19.476/ITS/H/2004

MILIK PERPUSTAKAAN
INSTITUT TEKNOLOGI
SEPULUH - NOPEMBER

PERANCANGAN DAN PEMBUATAN PERANGKAT LUNAK KAMUS BAHASA ARAB INDONESIA

TUGAS AKHIR

R51f
005.1
Hid
p-2
2000

PERPUSTAKAAN
ITS

Tgl. Terima	15-7-2003
Terima Dari	H
No. Agenda Prp.	218240

Oleh :

HERLAMBANG WAHYU HIDAYAT
2694 100 053

JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2000

PERANCANGAN DAN PEMBUATAN PERANGKAT LUNAK KAMUS BAHASA ARAB INDONESIA

TUGAS AKHIR

Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik Informatika
Jurusan Teknik Informatika
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya

Mengetahui / Menyetujui,

Dosen Pembimbing I	Dosen Pembimbing II
Dr. Ir. ARIF DJUNAIDY, M.Sc	**Ir. ARIS TJAHYANTO, M.Kom**
NIP. 131 633 403	NIP. 131 933 299

SURABAYA
Pebruari, 2000

بسم الله الرحمن الرحيم

ABSTRAK

ABSTRAK

Dalam mempelajari suatu bahasa, diperlukan adanya suatu kamus lengkap. Yang bisa mengartikan semua jenis kata, baik kata asli maupun kata turunan dari kata asli. Dalam Bahasa Arab, satu bentuk kata asli bisa diturunkan menjadi lebih dari seratus kata, sehingga untuk menyimpan keseluruhan kata-kata turunan ke dalam basis data sangat sulit untuk dilakukan, karena akan memerlukan basis data yang besar, yang pada gilirannya akan menyebabkan lambatnya proses pelacakan kata. Selain itu, oleh karena urutan kata dalam kamus bahasa yang ada selalu diurutkan berdasarkan urutan abjad dari bentuk dasarnya, maka untuk mengetahui arti kata yang diinginkan harus terlebih dahulu mencari bentuk dasar dari kata tersebut.

Dalam Tugas Akhir ini, dibuat suatu perangkat lunak Kamus Bahasa Arab yang mempunyai kemampuan untuk memberikan makna suatu kata turunan berdasarkan bentuk dasarnya. Untuk tujuan ini, maka basis data yang digunakan hanyalah berupa bentuk dasarnya saja. Proses pencarian bentuk dasar dilakukan dengan menggunakan sistem berbasis pengetahuan, dengan cara mencocokkan kata masukan dengan aturan-aturan yang ada pada basis pengetahuan. Setelah bentuk dasar ditemukan barulah kemudian kata tersebut dicari dalam basis data, kemudian selanjutnya makna kata tersebut ditampilkan sesuai dengan makna dari bentuk dasar dan makna dari bentuk turunan katanya.

Dengan adanya perangkat lunak Kamus Bahasa Arab Indonesia ini akan lebih memudahkan pencarian suatu makna kata dalam Bahasa Arab. Kemungkinan pengembangan dari perangkat lunak ini adalah dengan menambahkan proses untuk mengecek grammar suatu kalimat, sehingga bisa menjadi program penterjemah dari bahasa Arab ke bahasa Indonesia.

KATA PENGANTAR

KATA PENGANTAR

Alhamdulillahi Robbil Allamiin, segala puji dan syukur hanyalah ke hadirat Allah SWT, yang telah melimpahkan rahmat, hidayah, dan inayah-Nya sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul :

PERANCANGAN DAN PEMBUATAN PERANGKAT LUNAK KAMUS BAHASA ARAB INDONESIA

Ini dengan baik. Dan semoga shlawat serta salam senantiasa tercurah kepada junjungan kita Rasulullah Muhammad SAW beserta keluarganya yang suci dan disucikan, kepada sahabatnya dan kepada orang-orang yang teguh memegang dan memperjuangkan risalah beliau hingga akhir jaman. Aamiin.

Pada kesempatan ini penulis ingin menyampaikan rasa terima kasih tulus dan penghargaan yang sebesar-besarnya kepada semua pihak yang telah membantu baik secara teknis maupun non-teknis, khususnya kepada :

- Bapak DR. Ir. Arif Djunaidy, MSc, selaku Ketua Jurusan Teknik Informatika ITS sekaligus sebagai Dosen Pembimbing I yang telah banyak meluangkan waktu untuk membimbing, mengarahkan, serta memberikan motivasi selama masa pengerjaan Tugas Akhir ini.
- Bapak Ir. Aris Tjahyanto, MKom, selaku dosen wali sekaligus sebagai Dosen Pembimbing II yang senantiasa memberikan ide, motivasi, dan bimbingan selama masa pengerjaan Tugas Akhir ini.
- Mas Agus Zainal, SKom, atas diskusi dan segala bantuan yang diberikan selama pengerjaan Tugas Akhir ini.
- Segenap Dosen Jurusan Teknik Informatika atas curahan ilmu yang telah diberikan selama penulis menjadi mahasiswa.

MILIK PERPUSTAKAAN
INSTITUT TEKNOLOGI
SEPULUH - NOPEMBER

- Staf tata usaha dan karyawan Jurusan Teknik Informatika yang telah membantu kelancaran studi penulis.
- Ayah (alm) dan Ibu tercinta serta kakak-kakakku yang telah memberikan bimbingan, nasehat, dan motivasi serta do'a kepada penulis.
- Ustadz Achmad Rusydi Al-Idrus sebagai pembimbing spiritual sekaligus yang telah mengajarkan pelajaran dasar Bahasa Arab.
- Mbak Umi, Mbak Lilik juga Mbak Entin atas diskusi, motivasi dan do'anya sehingga penulis dapat menyelesaikan tugas akhir ini.
- Teman-temanku yang telah membantu kelancaran penyelesaian tugas akhir ini terutama kepada : Juhar, Arif, Asyhad, Ali, Bidin, Anda, Adik, Waskito, Seger, Dudi, Salman, Tom, Iduy, juga rekan-rekan C0A dan seluruh rekan TC lainnya.
- Dan semua pihak yang tidak bisa penulis sebutkan satu persatu, yang telah banyak membantu terselesaikannya Tugas Akhir ini.

Semoga Allah SWT membalas semua kebaikan dengan balasan yang sebaik-baiknya, karena sesunguhnya Allah SWT adalah sebaik-baik pemberi pembalasan.

Penulis menyadari, masih terdapat banyak keterbatasan dan kekurangan dalam Tugas Akhir ini, oleh karena itu kritik dan saran dari pembaca sangat diharapkan untuk pengembangan ide yang telah dibangun dalam Tugas Akhir ini.

Akhir kata, semoga karya sederhana ini bermanfaat bagi para pembaca dan Allah SWT akan mencatatnya sebagai bagian dari ilmu yang bermanfaat. *Allahumma Aamiin.*

Surabaya, 10 Pebruari 2000

Herlambang Wahyu Hidayat

DAFTAR ISI

DAFTAR ISI

LEMBAR JUDUL
LEMBAR PENGESAHAN
ABSTRAK i
KATA PENGANTAR ii
DAFTAR ISI iv
DAFTAR GAMBAR vi
DAFTAR TABEL vii

BAB I PENDAHULUAN 1
1.1 Latar Belakang 1
1.2 Tujuan dan Manfaat 2
1.3 Permasalahan 2
1.4 Batasan Masalah 3
1.5 Metodologi 3
1.6 Sistematika Pembahasan 4

BAB II SEKILAS TENTANG ISIM, FI'IL DAN HURUF 6
2.1 Isim (kata benda) 6
2.2 Fi'il (kata kerja) 9
2.2.1 Fi'il Tsulatsy 20
2.2.2 Fi'il Ruba'i 26
2.3 Huruf 27

BAB III SISTEM BERBASIS PENGETAHUAN 29
3.1 Basis Pengetahuan 30
3.2 Memory Kerja 32
3.3 Mesin Inferensi 33
3.3.1 *Forward Chaining* 33
3.3.2 *Backward Chaining* 34
3.3.3 *Certainty Factor* 35

BAB IV PERANCANGAN DAN PEMBUATAN PERANGKAT LUNAK 37
4.1 Deskripsi Perangkat Lunak 37
4.1.1 Masukan 37
4.1.2 Mesin Inferensi 38
4.1.3 Pencarian Arti 39
4.1.4 Luaran dan Penampilan Hasil 40
4.1.5 Modul Tambahan 40
4.2 Perancangan Data 40
4.2.1 Data Masukan 40
4.2.2 Data Mesin Inferensi 42
4.2.3 Data Keluaran 45
4.3 Pembuatan Perangkat Lunak 46
4.3.1 Diagram Alir Data 46
4.3.2 Pembuatan Modul 50
4.3.3 Pembuatan Form 63

4.3.4 Pembuatan Basis Data 64

BAB V UJI COBA DAN EVALUASI 65
5.1 Uji Coba Perangkat Lunak 66
5.2 Evaluasi Perangkat Lunak 70

BAB VI PENUTUP 73
6.1 Kesimpulan 73
6.2 Saran-saran 74

DAFTAR PUSTAKA 75
LAMPIRAN 76

DAFTAR GAMBAR

DAFTAR GAMBAR

Halaman

Gambar 3.1	Struktur Umum Sistem Berbasis Pengetahuan	30
Gambar 3.2	Forward Chaining dan Backward Chaining	34
Gambar 4.1	Struktur Objek Rule	44
Gambar 4.2	Diagram Alir data Level 0 Aplikasi Kamus Bahasa Arab Indonesia	46
Gambar 4.3	Diagram alir Data Level 1 Aplikasi Kamus Bahasa Arab Indonesia	47
Gambar 4.4	Diagram alir data Level 2 Proses Pembentukan Basisi Pngetahuan	48
Gambar 4.5	Diagram Alir Data Level 2 Proses Pencarian Arti Kata	49
Gambar 5.1	Uji Coba Dengan masukan kata masjid	66
Gambar 5.2	Makna dari kata masjid	67
Gambar 5.3	Tampilan Penjelasan	67
Gambar 5.4	Tampilan Hasil Input 'Aalimun	68
Gambar 5.5	Tampilan Makna kata Mu'minin	69
Gambar 5.6	Tampilan Makna kata Mu'minat	70

DAFTAR TABEL

DAFTAR TABEL

Halaman

Tabel 4.1 Transliterasi Huruf Arab 41
Tabel 4.2 Format tabel basisdata kosakata 63
Tabel 4.3 Format tabel basisdata wazan 63

BAB I

PENDAHULUAN

BAB I

PENDAHULUAN

1.1 LATAR BELAKANG

Bahasa Arab merupakan bahasa umat Islam, sehingga untuk dapat memahami agama Islam penguasaan bahasa Arab merupakan hal yang sangat penting. Karena sebagian besar pedoman ajaran Agama Islam ditulis dengan bahasa Arab, diantaranya Al-Qur'an dan Hadits. Sebagai orang yang berpegang kepada Al-Quran dan Islam, kita selayaknya memperhatikan bahasa yang dengannya Al-Quran diturunkan. Karena tidaklah mungkin kita dapat memahami Islam secara sempurna apabila kita tidak menguasai kaidah-kaidah bahasa Arab sampai batas-batas tertentu.

Maka wajiblah kaum Muslimin mendalami bahasa Arab agar dapat memahami nash-nash Arab ketika membaca Al-Quran, Hadits, dan lain-lain. Terutama pada saat shalat, bacaan-bacaannya dapat dipahami dengan benar sehingga shalatnya menjadi khusyu' bersama menghadapnya hati, selain juga dapat memahami apa yang kita ucapkan dalam do'a.

Mengingat pentingnya belajar bahasa Arab, maka kita harus berusaha untuk dapat menguasainya, minimal mengerti akan artinya. Dalam mempelajari bahasa Arab diperlukan penguasaan tata bahasa dan kosakata, tata bahasa dipelajari dalam Ilmu Nahwu dan Sharaf, sedangkan kosakata bisa dipelajari melalui kamus bahasa Arab. Berbeda dengan bahasa yang lain, dalam bahasa Arab bentuk dasar dari suatu kata bisa diturunkan menjadi lebih dari seratus kata sehingga yang terdapat dalam kamus bahasa Arab saat ini adalah bentuk dasarnya dan beberapa bentuk kata turunannya.

Kesulitan yang biasanya dihadapi dalam mencari suatu kata dalam kamus bahasa Arab adalah apabila kata yang dicari bukan merupakan bentuk dasar, karena kata tersebut dicantumkan dibawah bentuk dasarnya, atau malah tidak dicantumkan sama sekali. Hal ini disebabkan banyaknya turunan kata dari satu bentuk kata dasar.

1.2 TUJUAN DAN MANFAAT

Tujuan dari pembuatan perangkat lunak ini adalah untuk memudahkan pengguna dalam mencari arti suatu kata dalam bahasa arab tanpa harus mengetahui bentuk dasarnya, karena perangkat lunak ini yang akan membantu untuk mencari bentuk dasar dari kata bahasa Arab tersebut.

Dengan kemampuan untuk mengetahui bentuk dasar suatu kata, perangkat lunak ini juga dapat dimanfaatkan untuk belajar mencari bentuk dasar suatu kata bahasa Arab.

