62 | 62 | perbankan | WIRASWASTA |
| WIRASWASTA | PGMI | 26 | 26 | PGMI | WIRASWASTA |
| WIRASWASTA | psikologi | 133 | 133 | psikologi | WIRASWASTA |
| WIRASWASTA | teknik arsitek | 54 | 54 | teknik arsitek | WIRASWASTA |
| WIRASWASTA | teknik informatika | 51 | 51 | teknik informatika | WIRASWASTA |

Gambar 27

| alamat | tahun_lahir | count(*) | count(*) | tahun_lahir | alamat |
|------------|-------------|----------|----------|-------------|------------|
| Banyuwangi | 1990 | 21 | 21 | 1990 | Banyuwangi |
| banyuwangi | 1991 | 4 | 4 | 1991 | banyuwangi |
| banyuwangi | 1992 | 14 | 14 | 1992 | banyuwangi |

Gambar 28

| fakultas | angkatan | count(*) | count(*) | angkatan | fakultas |
|----------|----------|----------|----------|----------|----------|
| ekonomi | 2010 | 251 | 251 | 2010 | ekonomi |
| ekonomi | 2011 | 137 | 137 | 2011 | ekonomi |

Gambar 29

| angkatan | fakultas | count(*) | count(*) | fakultas | angkatan |
|----------|----------|----------|----------|----------|----------|
| 2010 | ekonomi | 27 | 27 | ekonomi | 2010 |
| 2010 | saintek | 53 | 53 | saintek | 2010 |

Gambar 30

| alamat | fakultas | count(*) | count(*) | fakultas | alamat |
|----------|----------|----------|----------|----------|----------|
| SURABAYA | ekonomi | 4 | 4 | ekonomi | SURABAYA |
| SURABAYA | saintek | 3 | 3 | saintek | SURABAYA |
| SURABAYA | syariah | 1 | 1 | syariah | SURABAYA |
| SURABAYA | tarbiyah | 1 | 1 | tarbiyah | SURABAYA |

Gambar 31

| alamat | fakultas | count(*) | count(*) | fakultas | alamat |
|------------|----------|----------|----------|----------|------------|
| Bojonegoro | ekonomi | 4 | 4 | ekonomi | Bojonegoro |
| BOJONEGORO | saintek | 8 | 8 | saintek | BOJONEGORO |

Gambar 32

| pekerjaan_orstu | jurusan | count(*) | count(*) | jurusan | pekerjaan_orstu |
|-----------------|--------------------|----------|----------|--------------------|-----------------|
| WIRASWASTA | akuntansi | 70 | 70 | akuntansi | WIRASWASTA |
| WIRASWASTA | AS | 53 | 53 | AS | WIRASWASTA |
| TANI/NELAYAN | biologi | 39 | 39 | biologi | TANI/NELAYAN |
| PNS | fisika | 53 | 53 | fisika | PNS |
| WIRASWASTA | HBS | 38 | 38 | HBS | WIRASWASTA |
| WIRASWASTA | kimia | 59 | 59 | kimia | WIRASWASTA |
| WIRASWASTA | manajemen | 47 | 47 | manajemen | WIRASWASTA |
| WIRASWASTA | matematika | 32 | 32 | matematika | WIRASWASTA |
| WIRASWASTA | PAI | 94 | 94 | PAI | WIRASWASTA |
| WIRASWASTA | PBA | 20 | 20 | PBA | WIRASWASTA |
| WIRASWASTA | pendidikan IPS | 20 | 20 | pendidikan IPS | WIRASWASTA |
| WIRASWASTA | perbankan | 73 | 73 | perbankan | WIRASWASTA |
| WIRASWASTA | PGMI | 32 | 32 | PGMI | WIRASWASTA |
| WIRASWASTA | psikologi | 144 | 144 | psikologi | WIRASWASTA |
| WIRASWASTA | teknik arsitek | 61 | 61 | teknik arsitek | WIRASWASTA |
| PEG.SWASTA | teknik informatika | 69 | 69 | teknik informatika | PEG.SWASTA |

Gambar 33

| jenis_kelamin | jurusan | count(*) |
|---------------|----------------|----------|
| laki-laki | akuntansi | 64 |
| laki-laki | AS | 33 |
| laki-laki | biologi | 35 |
| laki-laki | fisika | 45 |
| laki-laki | HBS | 35 |
| laki-laki | kimia | 45 |
| laki-laki | manajemen | 41 |
| laki-laki | matematika | 32 |
| laki-laki | PAI | 99 |
| laki-laki | PBA | 13 |
| perempuan | pendidikan IPS | 12 |
| perempuan | perbankan | 33 |
| laki-laki | PGMI | 23 |

| count(*) | jurusan | jenis_kelamin |
|----------|--------------------|---------------|
| 64 | akuntansi | laki-laki |
| 33 | AS | laki-laki |
| 35 | biologi | laki-laki |
| 45 | fisika | laki-laki |
| 35 | HBS | laki-laki |
| 45 | kimia | laki-laki |
| 41 | manajemen | laki-laki |
| 32 | matematika | laki-laki |
| 99 | PAI | laki-laki |
| 13 | PBA | laki-laki |
| 12 | pendidikan IPS | perempuan |
| 33 | perbankan | perempuan |
| 23 | PGMI | laki-laki |
| 176 | psikologi | laki-laki |
| 78 | teknik arsitek | laki-laki |
| 36 | teknik informatika | laki-laki |