1.3 PERMASALAHAN

Permasalahan yang dihadapi adalah banyaknya turunan kata dari satu jenis kata kerja, dimana satu bentuk dasar suatu kata bisa memiliki turunan kata lebih dari seratus kata.

Dalam hal mencari bentuk dasar kesulitan yang dihadapi adalah jika didapati suatu kata yang merupakan bentukan (turunan) dari bentuk dasar yang didalamnya terdapat huruf *illat* (*alif, wau, ya*'). Karena turunan dari kata tersebut seringkali hurufnya berubah atau bahkan dihilangkan.

MILIK PERPUSTAKAAN
INSTITUT TEKNOLOGI
SEPULUH - NOPEMBER

1.4 BATASAN MASALAH

Kamus yang lengkap adalah kamus yang memuat seluruh kata dalam bahasa Arab dan diartikan kedalam bahasa Indonesia. Karena target dari pembuatan kamus ini bukan untuk mencari kelengkapan kosakata akan tetapi untuk memudahkan pengguna kamus saat ini untuk mencari bentuk dasar suatu kata, sehinga isi kosakata dalam kamus hanya berupa contoh saja, yang selanjutnya masih dapat ditambahkan.

Sedangkan aturan-aturan pencarian bentuk dasar suatu kata masih terbatas untuk turunan-turunan dari kata yang berasal dari tiga huruf (Fi`il Tsulastsy).

1.5 METODOLOGI

Metode penelitian yang digunakan dalam tugas akhir ini adalah sebagai berikut :

1. Studi literatur dan interview dengan seseorang yang memahami bidang tata bahasa dan struktur bahasa Arab, serta teori kompilasi dengan mengumpulkan data-data baik berupa jurnal, tugas akhir, majalah maupun buku-buku yang mendukung penyelesaian tugas akhir ini.
2. Perancangan *scanner* untuk mengecek apakah input yang dimasukkan sesuai atau tidak, kemudian melakukan perancangan *parser* agar input tersebut dapat ditelusuri bentuk dasarnya.
3. Perancangan dan pembuatan procedure penelusuran bentuk dasar, dengan mencari metode yang paling sesuai apakah cukup menggunakan state automata atau menggunakan *forward* dan *backward chaining*.
4. Perancangan Struktur penyimpanan kata, dan proses searching data.

5. Penyempurnaan Program.
6. Untuk tiap-tiap tahap akan diadakan evaluasi dan penyempurnaan.
7. Penyempurnaan laporan Tugas Akhir.

1.6 SISTEMATIKA PEMBAHASAN

Untuk memudahkan pembacan dan pemahaman mengenai tugas akhir yang dibuat, dalam buku laporan tugas akhir ini terdapat enam bab dengan rincian sebagai berikut :

BAB I. Pendahuluan

BAB II. Sekilas Tentang Isim, Fi'il, dan Huruf.

BAB III. Sistem Berbasis Pengetahuan

BAB IV. Perancangan dan Pembuatan Perangkat Lunak

BAB V. Evaluasi Perangkat Lunak

BAB VI. Penutup

Bab I mencakup antara lain latar belakang pembuatan perangkat lunak, tujuan pembuatannya, permasalahan yang ditemui dalam proses perancangan dan pembuatannya, batasan masalah yang dibuat, serta metodologi yang digunakan.

Bab II membahas sekilas tentang jenis-jenis kata dalam Bahasa Arab dalam hal ini adalah kata benda *(isim)*, kata kerja *(fi'il)* dan huruf. Pembahasan dalam bab in lebih ditekankan kepada pembahasan mengenai masalah kata kerja (fi'il). Sedangkan jenis kata yang lain hanya dibahas contoh-contohnya saja.

Bab III membahas tentang dasar teori yang berkaitan dengan proses komputasi, dalam hal ini teori yang dibahas berkaitan dengan Sistem Berbasis Pengetahuan.

Dalam Bab IV dijelaskan mengenai perancangan dan implementasi perangkat lunak, mulai dari perancangan data, perancangan proses hinggga bagaimana hasil rancangan tersebut diimplementasikan kedalam suatu program.

Bab V akan menjelaskan tentang uji coba dan evaluasi perangkat lunak. Sedangkan Bab VI merupakan penutup dan kesimpulan.

BAB II

SEKILAS TENTANG ISIM, FI'IL DAN HURUF

BAB II

SEKILAS TENTANG ISIM, FI'IL DAN HURUF

Di dalam bab ini dibahas mengenai kata-kata yang ada dalam bahasa arab. Dalam bahasa Arab kata dikenal dengan istilah *kalam*, dimana kalam memiliki arti lafazh yang tersusun dan bermakna lengkap. Kata (*kalam)* dalam bahasa Arab terdiri dari tiga bagian, yaitu :

1. Kata Benda (*Isim*)
2. Kata Kerja (*Fi'il*)
3. Huruf

Pembahasan dalam bab ini lebih ditekankan pada pembahasan mengenai kata kerja *(fi'il)*, karena perubahan bentuk dan penambahan huruf paling banyak terjadi pada jenis kata kerja. Ilmu yang mempelajari mengenai perubahan-perubahan bentuk kata biasa dikenal dengan istilah *Ilmu Sharaf.*

2.1 *ISIM* (KATA BENDA)[1]

Isim (kata benda) adalah kata yang menunjukkan arti benda atau yang dibendakan. Perubahan susunan dan penambahan huruf tidak banyak terdapat dalam jenis kata benda *(isim).* Perubahan-perubahan ini hanya terjadi berkisar pada makna jumlah, yaitu untuk mendapatkan makna jumlah benda yang berbeda-benda. Berdasarkan makna jumlah, kata benda terdiri dari :

a) Kata benda tunggal *(isim mufrad)*, yaitu kata benda yang menunjukkan arti tunggal. Contoh :

---

[1] Anwar, Moch,KH, *Ilmu Nahwu*, Sinar Baru Algesindo, Bandung, 1995

مُسْلِمَةٌ (seorang wanita muslim)

مُسْلِمٌ (seorang laki-laki muslim)

b) Kata yang menunjukkan arti dua *(isim tatsniyah)*, yaitu kata benda yang menunjukkan dua benda. Dalam keadaan *rafa'* ditandai dengan penambahan huruf alif di akhir kata. Contoh :

مُسْلِمَتَانِ (dua orang wanita muslim)

مُسْلِمَانِ (dua oarang laki-laki muslim)

c) Kata benda jamak, yaitu kata benda yang menunjukkan jumlah tiga atau lebih. Kata benda jamak ada tiga macam :

1. Jamak Muannats Salim, kata benda jamak yang tidak berubah dari bentuk mufradnya, menunjukkan jenis perempuan. Ditandai dengan penambahan huruf alif dan ta, Contoh :

   مُسْلِمَاتٌ (wanita-wanita muslim) bentuk mufradnya مُسْلِمَةٌ

2. Jamak Mudzakar Salim, kata benda jamak yang tidak berubah dari bentuk mufradnya , menunjukkan jenis laki-laki. Ditandai dengan penambahan huruf wawu jamak. Contoh :

   مُسْلِمُوْنَ (beberapa orang laki-laki muslim) bentuk mufradnya مُسْلِمٌ

3. Jamak Taksir, kata benda jamak yang susunannya berubah dari bentuk mufradnya. Untuk kata benda jenis ini sifatnya *sima'i* sehingga tidak bisa dikiaskan dan harus dihafalkan. Contoh :

MILIK PERPUSTAKAAN
INSTITUT TEKNOLOGI
SEPULUH - NOPEMBER

كُتُبٌ (beberapa kitab) bentuk mufradnya كِتَابٌ

مَسَاجِدٌ(beberapa masjid) bentuk mufradnya مَسْجِدٌ

Jenis isim yang lain adalah :

1. Isim Ma'rifat

Isim Ma'rifat adalah lafazh yang menunjukkan benda tertentu. Misal lafazh Zaid, menunjukkan orang yang bernama Zaid.

Isim Ma'rifat ada lima macam , yaitu :

a. Isim dhamir (kata ganti) , Isim dhamir dibagi menjadi dua macam yaitu

i. Dhamir Muttasil, yaitu dhammir yang bersambung di tengah-tengah atau diakhir kata, meliputi :

هُ، هُمَا، هُمْ، هَا، هُمَا، هُنَّ، كَ، كُمَا، كُمْ، كِ،كُمَا ، كُنَّ، يَ، نَا

ii. Dhamir Munfashil, yaitu dhammir yang penulisannya selalu dipisah dan ditulis di awal kata, meliputi :

هُوَ، هُمَا، هُمْ، هِيَ، هُمَا، هُنَّ، اَنْتَ، اَنْتُمَا، انْتُمْ، انْتِ، اَنْتُمَا، انْتُنَّ، انَا، نَحْنُ

b. Isim 'alam (nama), seperti lafazh : Zaid (nama orang), Mekah (nama kota)

c. Isim mubham yaitu isim yang tidak diketahui dengan pasti, isim mubham ini mencakup isim isyarah (kata penunjuk) yaitu isim yang mengisyaratkan sesuatu dan isim maushul (kata ganti penghubung) contoh :

هذا ، هذِهِ ،الَّذِى ، اَلَّذِيْنَ

d. Isim yang diberi alif lam, untuk menunjukkan benda terrentu seperti lafazh :

اَلكِتَا بُ artinya kitab itu

الرَّجُلُ artinya seorang laki-laki itu

e. Isim yang didhafatkan (disandarkan) kepada salah satu di antara yang empat bagian (yaitu isim mudhmar (dhamir), isim 'alam, isim mubham, dan isim yang diberi alif dan lam). Contoh :

كِتَـــابُكَ (kitabmu) asalnya كِتَابٌ (kitab) dan أَنْتَ (kamu) digabungkan menjadi satu. أَنْتَ berubah menjadi كَ karena posisinya diakhir kata.

2. Isim Nakirah

Isim nakirah ialah, setiap isim yang jenisnya bersifat umum yang tidak menentukan sesuatu perkara dan lainnya, setiap isim yang layak dimasuki alif dan lam. Contoh :

رَجُلُ (seorang laki-laki /umum)

كِتَابُ (sebuah kitab/umum)

2.2 *FI'IL* (KATA KERJA)[2]

Kata kerja *(fi'il)* merupakan kata yang menunjukkan suatu pekerjaan dan disertai dengan makna waktu. Fi'il dalam bahasa arab kaya akan bentuk bentuk

---

[2]Anwar, Moh, KH, *Ilmu Sharaf*, Sinar baru Algesindo, Bandung, 1989

turunan, yang merupakan perubahan atau pengembangan arti makna dari yang dikandung oleh bentuk asal/dasar, dalam hal ini fi'il mujarrad (kata kerja dasar).

Kata kerja dasar dalam bahasa Arab ada dua macam, yang pertama jumlahnya terdiri atas tiga huruf dan yang kedua terdiri atas empat huruf. Kebanyakan kata kerja dalam bahasa Arab terdiri atas tiga huruf yang merupakan akarnya, sedangkan kata kerja yang akarnya terdiri atas empat huruf jumlahnya sedikit dan jarang digunakan. Sedangkan apabila ditinjau dari jenisnya, kata kerja terdiri dari tiga macam yaitu kata kerja bentuk lampau *(fi'il madhi)*, kata kerja bentuk sekarang/akan datang *(fi'il mudhari)*, dan kata kerja bentuk perintah *(fi'il amr)*.

Berikut ini akan dijelaskan mengenai pengembangan makna dari fi'il madhi. kemudian akan dibahas tentang pengembangan dari bentuk fi'il mujarrad. Kata kerja ini terdiri dari tiga macam, yaitu:

a) Kata kerja bentuk lampau (*fi'il madhi*), contoh :

(telah menulis) كَتَبَ

(telah menolong) نَصَرَ

b) Kata kerja bentuk sedang / akan datang (*fi'il mudhari'*), contoh :

(sedang /akan menulis) يَكْتُبُ

(sedang /akan menolong) يَنْصُرُ

c) Kata kerja bentuk perintah (*fi'il amr*), contoh :

(tulislah) أكْتُبْ

(tolonglah) أُنْصُرْ

Fi'il mudhari dan Fi'il Amr merupakan turunan dari *Fi'il madhi*, turunan-turunan yang lain adalah *Masdar*, *Isim Fail*, *Isim Maf'ul*, *Fi'il Nahi*, *Isim Zaman*, *Isim Makaan*, *Isim Alat*. Bentuk perubahan dari *fi'il madhi* menjadi kata–kata yang lain secara keseluruhan seperti dalam contoh dibawah ini [3]:

1. نَصَرَ : fi'il madhi., artinya sudah menolong
2. يَنْصُرُ : fi'il mudhorik, artinya sedang / akan menolong
3. نَصْرًا : Masdar ghoiru mim, artinya pertolongan
4. مَنْصَرًا : Masdar mim, artinya pertolongan
5. نَاصِرٌ : Isim Fail, artinya yang menolong
6. مَنْصُوْرٌ : Isim maf'ul. Artinya yang ditolong
7. أُنْصُرْ : fi'il amr, artinya tolonglah.
8. لاَتَنْصُرْ : fi'il nahi, artinya jangan ditolong
9. مَنْصَرٌ : isim zaman/makan , artinya tempat / waktu menolong
10. مِنْصَرٌ : isim alat, artinya alat untuk menolong

Perubahan fi'il yang lain berupa pemberian kata ganti (*dhamir muttasil*) pada akhiran fi'il yang menerangkan bilangan dan jenis., biasa dikenal dengan istilah *mabni ma'lum,* sedangkan apabila berupa kata kerja pasif disebut *mabni majhul*, berikut ini fi'il mabni ma'lum yang terdapat pada fi'il madhi, mudhari', amr dan nahi, yang jumlahnya ada empat belas macam, untuk fi'il 'amr, dan nahi susunan

---

[3] Ibid, halaman 2

orang pertama tunggal dan jama' tidak ada karena posisinya sebagai pemberi perintah/larangan [4]:

1. نَصَرَ artinya dia seorang laki-laki telah menolong

   يَنْصُرُ artinya dia seorang laki-laki sedang/akan menolong

   لِيَنْصُرْ artinya dia seorang laki-laki supaya menolong

   لاَيَنْصُرْ artinya dia seorang laki-laki jangan menolong

2. نَصَرَا artinya mereka dua orang laki-laki telah menolong

   يَنْصُرَانِ artinya mereka dua orang laki-laki sedang/akan menolong

   لِيَنْصُرَا artinya mereka dua orang laki-laki supaya menolong

   لاَيَنْصُرَا artinya mereka dua orang laki-laki jangan menolong

3. نَصَرُوا artinya mereka tiga orang laki-laki/lebih telah menolong

   يَنْصُرُوْنَ artinya mereka tiga orang laki-laki/lebih sedang/akan menolong

   لِيَنْصُرُوا artinya mereka tiga orang laki-laki/lebih supaya menolong

   لاَيَنْصُرُا artinya mereka tiga orang laki-laki/lebih jangan menolong

---

[4] Ibid, halaman 21

MILIK PERPUSTAKAAN
INSTITUT TEKNOLOGI
SEPULUH - NOPEMBER

4. نَصَرَتْ artinya dia seorang perempuan telah menolong

تَنْصُرُ artinya dia seorang perempuan sedang/akan menolong

لِتَنْصُرْ artinya dia seorang perempuan supaya menolong

لاتَنْصُرْ artinya dia seorang perempuan jangan menolong

5. نَصَرَتَا artinya mereka dua orang perempuan telah menolong

تَنْصُرَانِ artinya mereka dua orang perempuan sedang/akan menolong

لِتَنْصُرَا artinya mereka dua orang perempuan supaya menolong

لاتَنْصُرَا artinya mereka dua orang perempuan jangan menolong

6. نَصَرْنَ artinya mereka tiga orang perempuan/lebih telah menolong

يَنْصُرْنَ artinya mereka tiga orang perempuan/lebih sedang/akan menolong

لِيَنْصُرْنَ artinya mereka tiga orang perempuan/lebih supaya menolong

لايَنْصُرْنَ artinya mereka tiga orang perempuan/lebih jangan menolong

7. نَصَرْتَ artinya kamu seorang laki-laki telah *menolong*

تَنْصُرُ artinya kamu seorang laki-laki sedang/akan menolong

اُنْصُرْ artinya kamu seorang laki-laki tolonglah

لاتَنْصُرْ artinya kamu seorang laki-laki jangan menolong

8. نَصَرْتُمَا artinya kalian dua orang laki-laki telah menolong

تَنْصُرَانِ artinya kalian dua orang laki-laki sedang/akan menolong

اُنْصُرَا artinya kalian dua orang laki-laki harus menolong

لاتَنْصُرَا artinya kalian dua orang laki-laki jangan menolong

9. نَصَرْتُمْ artinya kalian tiga orang laki-laki/lebih telah menolong

تَنْصُرُوْنَ artinya kalian tiga orang laki-laki/lebih sedang/akan menolong

اُنْصُرُوْا artinya kalian tiga orang laki-laki/lebih harus menolong

لاتَنْصُرُوْا artinya kalian tiga orang laki-laki/lebih jangan menolong

10. نَصَرْتِ artinya kamu seorang perempuan telah menolong

تَنْصُرِيْنَ artinya kamu seorang perempuan sedang/akan menolong

انْصُرِى artinya kamu seorang perempuan harus menolong

لاتَنْصُرِى artinya kamu seorang perempuan jangan menolong

11. نَصَرْتُمَا artinya kalian dua orang perempuan telah menolong

تَنْصُرَانِ artinya kalian dua orang perempuan sedang/akan menolong

انْصُرَا artinya kalian dua orang perempuan harus menolong

لاتَنْصُرَا artinya kalian dua orang perempuan jangan menolong

12. نَصَرْتُنَّ artinya kalian tiga orang perempuan/lebih telah menolong

تَنْصُرْنَ artinya kalian tiga orang perempuan/lebih sedang/akan menolong

انْصُرْنَ artinya kalian tiga orang perempuan/lebih harus menolong

لاتَنْصُرْنَ artinya kalian tiga orang perempuan/lebih jangan menolong

13. نَصَرْتُ artinya saya seorang laki-laki/perempuan telah menolong

انْصُرُ artinya saya seorang laki-laki/perempuan sedang/akan menolong

14. نَصَرْنَا artinya kami telah menolong

نَنْصُرُ artinya kami sedang/akan menolong

Urutan pertama sampai ketiga menunjukkan orang ketiga laki-laki (tunggal, tatsniyah,jamak), urutan empat sampai enam menunjukkan orang ketiga perempuan (tunggal, tatsniyah,jamak), urutan tujuh sampai sembilan menunjukkan orang kedua laku-laki (tunggal, tatsniyah,jamak), urutan sepuluh sampai dua belas menunjukkan orang kedua perempuan (tunggal, tatsniyah,jamak), kemudian urutan tiga belas menunjukkan orang pertama tunggal, dan yang terakhir menunjukkan orang pertama jamak.

Untuk menyatakan pola kata kerja, ahli tata bahasa menggunakan konsonan kata kerja فعل (fa'ala) dan فعلل (fa'lala). Fa'ala digunakan untuk fi'il yang jumlah huruf asalnya tiga huruf *(tsulatsy)* , dan fa'lala digunakan untuk fi'il yang jumlah huruf asalnya ada empat huruf *(ruba'i)*. Untuk fi'il tsulatsy huruf ف (fa) menggambarkan akar pertama biasa disebut dengan fa' fi'il , huruf ع ('ain) akar kedua atau biasa disebut 'ain fi'il, dan ل (lam) akar ketiga atau lam fi'il, sedangkan untuk ruba'i huruf lam kedua sebagai akar keempat atau lam fi'il kedua. Pola-pola dasar dari kata kerja ini dikenal dengan istilah wazan.. Huruf-huruf tambahan yang memasuki fi'il mujarrad ada sepuluh macam yaitu س, ء ,

ل, ت , م , و , ن, ي, ه, ا .

Berikut ini pembagian fi'il berdasarkan jumlah huruf asalnya [5]:

1. *Fi'il Tsulatsy*, fi'il yang tersusun dari tiga huruf. Fi'il ini terbagi lagi menjadi dua bagian yaitu :
   - *Fi'il Tsulatsy Mujarrad,* yaitu fi'il yang terdiri dari tiga huruf dan didalamnya tidak didapatkan huruf tambahan, contoh :

     نَصَرَ، قَتَلَ، كَرَمَ، وَعَدَ، كَسَرَ

   - *Fi'il Tsulatsy Mazied,* yaitu fi'il yang huruf asalnya terdiri dari tiga huruf yang mendapatkan tambahan satu huruf disebut *Fi'il Tsulatsy Mazied Ruba'i* , yang mendapatkan tambahan dua huruf disebut *Fi'il Tsulatsy Mazied Khumasy* dan yang mendapatkan tambahan tiga huruf disebut *Fi'il Tsulatsy Mazied Sudasy.* Contoh :
     - Tsulatsy Mazied Ruba'i :

       قَاتَلَ asalnya قَتَلَ

       أَكْرَمَ asalnya كَرَمَ

       أَوْعَدَ asalnya وَعَدَ

     - Tsulatsy Mazied Khumasy

       انْكَسَرَ asalnya كَسَرَ

       تَوَاعَدَ asalnya وَعَدَ

       تَكَسَّرَ asalnya كَسَرَ

     - Tsulatsy Mazied Sudasy

       إِسْتَغْفَرَ asalnya غَفَرَ

2. *Fi'il Ruba'i*, fi'il yang huruf asalnya tersusun dari empat huruf. Fi'il ini terbagi lagi menjadi :

MILIK PERPUSTAKAAN INSTITUT TEKNOLOGI SEPULUH - NOPEMBER

[5] Ibid, halaman 4

- *Fi'il Ruba'i Mujarrad,* yaitu *fi'il* yang terdiri dari empat huruf dan tidak didapatkan huruf tambahan, contoh :

دَحْرَجَ، تَرجَمَ

- *Fi'il Ruba'i Mazied,* yaitu *fi'il* yang huruf asalnya terdiri dari empat huruf dan mendapatkan tambahan huruf satu disebut *Fi'il Ruba'i Mazied Khumasy* dan yang mendapatkan tambahan dua huruf disebut *Fi'il Ruba'i Mazied Sudasy*, contoh :
  - Fi'il Ruba'i Mazied Khumasy تَدَحْرَجَ
  - Fi'il Ruba'i Mazied Sudasy اِحْرَنْجَمَ
- *Fi'il Ruba'i Mulhaq,* yaitu fi'il yang asalnya berasal dari tsulatsy mujarrad akan tetapi diikutkan sertakan kepada *ruba'i mujarrad, Fi'il Ruba'i Mulhaq* yang mendapatkan tambahan satu huruf disebut *Fi'il Ruba'i Mulhaq Mazied Khumasy* contoh

جَلْبَبَ، وَحْقَلَ، جَهْوَرَ، تَجَلْبَبَ، تَجَهْوَرَ

Bina fi'il ada tujuh macam yaitu [6]:

1. *Fi'il Bina Shoheh* ialah tiap-tiap fi'il yang Fa' Fi'il , 'Ain Fi'il, dan Lam Fi'il tidak berupa huruf hamzah dan huruf illat (wawu, alif, ya') serta 'Ain dan Lam fi'ilnya tidak terdiri dari huruf sejenis, contoh :

نَصَرَ، ضَرَبَ

2. *Fi'il Bina 'Ajwaf* , ialah tiap-tiap fi'il yang ai'n fi'ilnya berupa huruf ilat, Kalau huruf ilatnya berupa huruf wawu maka disebut *Bina' Ajwaf Wawi,* Jika berupa huruf ya disebut *Bina Ajwaf Ya'i* contoh :

صَانَ asalnya صَوَنَ, 'ain fi'il berupa huruf wawu

سَارَ asalnya سَيَرَ, 'ain fi'il berupa huruf ya'

3. *Fi'il Bina Mahmuz* ialah tiap-tiap fi'il yang Fa', 'Ain, atau Lam Fi'ilnya berupa hamzah. Kalau fa' fi'ilnya berupa huruf hamzah disebut *Bina Mahmuz*

Fa', kalau 'Ain Fi'ilnya yang berupa huruf hamzah disebut *Mahmuz 'Ain,* dan kalau yang berupa huruf hamzah lam fi'ilnya disebut *Mahmuz Lam,* contoh :

أَكَلَ mahmuz fa', fa' fi'ilnya berupa huruf hamzah

رَأَي mahmuz 'ain, 'ain fi'ilnya berupa huruf hamzah

نَشَأَ mahmuz lam, lam fi'ilnya berupa huruf hamzah

4. *Fi'il Bina Mudlo'af , dibagi menjadi dua bagian yaitu :*
   - *Fi'il Bina Mudlo'af Tsulatsy* ialah tiap-tiap fi'il yang 'ain dan lam fi'il terdiri dari huruf yang sejenis, contoh:

     مَدَّ asalnya مَدَدَ

     فَرَّ asalnya فَرَرَ

   - *Fi'il Bina Mudlo'af Ruba'i*, ialah tap-tiap Fi'il yang Fa' Fi'il dan lam fi'il yang pertama terdiri dari huruf yang sejenis, 'Ain fi'il dan lam Fi'il yang kedua terdiri dari hutuf yang sejenis juga, contoh :

     وَسْوَسَ، قَلْقَلَ

5. *Fi'il Bina' Naqis*, ialah tiap-tiap fi'il yang lam fi'ilnya berupa huruf illat. Kalau berupa huruf ilat wawu disebut *Fi'il Bina' Naqis Wawi,* kalau huruf ilat ya' disebut *Fi'il Bina Naqis Ya'i*, contoh

   غَزَا naqis wawi asalnya غَزَوَ, lam fi'ilnya berupa huruf wawu

   سَرَى naqis ya'I asalnya سَرَيَ , lam fi'ilnya berupa huruf ya'

6. *Fi'il Bina' Lafif,* dibagi menjadi dua bagian yaitu :
   - *Fi'il Bina' Lafif Maqrun*, ialah tiap-tiap fi'il yang 'Ain dan lam Fi'ilnya terdiri dari huruf illat contoh :

---

[6]Nadwi, M Maftuhin Sholeh, *Ilmu Bahasa Arab Shorof,* Pustaka Jaya, Surabaya, 1982

قَوِيَ ، شَوَى 'ain fi'il berupa huruf wawu dan lam fi'il berupa huruf ya'

- *Fi'il Bina' Lafif Mafruq* ialah tiap-tiap fi'il yang fa' fi'il dan lam fi'ilnya terdiri dari huruf illat, contoh:

وَلِى ، وقى lam fi'il berupa huruf wawu dan lam fi'il berupa huruf ya'

7. *Fi'il Mu'tal Misal*, ialah tiap-tiap fi'il yang fa' fi'ilnya berupa huruf illat, kalau berupa huruf illat wawu disebut Fi'il bina' Mu'tal Misal Wawi, dan kalau berupa huruf ya' disebut Fi'il Bina' Mu'tal Misal Ya'i , contoh:

وَعَدَ fa' fi'il berupa huruf illat wawu

يَسَرَ fa' fi'il berupa huruf illat ya'

Pencarian pola kata untuk jenis selain bina' shohih merupakan suatu masalah tersendiri karena dalam susunan hurufnya terdapat huruf illat yang bisa menyebabkan kerancuan dengan huruf tambahannya, juga pada keadaan tertentu ada sebagian yang dihilangkan dan ada juga yang digantikan dengan huruf yang lain.