Gambar 34

| pekerjaan_oru | jurusan | count(*) |
|---------------|--------------------|----------|
| TNI/POLISI | AS | 2 |
| TNI/POLISI | HBS | 2 |
| TNI/POLISI | PBA | 1 |
| TNI/POLISI | teknik informatika | 1 |

| count(*) | jurusan | pekerjaan_oru |
|----------|--------------------|---------------|
| 2 | AS | TNI/POLISI |
| 2 | HBS | TNI/POLISI |
| 1 | PBA | TNI/POLISI |
| 1 | teknik informatika | TNI/POLISI |

Gambar 35

| jurusan | count(*) |
|---------|----------|
| fisika | 97 |

| count(*) | jurusan |
|----------|---------|
| 97 | fisika |

Gambar 36

| jenis_kelamin | fakultas | count(*) |
|---------------|-----------|----------|
| perempuan | ekonomi | 6 |
| perempuan | psikologi | 2 |
| perempuan | saintek | 5 |
| perempuan | syariah | 1 |
| perempuan | tarbiyah | 3 |

| count(*) | fakultas | jenis_kelamin |
|----------|-----------|---------------|
| 6 | ekonomi | perempuan |
| 2 | psikologi | perempuan |
| 5 | saintek | perempuan |
| 1 | syariah | perempuan |
| 3 | tarbiyah | perempuan |

Gambar 37

| alamat | fakultas | count(*) |
|--------|-----------|----------|
| malang | ekonomi | 8 |
| Malang | psikologi | 6 |
| Malang | saintek | 15 |
| Malang | syariah | 2 |
| MALANG | tarbiyah | 7 |

| count(*) | fakultas | alamat |
|----------|-----------|--------|
| 8 | ekonomi | malang |
| 6 | psikologi | Malang |
| 15 | saintek | Malang |
| 2 | syariah | Malang |
| 7 | tarbiyah | MALANG |

Gambar 38

| jabatan_dosen | fakultas | count(*) |
|---------------|-----------|----------|
| Lektor | ekonomi | 8 |
| Lektor | humbud | 19 |
| Lektor | psikologi | 7 |
| Lektor | saintek | 40 |
| Lektor | syariah | 10 |
| Lektor | tarbiyah | 28 |

| count(*) | fakultas | jabatan_dosen |
|----------|-----------|---------------|
| 8 | ekonomi | Lektor |
| 19 | humbud | Lektor |
| 7 | psikologi | Lektor |
| 40 | saintek | Lektor |
| 10 | syariah | Lektor |
| 28 | tarbiyah | Lektor |

Gambar 39

| golongan_dosen | fakultas | count(*) |
|----------------|-----------|----------|
| III/d | ekonomi | 14 |
| III/d | humbud | 19 |
| III/d | psikologi | 12 |
| III/d | saintek | 34 |
| III/d | syariah | 11 |
| III/d | tarbiyah | 29 |

| count(*) | fakultas | golongan_dosen |
|----------|-----------|----------------|
| 14 | ekonomi | III/d |
| 19 | humbud | III/d |
| 12 | psikologi | III/d |
| 34 | saintek | III/d |
| 11 | syariah | III/d |
| 29 | tarbiyah | III/d |

Gambar 40

| angkatan | jurusan | count(*) |
|----------|--------------------|----------|
| 2011 | akuntansi | 42 |
| 2011 | AS | 113 |
| 2011 | biologi | 54 |
| 2011 | HBS | 81 |
| 2011 | manajemen | 85 |
| 2011 | PAI | 198 |
| 2011 | PBA | 35 |
| 2011 | pendidikan IPS | 36 |
| 2011 | perbankan | 10 |
| 2011 | PGMI | 69 |
| 2011 | psikologi | 311 |
| 2011 | teknik arsitek | 123 |
| 2011 | teknik informatika | 34 |

| count(*) | jurusan | angkatan |
|----------|--------------------|----------|
| 42 | akuntansi | 2011 |
| 113 | AS | 2011 |
| 54 | biologi | 2011 |
| 81 | HBS | 2011 |
| 85 | manajemen | 2011 |
| 198 | PAI | 2011 |
| 35 | PBA | 2011 |
| 36 | pendidikan IPS | 2011 |
| 10 | perbankan | 2011 |
| 69 | PGMI | 2011 |
| 311 | psikologi | 2011 |
| 123 | teknik arsitek | 2011 |
| 34 | teknik informatika | 2011 |