2.2.1 *Fi'il Tsulatsy*

Fi'il yang tersusun dari tiga huruf dikenal dengan istilah *fi'il tsulatsy* , fi'il tsulatsy yang tidak mendapatkan huruf tambahan disebut dengan *fi'il tsulatsy mujarrad*. Pembahasan mengenai pola perubahan (wazan) *fi'il tsulatsy mujarrad* terbagi menjadi enam pembahasan (enam wazan), keenam wazan tersebut adalah :

a) *Wazan Fa'ala Yaf'ulu* - فَعَلَ – يَفْعُلُ , contoh : كَتَبَ – يَكْتُبُ

b) *Wazan Fa'ala Yaf'ilu* - فَعَلَ – يَفْعِلُ , contoh: ضَرَبَ – يَضْرِبُ

c) *Wazan Fa'ala Yaf'alu* - فَعَلَ – يَفْعَلُ , contoh: فَتَحَ – يَفْتَحُ

d) *Wazan Fa'ila Yaf'alu* - فَعِلَ – يَفْعَلُ , contoh: عَلِمَ – يَعْلَمُ

e) *Wazan Fa'ila Yaf'ilu* - فَعِلَ – يَفْعِلُ , contoh: حَسِبَ – يَحْسِبُ

f) *Wazan Fa'ula Yaf'ulu* - فَعُلَ – يَفْعُلُ , contoh: حَسُنَ – يَحْسُنُ

Fi'il Tsulatsy yang mendapatkan tambahan satu huruf disebut dengan *fi'il tsulatsy mazied ruba'i,* terdapat tiga wazan , yaitu :

a) Wazan Fa' 'ala - فَعَّلَ, ditambah 'ain fi'ilnya.

Makna fi'il wazan fa' 'ala adalah :

i. Sebagai fi'il muta'adi yaitu fi'il yang menuntut keberadaan maf'ul / pelaku.

ii. Menunjukkan arti jamak

iii. Menisbatkan maful kepada arti fi'il

iv. Meniadakan / memiliki makna lain (kias)

v. Menjadikan fi'il sebagai isim.

Contoh : فَرَّحَ artinya menggembirakan, asalnya فَرِحَ (gembira)

b) Wazan Faa'ala - فَاعَلَ, ditambahkan huruf alif setelah fa' fi'il.

Makna fi'il wazan faa'ala adalah :

i. Menunjukkan arti saling berbuat

ii. Memiliki makna taksir (banyak)

iii. Memiliki makna kesengajaan

iv. Makna tsulatsy mujarrad (tidak berubah)

Contoh : قَا تَلَ (saling membunuh/berperang) asalnya قَتَلَ (membunuh)

c) *Wazan Af'ala* - أَفْعَلَ ditambahkan huruf hamzah sebelum fa' fi'il, contoh :

Makna fi'il wazan af'ala adalah :

i. Sebagai fi'il muta'adi yaitu fi'il yang menuntut keberadaan maf'ul / pelaku

ii. Menjadi sesuatu

iii. Memiliki makna hilang

iv. Menuju tempat tujuan, dalam hal ini huruf asalnya bukan berupa fi'il akan tetapi berupa isim

v. Memasuki sesuatu (waktu)

vi. Memiliki kesan bersungguh-sungguh

vii. Memiliki makna diambil dari fi'ilya

Contoh : أَكْرَمَ (memuliakan) asalnya كَرُمَ (mulia)

Fi'il Tsulatsy yang mendapatkan tambahan dua huruf disebut dengan fi'il *tsulatsy mazied khumasy,* terdapat lima wazan, yaitu :

a) *Wazan Infa'ala* – إِنْفَعَلَ mendapat tambahan huruf hamzah dan nun.

Makna fi'il wazan infa'ala adalah :

i. Hasil dari suatu perbuatan

ii. Menunjukkan arti pengaruh dari suatu perbuatan.

Contoh : إِنْكَسَرَ (menjadi pecah) asalnya كَسَرَ (pecah)

MILIK PERPUSTAKAAN
INSTITUT TEKNOLOGI
SEPULUH - NOPEMBER

b) *Wazan If ta'ala* – إِفْتَعَلَ, mendapat tambahan huruf hamzah dan ta'.

Makna fi'il wazan ifta'ala adalah :

i. Hasil dari suatu perbuatan

ii. Isim yang dijadikan fi'il

iii. Menunjukkan arti sungguh-sungguh

iv. Menunjukkan suatu kesulitan

v. Menunjukkan makna fa'ala

vi. Memiliki makna saling

Contoh : إِجْتَمَعَ(menjadi berkumpul) asalnya جَمَعَ (berkumpul)

c) *Wazan If'alla* – إِفْعَلَّ , mendapat tambahan huruf hamzah dan huruf fa'

fi'ilnya diberi tasydid.

Makna fi'il wazan if'alla adalah

i. Menunjukkan arti sifat / warna

ii. Menunjukkan arti sangat

Contoh : إِحْمَرَّ(sangat merah) asalnya حَمَرَ (merah)

d) *Wazan Tafa' 'ala* – تَفَعَّلَ, mendapat tambahan huruf ta' dan huruf 'ain fi'il

diberi tasydid.

Makna fi'il wazan tafa' 'ala

i. Memperoleh / mendapatkan beban suatu pekerjaan

ii. Memaksakan sesuatu

iii. Hasil suatu perbuatan

iv. Fa'il menjadikan fi'il sebagai maf'ul

v. Memiliki makna menjadi

vi. Hasil dari suatu perbuatan secara beruntun

vii. Menuntut sesuatu

viii. Isim dihukumi fi'il

ix. Menisbatkan kepada sesuatu

Contoh : تَكَسَّرَ (menjadi pecah) asalnya كَسَرَ (pecah)

e) *Wazan Tafaa'ala* - تَفَاعَلَ, mendapat tambahan huruf ta' dan alif

Makna fi'il wazan tafa'ala adalah :

i. Menunjukkan makna saling melakukan

ii. Hasil dari suatu perbuatan

iii. Untuk menampakkan apa-apa yang tidak ada / pura-pura

iv. Memiliki makna berangsur-angsur / bertahap

v. Memiliki makna mujarrad (dinisbatkan kepada nama Allah / sifat Allah)

Contoh : تَبَا عَدَ (saling menjauhi) asalnya بَعُدَ (jauh)

Fi'il tsulatsy yang mendapatkan tambahan tiga huruf disebut dengan *fi'il tsulatsy mazied sudasy*, terdapat enam wazan, yaitu :

a) *Wazan Istaf'ala* – إِسْتَفْعَلَ, mendapat tambahan huruf hamzah, sin dan ta'

Makna wazan istaf'ala

i. Menunjukkan makna mencari sesuatu / menuntut sesuatu

ii. Memiliki tujuan maf'ul sebagai sifat atau maf'ul itu hendak disifati

iii. Perubahan

iv. Makna terpaksa

v. Sebagai akibat dari suatu perbuatan

Contoh : إستَغْفَرَ (meminta maaf) asalnya غَفَرَ (memaafkan)

b) *Wazan If'au'ala* – إِفْعَوْعَلَ mendapat tambahan huruf hamzah,wau dan huruf 'ain fi'il.

Makna wazan if'au'ala adalah :

i. Memiliki makna sangat

ii. Makna mujarrad

Contoh : إحْقَوْقَفَ – حَقَفَ

c) *Wazan If'awwala* – إِفْعَوَّلَ, mendapat tambahan huruf hamzah dan wau yang bertasydid.

Contoh : اجلَوَّذَ (lalu lalang dengan cepat) asalnya جَلَذَ

d) *Wazan If'aalla* – إِفْعَالَّ , mendapat tambahan huruf hamzah, alif dan lam fi'ilnya diberi tasydid

Contoh : احْمَارَّ (menjadi merah) asalnya حَمِرَ (merah)

Makna wazan if'awalla dan if'aalla adalah memiliki arti sangat

e) *Wazan If'anlaa* – إِفْعَنْلَى mendapat tambahan huruf hamzah, nun, dan alif mankus.

Contoh : إِسْلَنْقَى (terlentang) asalnya سَلَقَ (merebus)

f) *Wazan If'anlala* - إِفْعَنْلَلَ, mendapat tambahan huruf hamzah, nun dan lam fi'il kedua.

Contoh : إِقْعَنْسَسَ (terlambat dan mundur kebelakang) قَعَسَ

2.2.2 *Fi'il Ruba'i*

Fi'il ruba'I merupakan fi'il yang huruf asalnya tersusun dari empat huruf, fi'il ruba'i yang tidak mendapatkan huruf tambahan disebut fi'il ruba'i mujarrad, yang memiliki hanya satu wazan yaitu wazan *fa'lala*.

Contoh : دَحْرَجَ (د = fa' fi'il, ح = 'ain fi'il, ر = lam fi'il pertama dan ج = lam fi'il kedua)

Fi'il ruba'i mazied khumasi, merupakan fi'il yang asalnya dari empat huruf kemudian ditambah satu huruf sehingga jumlahnya menjadi lima huruf, memiliki satu wazan, yaitu wazan *tafa'lala* (mendapat tambahan huruf ta'). Makna wazan tafa'lala adalah sebagai akibat dari suatu perbuatan.

Contoh : تَدَحْرَجَ (menjadi terguling) asalnya دَحْرَجَ (menggulingkan)

Fi'il ruba'i mazied sudasy, merupakan fi'il ruba'i yang mendapatkan tambahan dua huruf sehingga jumlahnya menjadi enam huruf, memiliki dua wazan, yaitu : *if'anlala* (mendapat tambahan huruf hamzah dan nun) dan *if'alalla*

(mendapat tambahan huruf hamzah dan huruf lam fi'il kedua bertasydid). Makna fi'il ruba'i mazied sudasy adalah memiliki arti sangat.

Contoh : اِحْرَنْجَمَ (berdesakan) asalnya حَرَجَمَ (sempit)

Fi'il Ruba'i Mulhaq, merupakan fi'il yang asalnya terdiri dari tiga huruf kemudian ditambah satu huruf disamakan wazannya dengan ruba'i mujarrad, terdapat enam wazan, yaitu : *fau'ala, fa'wala, fai'ala, fa'yala, fa'laa, fa'lala.*

Sedangkan fi'il ruba'i mulhaq yang mendapatkan tambahan satu huruf disebut fi'il ruba'i mulhaq mazied khumasy, fi'il ini diserupakan dengan wazan fi'il ruba'i mazied khumasy. Memiliki enam wazan yaitu : *tafau'ala, tafa'wala, tafai'ala, tafa'yala, tafa'laa, tafa'lala.*

2.3 HURUF[7]

Huruf adalah kalam / kata yang tidak dapat berdiri sendiri. Maksudnya huruf merupakan kata yang dalapat menunjukkan arti apabila dirangkaikan dengan kata yang lain. Macam-macam huruf ada banyak sekali seperti *huruf jarr, huruf athaf, huruf nashab, huruf jazm* dan sebagainya.

Berikut ini huruf-huruf Jarr, Nashab, Jazm :

Huruf-huruf Jarr meliputi : حَتّى ، رُبَّ ، البَاءُ ، الكَافَ ، اللاَّمُ ، على ، فِي ، عَن ، إلى ، مِنْ dan huruf-huruf qasam(sumpah) seperti : بَاءٌ ، تَاءٌ ، وَاوٌ

Huruf-huruf Nashob meliputi : إذَنْ ، لِكَى ، لَنْ ، أَنْ

Huruf-huruf jazem meliputi : لاَمُ الأَمِ ، أَلَمَّا ، لَمَّا ، أَلَمْ ، لَمْ ، لاَمَ النَّهِى

[7]Anwar, Moch, KH, *Ilmu Nahwu*, Sinar Baru Algesindo, Bandung, 1995

MILIK PERPUSTAKAAN
INSTITUT TEKNOLOGI
SEPULUH - NOPEMBER

Contoh pemakaian huruf :

- اَلْمَالُ لِزَيْدٍ = harta milik Zaid , huruf lam artinya untuk,karena, milik.

- زَيْدٌ كَالْبَدْرِ = Zaid itu bagaikan bulan purnama, huruf kaf artinya seperti

- وَاللهِ، بِاللهِ، تَاللهِ = demi Allah , huruf qasam.

Huruf sengaja tidak dibahas secara lebih detil karena begitu luasnya pembahasan bahasa arab mengenai huruf, sampai-sampai ada suatu ilmu tersendiri yang khusus membahas tentang huruf. Selain itu huruf juga tidak mempunyai pola perubahan sehingga bisa diketahui artinya secara langsung. Dalam tugas akhir ini lebih ditekankan kepada perubahan yang terjadi pada suatu kata kerja / fi'il.

BAB III

SISTEM BERBASIS PENGETAHUAN

BAB III

SISTEM BERBASIS PENGETAHUAN

Sistem Berbasis Pengetahuan merupakan salah satu bagian dari Kecerdasan Buatan, dengan dasar pengetahuan yang dimiliki komputer bisa menemukan solusi yang tepat dari sejumlah alternatif solusi yang ada. Tugas akhir ini menggunakan Sistem Berbasis Pengetahuan untuk mencari bentuk dasar dari suatu kata dalam bahasa arab, dalam hal ini tentu hanya dibatasi kepada kata-kata yang bisa dikias saja. Untuk kata-kata yang sifatnya tidak bisa dikias (*sima'i*) harus disimpan langsung kedalam basisdata.

Pengetahuan terdiri dari fakta, pemikiran, teori, prosedur dan hubungannya satu sama lain. Pengetahuan juga merupakan informasi terorganisasi dan teranalisa agar bisa lebih mudah dimengerti dan bisa diterapkan pada pemecahan masalah dan pengambilan keputusan.

Dalam istilah umum, sebuah sistem Berbasis Pengetahuan dapat didefinisikan sebagai :

> *"Sebuah sistem terkomputerisasi yang menggunakan pengetahuan tentang beberapa domain yang mendatangkan pemecahan sebuah masalah dari domain tersebut. Pemecahan ini sama halnya dengan hasil dari pengetahuan manusia tentang domain sebuah masalah jika dihadapkan pada masalah tersebut*[8]

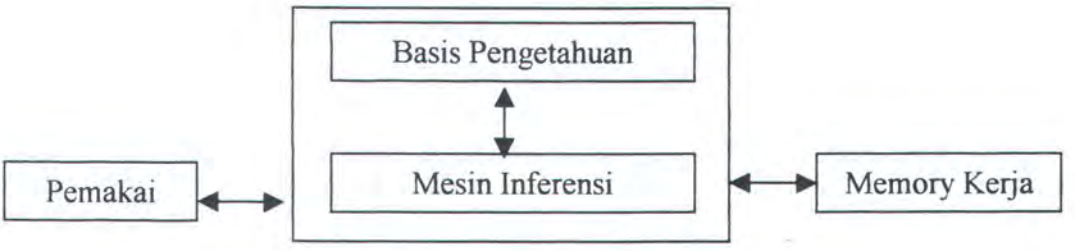
Sistem Berbasis Pengetahuan merupakan suatu sistem yang didalamnya menggunakan dasar pengetahuan yang dimilikinya untuk mencari suatu solusi

---

[8] Gonzalez Avelino J and Dankel Douglas., *"The Engineering of Knowledge-Based System Theory and Practice"*, Prentice Hall International Edition, 1993

terhadap suatu permasalahan tertentu. Dasar pengetahuan merupakan refleksi dari kemampuan memecahkan masalah dari seorang pakar mengenai beberapa domain dan menggunakan beberapa kemampuan untuk memecahkan masalah dengan cara yang hampir sama dengan seorang pakar.

Struktur umum dari suatu sistem berbasis pengetahuan terdiri dari tiga bagian yaitu pemakai/user, mesin inferensi, basis pengetahuan, dan memory kerja, seperti yang dapat dilihat pada gambar 3.1.



Gambar 3.1 Struktur Umum Sistem Berbasis Pengetahuan.

Dengan struktur seperti diatas di mana algoritma (dasar pengetahuan) dan program (mesin inferensi) strukturnya terpisah, sehingga memungkinkan untuk memperbaharui pengetahuan tanpa harus mengkompile ulang program.

3.1. Basis Pengetahuan

Basis pengetahuan merupakan inti dari progam Sistem Berbasis Pengetahuan di mana basis pengetahuan ini merupakan representasi pengetahuan *(Knowledge Representation)* dari seorang pakar. Dalam hal ini representasi pengetahuan yang digunakan dalam Tugas Akhir ini menggunakan Metode Kaidah Produksi. Metode Kaidah Produksi merupakan salah satu metode yang cukup populer dalam Sistem Berbasis Pengetahuan. Cukup banyak

program-program sistem pakar yang dikembangkan dengan menggunakan metode ini.

Program-program terkenal yang menggunakan metode ini antara lain[9] :

1. *Dendral*, merupakan program yang dikembangkan oleh Ilmuwan Stanford University (Buchanan, Feigenbaum dan Lederberg) bekerja sama dengan Sutherland & Co dan Massachusetts Spectrum Laboratory. Program ini berguna untuk mencari rumus kimia dari suatu campuran senyawa organik.
2. *Waterman's Program (1970)*, merupakan program permainan poker dengan menggunakan metoda heuristic.
3. *Mycin – Teiresias,* merupakan program untuk mendiagnosa suatu penyakit. Program ini dikembangkan oleh E. Shortliffe, R. Davis, dan ilmuwan Stanford University (Buchanan-Feigenbaum team).
4. Dan masih banyak lagi program program sistem pakar yang lain, diantaranya SU/X, VM, Crysalis, Molgen, Rita, Prospector, dan sebagainya.

Basis pengetahuan ini tersusun atas fakta yang berupa informasi tentang obyek dan kaidah (rule) yang merupakan informasi tentang cara bagaimana membangkitkan fakta baru dari fakta yang sudah diketahui.

Metode kaidah produksi biasanya dituliskan dalam bentuk jika-maka(if-then). Kaidah ini dapat dikatakan sebagai hubungan dua implikasi bagian, yaitu bagian premis (jika) dan bagian konklusi (maka). Apablika bagian premis dipenuhi maka bagian konklusi juga akan bernilai benar. Yang selanjutnya konklusi tersebut ditambahkan kedalam data fakta, yang akan digunakan sebagai informasi selanjutnya.

---

[9]Lauriere, Jean louis,*Problem Solving and Artificial Inteligence,* Prentice Hall International (UK), Great Britain, 1987

Sebuah kaidah terdiri-dari klausa-klausa. Sebuah klausa mirip sebuah kalimat dengan subyek, kata kerja dan obyek yang menyatakan suatu fakta. Ada sebuah klausa premise dan sebuah klausa konklusi pada setiap kaidah. Suatu kaidah juga dapat terdiri dari beberapa premise dan lebih dari satu konklusi.

Contoh :

Rule 1 : IF A AND B THEN C

Rule 2 : IF D AND E THEN F

Rule 3 : IF C AND F AND G THEN H

3.2 Memory Kerja

Bagian penting Sistem berbasis Pengetahuan yang lainya adalah memory kerja, kadang disebut juga sebagai pangkalan data. Semua informasi mengenai fakta-fakta disimpan disini, baik yang merupakan hasil dari input user maupun fakta yang didapat dari hasil konklusi suatu kaidah produksi.

Pada kondisi awal, memory kerja ini isinya kosong, kemudian user diminta untuk memasukkan input awal, yang kemudian digunakan sebgai fakta awal dari sistem. Input awal ini bisa berupa jawabanyang harus diketik, atau mungkin berupa menu pilihan yang harus dipilih oleh user, atau bisa juga berasal dari output suatu perangkat keras yang lain. Dari informasi ini sistem akan mulai melakukan proses pencarian fakta yang lain.

Mesin inferensi mulai melakukan pencarian, mencocokkan kaidah dalam basis pengetahuan terhadap informasi yang ada dalam memory kerja. Apabila informasi yang ada sesuai dengan syarat dalam suatu aturan, maka mesin inferensi akan menambahkan konklusi dari aturan tersebut sebagai fakta baru kedalam memory kerja. Fakta baru yang tersedia digunakan untuk proses

pengambilan keputusan, apabila diperlukan fakta baru untuk pembuktian suatu aturan, mesin bisa meminta user untuk menambahkan informasi yang lain. Memory kerja juga menyimpan data asal dari fakta tersebut, apabila berasal dari suatu aturan produksi, memory kerja akan menyimpan nomor dari aturan tersebut sehingga memungkinkan untuk melakukan pelacakan. Apabila user memerlukan penjelasan tentang proses penalaran.

3.2.3 Mesin Inferensi

Mesin Inferensi merupakan bagian yang mengandung mekanisme fungsi berfikir dan pola-pola penalaran sistem yang digunakan oleh seorang pakar. Bagian inilah yang menghubungkan antara user, basis pengetahuan dan memory kerja.

Secara deduktif , mesin inferensi memilih pengetahuan yang relevan dalam rangka mencapai kesimpulan. Mesin inferensi memulai pelacakannya dengan mencocokkan kaidah-kaidah dalam basis pengetahuan dengan fakta-fakta yang ada dalam memory kerja.

Dalam proses pencarian solusi, ada dua cara yang digunakan oleh mesin inferensi untuk membuktikan kaidah-kaidah yang ada dengan fakta-fakta yang dimilikinya. Cara yang pertama dikenal dengan istilah pencarian maju *(forward chaining)* dan pencarian mundur *(backward chaining)*.

3.3.1 Forward Chainning [10]

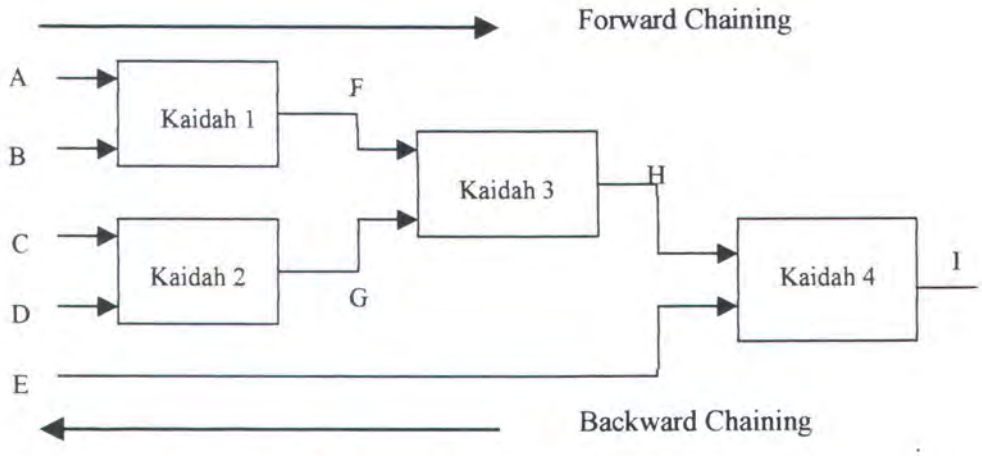
Forward Chaining merupakan strategi pencarian dengan arah maju yaitu dimulai dari sekumpulan fakta-fakta awal kemudian dicocokkan dengan aturan-

---

[10] Durkin, John, *Expert System Design and Development*, macMillan Publishing, New York, 1994

aturan yang ada di basis pengetahuan hingga diperoleh kesimpulan akhir yang paling tepat dengan fakta-fakta dan kaidah-kaidah yang ada atau sudah tidak ada lagi kaidah-kaidah yang sesuai. Gambar 3.2 merupakan diagram proses forward chaining dan backward chaining.

Forward Chaining ini sangat cocok untuk suatu permasalahan yang dimulai dengan informasi, dan kemudian menghasilkan kesimpulan yang logis. Metoda ini sangat cocok untuk problem pemecahan masalah seperti perencanaan, pemantauan, control, dan sebagainya. Sedangkan kekurangan dari metoda ini adalah kadangkala metoda ini menelusuri suatu aturan yang tidak diperlukan, atau menanyakan sesuatu yang tidak berhubungan sama sekali dengan pertanyaan sebelumnya.



Gambar 3.2 Forward Chaining dan Backward Chaining

3.3.2 Backward Chaining[11]

Cara yang kedua disebut dengan penelusuran kebelakang, cara ini dimulai dari suatu kesimpulan, kemudian bergerak maju untuk membuktikan kesimpulan tersebut dengan mengecek informasi yang ada didalam memory

[11]Ibid, Halaman 106

kerja. Apabila data fakta tidak ada, maka akan ditelusuri darimana data tersebut berasal apakah berasal dari input user atau berasal dari konklusi suatu aturan. Bila merupakan konklusi dari suatu aturan akan di proses lagi pembuktian terhadap aturan tersebut, begitu seterusnya sampai kesimpulan akhir tadi terbukti kebenarannya.