Gambar 41

| alamat | fakultas | count(*) |
|--------|-----------|----------|
| KEDIRI | ekonomi | 3 |
| KEDIRI | psikologi | 2 |
| KEDIRI | saintek | 3 |
| KEDIRI | syariah | 2 |
| Kediri | tarbiyah | 2 |

| count(*) | fakultas | alamat |
|----------|-----------|--------|
| 3 | ekonomi | KEDIRI |
| 2 | psikologi | KEDIRI |
| 3 | saintek | KEDIRI |
| 2 | syariah | KEDIRI |
| 2 | tarbiyah | Kediri |

Gambar 42

| jenis_kelamin | fakultas | count(*) |
|---------------|-----------|----------|
| perempuan | ekonomi | 213 |
| perempuan | psikologi | 134 |
| laki-laki | saintek | 264 |
| perempuan | syariah | 95 |
| perempuan | tarbiyah | 151 |

| count(*) | fakultas | jenis_kelamin |
|----------|-----------|---------------|
| 213 | ekonomi | perempuan |
| 134 | psikologi | perempuan |
| 264 | saintek | laki-laki |
| 95 | syariah | perempuan |
| 151 | tarbiyah | perempuan |

Gambar 43

| alamat | tahun_lahir | count(*) | count(*) | tahun_lahir | alamat |
|---------|-------------|----------|----------|-------------|---------|
| Malang | 1989 | 8 | 8 | 1989 | Malang |
| malang | 1990 | 113 | 113 | 1990 | malang |
| JOMBANG | 1991 | 53 | 53 | 1991 | JOMBANG |
| Malang | 1992 | 70 | 70 | 1992 | Malang |

Gambar 44

| jenis_kelamin | angkatan | count(*) | count(*) | angkatan | jenis_kelamin |
|---------------|----------|----------|----------|----------|---------------|
| perempuan | 2010 | 501 | 501 | 2010 | perempuan |
| laki-laki | 2011 | 863 | 863 | 2011 | laki-laki |

Gambar 45

| pekerjaan_orbu | angkatan | count(*) | count(*) | angkatan | jenis_kelamin |
|----------------|----------|----------|----------|----------|---------------|
| WIRASWASTA | 2010 | 49 | 191 | 2010 | perempuan |
| WIRASWASTA | 2011 | 92 | 294 | 2011 | perempuan |

Gambar 46

| golongan_dosen | fakultas | count(*) | count(*) | fakultas | golongan_dosen |
|----------------|-----------|----------|----------|-----------|----------------|
| III/d | ekonomi | 5 | 5 | ekonomi | III/d |
| III/d | humbud | 12 | 12 | humbud | III/d |
| III/d | psikologi | 5 | 5 | psikologi | III/d |
| III/d | saintek | 28 | 28 | saintek | III/d |
| III/d | syariah | 3 | 3 | syariah | III/d |
| III/d | tarbiyah | 18 | 18 | tarbiyah | III/d |

Gambar 47

| golongan_dosen | fakultas | count(*) | count(*) | fakultas | golongan_dosen |
|----------------|----------|----------|----------|----------|----------------|
| IV/a | tarbiyah | 2 | 2 | tarbiyah | IV/a |

Gambar 48

| golongan_dosen | fakultas | count(*) | count(*) | fakultas | golongan_dosen |
|----------------|-----------|----------|----------|-----------|----------------|
| IV/a | ekonomi | 14 | 14 | ekonomi | IV/a |
| III/c | humbud | 26 | 26 | humbud | III/c |
| IV/a | psikologi | 10 | 10 | psikologi | IV/a |
| IV/a | saintek | 46 | 46 | saintek | IV/a |
| IV/a | syariah | 15 | 15 | syariah | IV/a |
| IV/a | tarbiyah | 44 | 44 | tarbiyah | IV/a |

Gambar 49

| alamat | fakultas | count(*) | count(*) | fakultas | alamat |
|-----------|----------|----------|----------|----------|-----------|
| Mojokerto | saintek | 4 | 4 | saintek | Mojokerto |

Gambar 50

| jenis_kelamin | angkatan | count(*) | count(*) | angkatan | jenis_kelamin |
|---------------|----------|----------|----------|----------|---------------|
| laki-laki | 2010 | 44 | 81 | 2010 | laki-laki |
| laki-laki | 2011 | 99 | 155 | 2011 | laki-laki |