Keuntungan dari penggunaan cara ini adalah penelusuran hanya dilakukan pada aturan-aturan yang diperlukan saja. Sehingga cara ini sangat bagus untuk model pemecahan masalah seperti diagnosa, debugging dan lain-lain. Sedangkan kekurangan metoda ini adalah metode ini akan terus menelusuri sampai selesai suatu aturan, meskipun seharusnya sudah menghentikannya atau berpindah ke rule yang lain.

Pemilihan strategi apakah menggunakan forward atau backward chaining ditentukan oleh rancangan dari sistem tersebut. Pemilihan ini juga tergantung kepada masalah yang akan dipecahkan.

Dalam praktek, user kurang memperhatikan penggunaan teknik inferensi apakah sistem menggunakan metoda forward atau backward chaining. Di lain pihak, pembuat sistem pakar harus memperhatikan hal-hal semacam ini.

3.3.3 Certainty Factor [12]

Perhitungan faktor ketidak pastian dalam suatu aturan , memiliki rumus sebagai berikut :

MILIK PERPUSTAKAAN
INSTITUT TEKNOLOGI
SEPULUH - NOPEMBER

Jika Rule memiliki satu premis :

IF E THEN H CF(Rule)

Maka :

$$CF(H,E) := CF(E) * CF(Rule) \quad \textbf{(3.1)}$$

[12] Ibid, halaman 360

Jika Rule memiliki lebih dari satu premis :

IF (E_1 AND E_2 AND E_3 AND ...) THEN H CF(Rule)

Maka:

$$CF(H, E_1 \text{ AND } E_2 \text{ AND } E_3 \text{ AND } ...) = \min \{CF(E_i)\} * CF(Rule) \quad \textbf{(3.2)}$$

Dan

IF (E_1 OR E_2 OR E_3 OR ...) THEN H CF(Rule)

Maka:

$$CF(H, E_1 \text{ OR } E_2 \text{ OR } E_3 \text{ OR } ...) = \max \{CF(E_i)\} * CF(Rule) \quad \textbf{(3.3)}$$

BAB IV

PERANCANGAN DAN PEMBUATAN PERANGKAT LUNAK

BAB IV

PERANCANGAN DAN PEMBUATAN PERANGKAT LUNAK

Dalam bab ini dibahas tentang deskripsi perangkat lunak, perancangan data serta perancangan proses, dan dibahas pula mengenai bagaimana rancangan tersebut dibuat menjadi suatu program.

4.1. DESKRIPSI PERANGKAT LUNAK

Program kamus bahasa Arab yang dibuat untuk Tugas Akhir ini tidak seperti program-program kamus yang lain. Kelebihan dari program ini adalah didalamnya dimasukkan unsur kecerdasan buatan dengan memakai metode Sistem Berbasis Pengetahuan, sehingga dapat menghemat penyimpanan database dan dapat membantu user untuk mencari susunan bentuk dasar suatu kata.

Sebagai suatu program sistem berbasis pengetahuan, program Kamus Bahasa Arab Indonesia ini secara umum terdiri dari tiga bagian yaitu masukan, mesin inferensi, dan penampilan hasil. Untuk menutup kekurangan kosakata juga terdapat modul untuk menambah data kosakata kedalam kamus. Berikut ini dijelaskan secara lebih rinci mengenai bagian-bagian dari perangkat lunak serta tahapan-tahapan prosesnya mulai dari input sampai didapatkan hasil.

4.1.1 Masukan

Dalam program tugas akhir ini, user memberikan input berupa susunan karakter Arab yang disertai dengan harokatnya. Pemberian harokat ini sangat penting karena perbedaan harokat bisa menyebabkan perbedaan pada artinya.

Akan tetapi user juga masih mungkin untuk memasukkan input kata yang tidak disertai harokat, karena program ini akan mencarikan alternatif-alternatif harokat yang sesuai.

Karakter-karakter input tersebut selanjutnya diproses menjadi token-token premis, yang kemudian token-token tersebut merupakan informasi yang akan digunakan untuk pencarian susunan asal kata.

4.1.2 Mesin Inferensi

Dalam suatu program Sistem Berbasis Pengetahuan mesin inferensi merupakan bagian yang melakukan proses terhadap data masukan sehingga didapatkan suatu kesimpulan yang sesuai. Proses yang ada di dalam mesin inferensi ini sangat ditentukan oleh aturan-aturan yang ada di dalam basis pengetahuan, karena pemrosesan dilakukan dengan mengikuti aturan-aturan yang ada di dalam basis pengetahuan.

Dalam program tugas akhir ini data aturan-aturan disimpan di dalam suatu file bertipe text yang otomatis dibaca ketika program dijalankan. Dengan menyimpan data aturan-aturan yang terpisah dengan program, memungkinkan terjadinya penambahan dan perubahan terhadap aturan-aturan yang ada tanpa harus mengkompile ulang program.

Metoda penelusuran yang digunakan dalam program Kamus Bahasa Arab Indonesia ini diawali dengan menggunakan metoda penelusuran maju atau biasa dikenal dengan istilah *forward chaining* kemudian apabila diperlukan pengujian terhadap kebenaran suatu rule secara otomatis akan digunakan metoda penelusuran mundur atau biasa dikenal dengan istilah *backward chaining.* Metoda ini prosesnya berawal dari pengecekan kebenaran terhadap

aturan yang pertama, apabila aturan ini bernilai benar, secara otomatis data konklusi dari aturan ini akan dimasukkan kedalam memory kerja sehingga data informasi akan bertambah, kemudian dilanjutkan dengan melakukan pengecekan terhadap aturan yang berhubungan dengan data konklusi tadi. Sampai akhirnya didapatkan suatu kesimpulan.

Mesin inferensi yang dibuat untuk program tugas akhir ini juga dilengkapi dengan kemampuan untuk mengetahui level kebenaran suatu data informasi, yang biasa dikenal dengan istilah *certainty factor.* Dengan adanya kemampuan ini memungkinkan untuk mendapatkan suatu kesimpulan dari data informasi yang kurang lengkap, dalam hal ini penulisan tanda baca *(harokat).* Sehingga apabila user memasukan data susunan karakter arab yang tidak menggunakan harokat, masih memungkinkan untuk diperoleh suatu kesimpulan terhadap pola dari susunan karakter input tersebut.

Selain basis pengetahuan, mesin inferensi juga berhubungan dengan memory kerja yang berfungsi untuk menyimpan data-data informasi selama proses penelusuran, hingga didapatkan suatu kesimpulan.

Hasil output dari mesin inferensi ini adalah jenis pola *(wazan)*, bentuk susunan kata *(bina'),* posisi huruf asal dan subyek *(fail)* serta obyek *(maful)* yang menyertai apabila kata berupa suatu kata kerja yang disertai subyek dan obyek.

4.1.3 Pencarian Arti

Setelah didapatkan kesimpulan berupa jenis pola (wazan), bentuk susunan kata (bina'), posisi huruf asal dan subyek serta obyek yang menyertai kata. Proses selanjutnya adalah melakukan penyusunan huruf-huruf asal yang diambil dari data masukan yang telah diketahui posisi huruf asalnya, selanjutnya

barulah dilakukan pencarian arti kata dalam basis data. Dalam hal ini dilakukan dua kali pencarian arti, yang pertama pencarian arti kata dari asal kata, dan yang kedua pencarian arti yang didapat dari wazan.

4.1.4 Luaran dan Penampilan Hasil

Setelah arti kata dan makna wazan didapatkan, proses selanjutnya adalah penampilan hasil berupa arti dari kata masukan, yang meliputi arti dari asal kata dan arti dari wazan sedangkan informasi lainnya adalah bentuk susunan kata (bina'), serta subyek dan obyek dari kata tersebut.

4.1.5 Modul Tambahan

Modul tambahan dari program ini berupa modul untuk melakukan penambahan atau perubahan data kosakata kamus, juga untuk melakukan penambahan atau perubahan untuk data kosakata wazan.

4.2 PERANCANGAN DATA

Berikut ini dijelaskan mengenai perancangan data yang meliputi data masukan, data mesin inferensi, dan data keluaran.

4.2.1 Data Masukan

Data masukan untuk program ini berupa susunan huruf arab yang diketikkan langsung menggunakan keyboard. Sebelum diproses, data masukan ini disimpan kedalam suatu variable bertipe string kemudian dilakukan proses scaning terhadap data masukan. Proses ini berfungsi untuk mengecek jika terdapat kesalahan dalam data masukan, serta untuk merubah data masukan

menjadi token-token yang digunakan untuk penelusuran pola kata. Token-token data tersebut disimpan kedalam memory kerja sebagai informasi awal untuk basis pengetahuan dalam melakukan pencarian pola kata.

Token-token input terdiri dari token huruf, harokat dan jumlah huruf. Token Huruf merupakan token yang berisi informasi mengenai huruf-huruf yang diinputkan oleh user, mulai dari huruf yang pertama hingga huruf yang terakhir. Harokat merupakan token yang berisi informasi mengenai tanda baca dari huruf-huruf tersebut, sedangkan jumlah huruf merupakan informasi mengenai jumlah dari huruf-huruf yang diinputkan. Format penulisan dari token-token ini disesuaikan dengan aturan-aturan yang ada di basis pengetahuan. Berikut ini tabel transliterasi dari huruf-huruf tersebut menjadi token-token.

Tabel 4.1 Transliterasi Huruf Arab

Huruf Latin	Huruf Arab		Huruf Latin	Huruf Arab
Fa	ف		Alif	ا
Qof	ق		Ba	ب
Kaf	ك		Ta	ت
Lam	ل		Tsa	ث
Mim	م		Jim	ج
Nun	ن		Kha	ح
Wau	و		Kho	خ
Ha	ه		Dal	د
Hamzah	ء		Dzal	ذ
Ya	ي		Ro	ر
Ta marbuto	ة		Za	ز
Alif Ya	ى		Sin	س
Fathah	َـ		Syin	ش

Fathah tain	ـً		Shad	ص
Kasrah	ـِ		Dhad	ض
Kasrah tain	ـٍ		Tha	ط
Dhammah	ـُ		Zha	ظ
Dhammah tain	ـٌ		Ain	ع
Tasydid	ـّ		Ghain	غ
Sukun	ـْ			

4.2.2 Data Mesin Inferensi

Pada mesin inferensi data-data penting yang berkaitan dengan proses adalah kaidah-kaidah basis pengetahuan dan data-data fakta yang disimpan dalam memory kerja.

Data-data untuk menyimpan kaidah-kaidah basis pengetahuan adalah sebagai berikut :

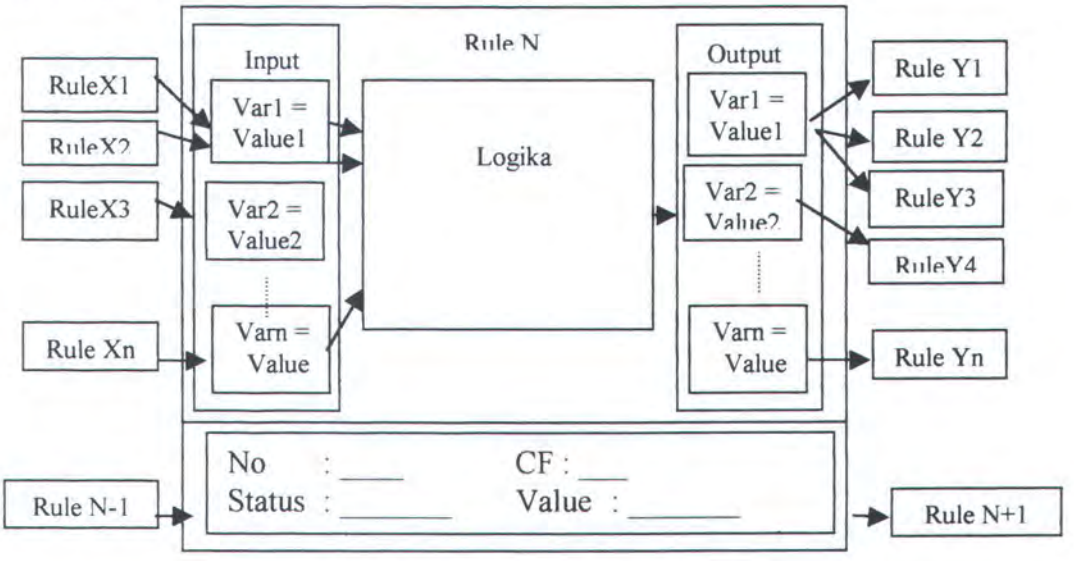
- PLogic adalah variabel bertipe pointer yang menunjuk ke suatu variabel rekaman *(record)*. Variabel ini berguna untuk menyimpan struktur logika dari kaidah basis pengetahuan. Variabel rekaman tersebut berisi data-data sebagai berikut :
  - Input : PNodeLogic; { menyimpan list data input }
  - Parent : PLogic; { variabel untuk menunjuk diatasnya}
  - InputRef : PVarList; { referensi ke pointer Variable input}
  - LogicType : Byte; { jenis gerbang logika (and/or) }
  - Value : integer; { hasil output dalam hal ini outputnya –1,0 atau 1}
  - cecked :boolean; { sebagai tanda apakah sudah pernah diproses/belum}
  - next : PLogic; { menunjuk ke gerbang logika yang lain }

- PVarList adalah variabel bertipe pointer yang berfungsi untuk menyimpan data-data variable premis dan konklusi dari suatu Rule /kaidah. Rekaman yang ditunjuk oleh variabel ini adalah:
  - RuleRef : PRuleList; {menyimpan hubungan dengan Rule Asal / tujuan}
  - VarName : medstr; { Nama Variabel premis}
  - Sign : shortstr; { Tanda (=,<,>,<=,>=) }
  - VarValue : medstr; { isi variabel }
  - next : PVarList; { menunjuk ke data selanjutnya }
- PNodeLogic adalah suatu variabel pointer untuk menyimpan list dari Gerbang logika.
- PruleType adalah variabel yang menunjuk ke sutu rekaman yang menyimpan semua informasi dari suatu kaidah. Isi dari variabel ini adalah sebagai berikut :
  - Input : PVarList; { berisikan list dari variabel-variabel premis }
  - Output : PVarList; { berisikan list dari variabel-variabel konklusi }
  - status : Boolean; { sebagai tanda apakah kaidah ini telah diproses / belum}
  - value : integer; { derajat kebenaran eksekusi rule nilainya –1,0,1}
  - No : Word; { No Rule }
  - LogicInput : Plogic; { Menyimpan struktur gerbang logika dari Input }
  - next : PRuleType; { Untuk menunjuk rule setelahnya }
- PRulelist adalah variabel yang digunakan untuk menyimpan hubungan antara rule yang satu dengan rule yang lainnya. Isi dari variabel ini adalah
  - Item : PRuleType; { menunjuk ke rekaman Rule }

- status : boolean; { menyimpan status hubungan antar Rule (terhubung/putus}
- next : PRulelist; { untuk menunjuk data selanjutnya}

- PPremise adalah variabel yang berguna untuk menyimpan data-data fakta yang telah diperoleh selama proses, baik data fakta yang berasal dari input maupun data fakta yang diperoleh dari eksekusi rule. Sebagai variabel memory kerja, isi dari variabel ini adalah :
  - No : word; {menyimpan no asal dari rule yang menghasilkan premis ini}
  - VarName : MedStr; {nama variabel }
  - Sign : shortstr; {tanda (=,<,>,<=,>=,<>)}
  - VarValue : MedStr; {nilai variable }
  - CF : Integer; {Certainty Factor}
  - next : PPremise; {menunjuk data selanjutnya}

Struktur objek apabila digambarkan kedalam suatu diagram menjadi seperti dibawah ini :



Gambar 4.1 Struktur Objek Rule

Selain variabel-variabel diatas juga diperlukan variabel-variabel untuk menyimpan status-status proses. Seperti variabel untuk menunjuk rule awal dan rule akhir, variabel yang menunjuk ke memory kerja, dan sebagainya.

4.2.3 Data Keluaran

Data keluaran dari program kamus Bahasa Arab-Indonesia ini adalah arti kata, menurut wazan yang sesuai. Dalam hal ini memungkinkan diperoleh data kesimpulan yang lebih dari satu. Untuk itu variabel yang digunakan untuk menyimpan hasil akhir ini berupa suatu variabel rekaman *(record)*, yang disimpan kedalam suatu linked list kemudian data tersebut akan ditampilkan satu persatu. Data yang ditampilkan meliputi :

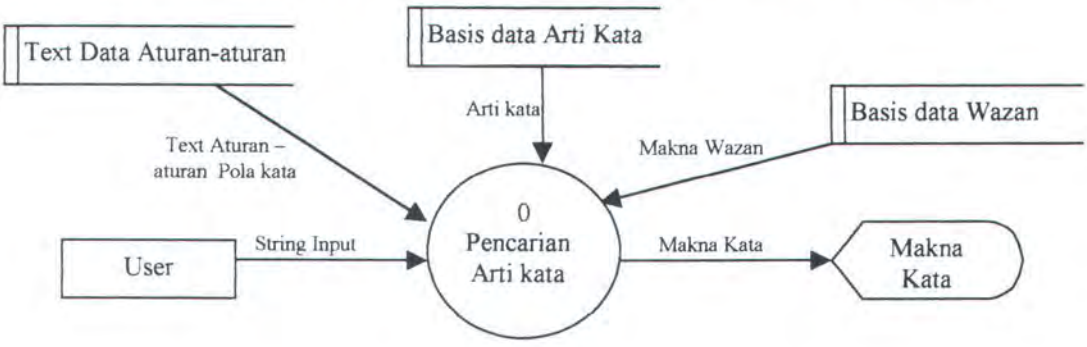
- Jenis Kata, menjelaskan tentang jenis kata input.
- Wazan, menjelaskan tentang wazan input.
- Kata dasar, menjelaskan kata dasar dari kata inout yang disimpan di dalam kamus.
- Arti kata, menjelaskan arti kata dasar
- Makna tambahan, menjelaskan makna tambahan yang diperoleh dari wazan.
- Fail, menjelaskan tentang pelaku/subyek, apabila kata input berupa suatu kata kerja
- Maful, menjelaskan tentang obyek apabila dalam kata input juga disertai obyek.

4.3 PEMBUATAN PERANGKAT LUNAK

Sebagai suatu program sistem berbasis pengetahuan, proses utama yang ada dalam program ini adalah pencarian konklusi dalam basis pengetahuan,. Dengan kata lain, algoritma penelusuran asal kata terpisah dengan program. Sebelum dibahas tentang pembuatan modul-modul yang ada dalam perangkat lunak, akan dibahas terlebih dahulu proses yang ada dalam perangkat lunak tersebut yang digambarkan dengan sutu diagram alir data.

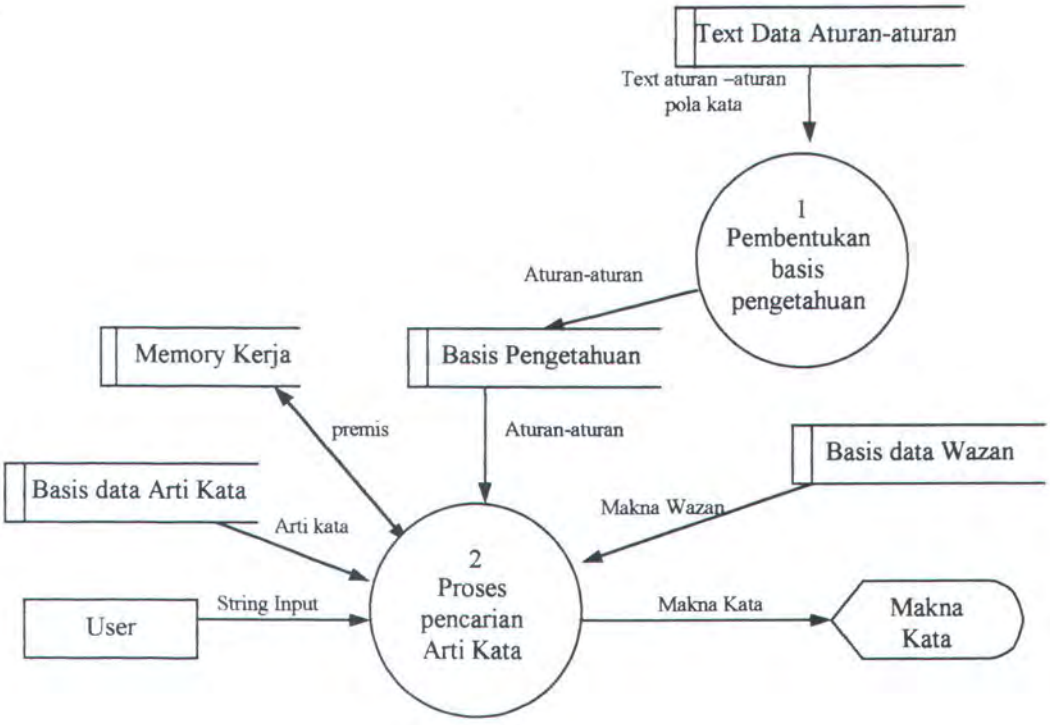
4.3.1 Diagram Alir Data

Diagram alir data merupakan teknik grafis untuk menggambarkan aliran informasi dari data masukan sampai data keluaran. Gambar 4.2 menampilkan diagram alir data level 0 untuk aplikasi Kamus Bahasa Arab Indonesia ini.



Gambar. 4.2 Diagram Alir Data level 0 Aplikasi Kamus Bahasa Arab Indonesia.

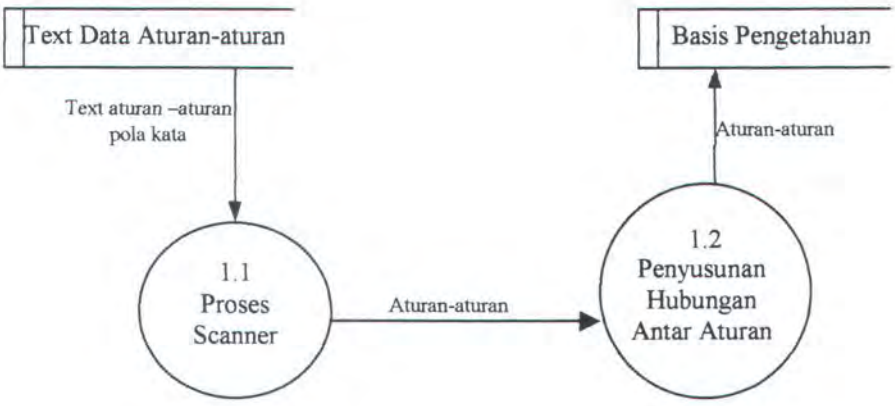
Diagram alir data diatas merupakan gambaran secara umum dari semua proses yang ada dalam aplikasi. Untuk lebih memperjelas proses yang ada, diagram alir data diatas akan dijabarkan ke level-level yang lebih terperinci seperti dibawah ini.



Gambar 4.3 Diagram Alir Data level 1 dari Aplikasi Kamus Bahasa Arab-Indonesia

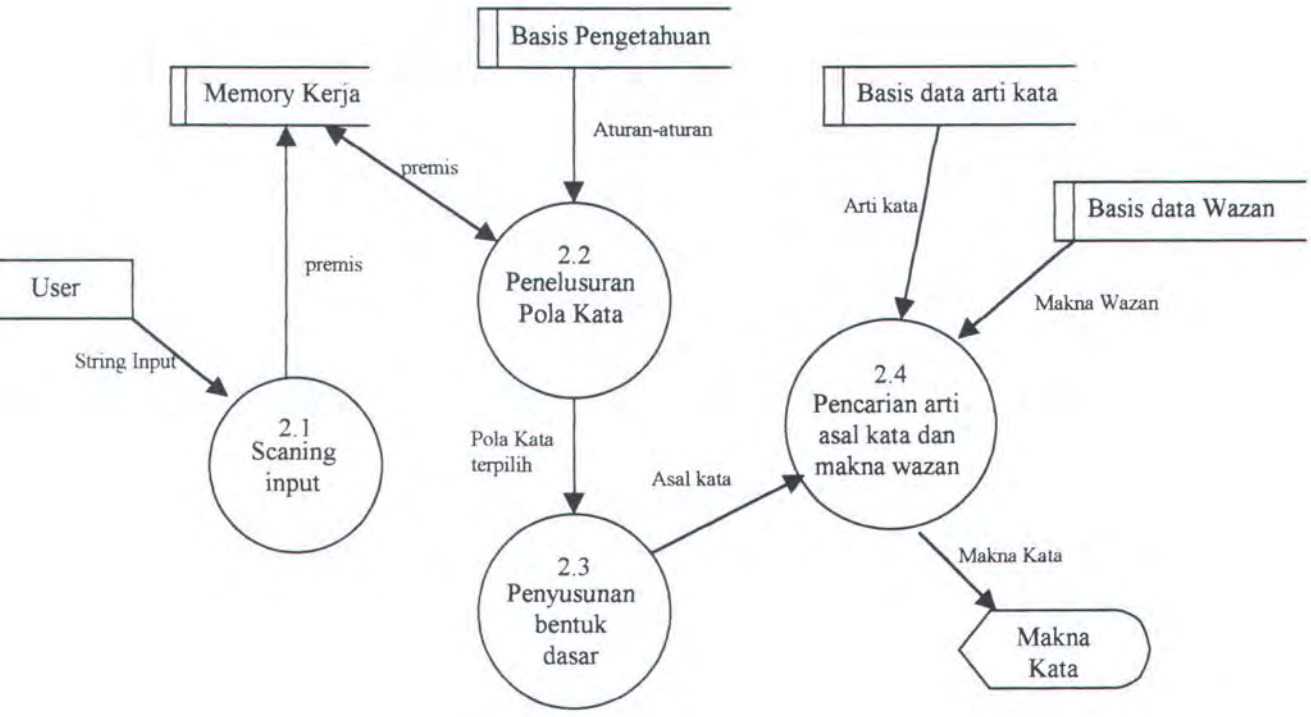
Dari diagram diatas dapat proses-proses yang ada dalam perangkat lunak dan aliran-aliran datanya. Secara umum proses yang ada adalah proses pembentukan basis pengetahuan dan proses untuk pencarian arti kata. Dalam proses pembentukan basis pengetahuan, data masukan berupa data aturan-aturan yang disimpan di dalam file text dan menghasilkan keluaran berupa aturan-aturan yang disimpan dalam memory. Sedangkan proses pencarian arti kata mendapatkan data masukkan dari user dan basis data. Untuk lebih jelasnya bisa dilihat pada diagram alir level 2.

MILIK PERPUSTAKAAN
INSTITUT TEKNOLOGI
SEPULUH - NOPEMBER



Gambar 4.4 Diagram Alir Data level 2. Proses Pembentukan Basis Pengetahuan

Pada diagram alir data level 2, Proses Pembentukan Basis Pengetahuan diatas bisa dilihat secara lebih detil aliran-aliran data yang ada. Untuk pertama kali proses yang dilakukan adalah proses scanning input text data aturan-aturan. Pada proses ini dilakukan pembacaan terhadap dokumen aturan-aturan, kemudian aturan-aturan ini disimpan kedalam memory. Setelah didapatkan data aturan-aturan yang telah disimpan di dalam memory, kemudian dilakukan proses penyusunan hubungan antar aturan. Proses ini berguna untuk mempercepat proses *forward chaining* dan *backward chaining* pada mesin inferensi. Hasil akhir dari proses pembentukan basis pengetahuan diatas adalah basis pengetahuan sesuai dengan aturan-aturan yang diperlukan.



Gambar 4.5 Diagram Alir Data level 2. Proses Pencarian Arti Kata

Pada proses pencarian arti dibagi menjadi empat proses yang lebih kecil, yaitu : scanning input, penelusuran pola kata, penyusunan bentuk dasar, dan yang terakhir adalah pencarian arti dari asal kata dan makna wazan. Proses scanning input mendapatkan masukan berupa string input dari user. Data string input ini kemudian diproses menjadi token-token premis sekaligus dilakukan pengecekan apakah ada kesalahan terhadap data input. Pada proses yang kedua dilakukan proses penelusuran pola kata, pada proses ini menggunakan metoda forward chaining yang menghasilkan kesimpulan akhir berupa pola-pola kata yang sesuai. Dari pola kata ini bisa diketahui posisi huruf-huruf asalnya yang kemudian menjadi input proses selanjutnya, selain itu juga didapatkan beberapa informasi penting yang lain seperti jenis kata, subyek, dan obyek serta bina' . Proses selanjutnya adalah membentuk bentuk dasar dari informasi yang telah

didapatkan yang kemudian dari bentuk dasar ini baru bisa dicari artinya untuk kemudian ditampilkan.

4.3.1 Pembuatan Modul

Dibawah ini dibahas mengenai pembuatan modul Mesin inferensi. Modul mesin inferensi merupakan modul yang paling utama dalam program kamus bahasa Arab Indonesia. Karena didalam modul mesin inferensi ini semua proses dihubungkan. Struktur data untuk menyimpan basis pengetahuan mempunyai tipe data seperti di bawah ini::

```
// Untuk menyimpan data logika dari Rule
PLogic      = ^Logic;
Logic      = record
             Input      : PNodeLogic; // list input
             Parent     : PLogic;
             InputRef   : PVarList; // Variable
             LogicType  : Byte;     // type of logic (and/or)
             Value      : integer;  // value (-1,0,1)
             cecked     : boolean;
             next       : PLogic;
           end;

// Menyimpan Variabel premis input dan konklusi
PVarList    = ^VarList;
VarList   = record
             RuleRef : PRuleList;     // Rule Asal / tujuan
             VarName : medstr;        // Nama Variabel
             Sign    : shortstr;      // Tanda (=,<,>,<=,>=)
             VarValue : medstr;       // isi variabel
             next    : PVarList;
           end;

// Menyimpan Node-Node Rule
PRuleType   = ^RuleType;
RuleType    = record
             Input    : PVarList;     // list input
             Output   : PVarList;     // list output
             status   : Boolean;      // status Rule
             value    : integer;      // -1,0,1
             No       : Word;         // No Rule
             CF       : integer;      // Certainty factor
             LogicInput : Plogic;     // Operasi logika
             next     : PRuleType;
           end;

// Untuk menyimpan list gerbang
```

```
PNodeLogic = ^NodeLogic;
NodeLogic = record
            Item : PLogic;
            next : PNodeLogic;
          end;

// Untuk menyimpan list Node Rule
PRulelist  = ^Rulelist;
RuleList = record
           Item : PRuleType;
           status : boolean;
           next : PRulelist;
         end;

// Sebagai memory kerja
PPremise = ^TPremise;
TPremise = record
            No       : word;
            VarName  : MedStr;
            Sign     : shortstr;
            VarValue : MedStr;
            CF       : Integer;
            next     : PPremise;
          end;
```

Untuk memudahkan mengelola data-data diatas dibuat suatu kelas dengan nama NewChain, struktur kelas tersebut adalah :

```
TNewChain = class
  private // Private Declaration
   akhir : boolean;
   HeadRule : PRuleType;
   TailRule : PRuleType;
   HeadPrem : PPremise;
   TailPrem : PPremise;
   TextInput : string;
   NewTextInput :string;
   TextHuruf : string;
   CancelProcess : boolean;
   FChainStatus : Boolean;
   RuleExecute : PRuletype;
   Procedure AddNodeLogic(var VarLogic :Plogic;var Item:PLogic);
   Procedure AddLogic(var Head : Plogic;LogicType : Byte;var
         VarRef :PVarlist);
   Function AddVarList(var Head: PVarlist; VarName: medstr;
            Sign : ShortStr; Value: MedStr; QId :Word):PVarlist;
   Procedure AddRuleRef(var RuleRef:PruleType;var VarRef :
         PVarList);
   Procedure ClearLogic(var NodeLogic :PLogic);
   Procedure ClearVarList(var NodeVarList : PVarList);
   Procedure ClearNodeLogic(var Node : PNodeLogic);
   Procedure ClearPremis;
   Procedure CreateConnection(var RuleA,RuleB : PRuleType);
```

```
  function Cekvalue(VarRef: PVarList;value : medstr):integer;
  Procedure DestroyAll;
  function ExecuteLogic(var LogicRef : PLogic;No:word):integer;
  Function ExecuteRule(var RuleRef : PRuleType):integer;
  procedure InitRule;
  Procedure MakeRuleConnection;
  Procedure MakeLogicConnect(var RuleREf : PRuleType);
  function premis_ada(var_name : medstr;var value : medstr;var
        sign: shortstr):PPremise;
  procedure setharokat(index : integer;harokat :byte);
  Procedure SortVarlist(var RuleREf : PRuleType);
  Procedure Telusur(hnd : pruletype);
  function asal_kata(inputpos,textasal : string):string;
 public // Public Declaration
  constructor Create;
  destructor destroy;override;
  Procedure AddPremise(No : word;N_IF,VIF : medstr;Sgn :
        shortstr;CF :Integer);
  Procedure ForwardChaining;
  Procedure How(hnd : pruletype);
  procedure GetInput(StrInput : string);
  Procedure LoadRule(FileName : string);
  property RulePosition : PRuletype read RuleExecute;
  Property ChainStatus : Boolean read FChainstatus;
  property stringinput : string read texthuruf;
  procedure tampilkan;
  procedure tampilPremis;
 Protected

 Published

end;
```

Prosedur-prosedur penting yang berkaitan langsung dengan proses pembangunan basis pengetahuan adalah prosedur LoadRule dan MakeRuleConnection;

Sedangkan prosedur yang berkaitan dengan proses pencarian arti kata adalah Prosedur GetInput, Fungsi asal_kata, dan prosedur forwardchaining. Untuk proses penampilan hasil dibuatkan kelas tersendiri dengan nama kelas PMakna.

Berikut ini listing dari prosedur prosedur diatas.

Prosedur LoadRule;

```
Procedure TNewChain.LoadRule(FileName : string);
var FileInput : textfile;
    strbuff : string;
    tokenIf,
    tokenvalue,tmptoken : medstr;
    tokensign : shortstr;
    tmpRule : PRuletype;
    tmpVarlist : PVarlist;
    statusRule : boolean; { 0 new rule,1 continued rule}
    tmpString : string;
    RuleNo : word;
    Posisi,i : byte;
    flag : boolean;

 Procedure ResetTmpVar;
 begin
  tokenif   := '';
  tokensign := '';
  tokenvalue:= '';
  tmptoken  := '';
 end;

 Procedure AddRule(strRule : string);
 Begin
  i :=1;
  statusrule:= false;
  if strRule<>'' then
  begin
   tmpString := strRule;
   if (tmpstring[1]<> '_') then
   begin
    statusrule := true;
    flag := false;
    posisi := Pos(':',tmpString);
    try
     RuleNo := strtoint(copy(tmpString,1,Posisi-1));
    except
     on EConvertError do
     begin
       MessageDlG('Error  Pada  Penulisan  Nomor Rule', mtWarning,
[mbOK], 0);
       exit;
     end;
    end; //end exception
   ///////          init            //////
   new(tmpRule);
   tmpRule^.No := RuleNo;
   tmpRule^.Input := nil;
   tmpRule^.OutPut := nil;
   tmpRule^.status := false;
   tmpRule^.value := -1;
   tmpRule^.next := nil;
```

```
tmpRule^.LogicInput := nil;
//////
Delete(tmpString,1,Posisi);
end else Delete(tmpString,1,1);
ResetTmpVar; // Set Temporary variable = ''
While i<=length(tmpString) do
begin
  case tmpstring[i] of
    '0'..'9','_','+','-','/','*','\','$','@','~','#','(',')',
     '%','<','>','=','a'..'z','A'..'Z','.','&','{','}','`':
        begin
        tmptoken := tmpToken + tmpString[i];
        if not flag then
        begin
          if UpperCase(tmpToken) ='IF' then tmpToken := '';
          if UpperCase(tmpToken) ='AND' then
          begin
            if (tokenif <>'') and (tokensign<> '') and
                    (tokenvalue <>'') then
            begin
             tmpVarList:=AddVarList(tmpRule^.Input,tokenIf,
                  tokenSign,tokenValue,1);
             AddLogic(tmpRule^.LogicInput,0,tmpVarList);
             ResetTmpVar; // Set Temporary variable = ''
             tmpVarList := nil;
             // And type
             AddLogic(tmpRule^.LogicInput,1,tmpVarList);
            end else begin
                      MessageDlg('Aturan Penulisan Variabel
                          Pada Rule No:'+ InttoStr(RuleNo)+'
                          Salah', mtWarning, [mbOK], 0);
                      exit;
                    end
          end
          else if UpperCase(tmpToken) ='OR' then
          begin
            if (tokenif <>'') and (tokensign<> '') and
                   (tokenvalue <>'') then
            begin
            tmpVarList:= AddVarList(tmpRule^.Input,tokenif,
                   tokenSign,tokenValue,1);
            AddLogic(tmpRule^.LogicInput,0,tmpVarList);
            ResetTmpVar; // Set Temporary variable = ''
            tmpVarList := nil;
            // Or Type
            AddLogic(tmpRule^.LogicInput,2,tmpVarList);
          end else begin
                      MessageDlg('Aturan Penulisan Variabel
                           Pada Rule No:'+ InttoStr(RuleNo)+'
                           Salah', mtWarning, [mbOK], 0);
                      exit;

           end
         end else if UpperCase(tmpToken)='THEN' then
            begin
              if (tokenif <>'') and (tokensign<> '') and
```

MILIK PERPUSTAKAAN
INSTITUT TEKNOLOGI
SEPULUH - NOPEMBER

```
            (tokenvalue <>'') then
        begin
          tmpVarList:=AddVarList(tmpRule^.Input,
                    tokenif,tokenSign, tokenValue,1);
          AddLogic(tmpRule^.LogicInput,0,tmpVarList);
          ResetTmpVar; // Set Temporary variable = ''
        end else begin
                  MessageDlg('Aturan Penulisan Variabel
                        Pada Rule No:'+ InttoStr(RuleNo)+'
                        Salah', mtWarning, [mbOK], 0);
                  exit;
                end
        end
    end else
    IF tmpString[i] in ['=','<','>'] then
      begin
        MessageDlg('Aturan Penulisan Variabel Pada Rule
                        No:'+ InttoStr(RuleNo)+' Salah',
                        mtWarning, [mbOK], 0);
        exit;
      end

    end;
'"' : begin
       if flag then
       begin
          if TokenIf='' then tokenIf := tmpToken else
          if TokenValue='' then tokenvalue := tmpToken;
       end else begin
                  if (tmptoken = '=') or (tmptoken = '<=')
                    or (tmptoken = '>=')or(tmptoken = '<')
                    or (tmptoken = '>')or(tmptoken = '<>')
                  then  tokensign:= tmptoken;
                end;
       tmptoken :='';
       flag := not flag;
      end;
',' : begin
        if flag then tmptoken := tmpToken + tmpString[i]
        else begin
           if (tokenIf<>'') and (tokenSign<>'') and
               (tokenValue<>'')
           then
                AddVarList(tmpRule^.Output,tokenif,
                tokenSign,tokenValue,1);
                tokenif := '';
                tokensign := '';
                tokenValue := '';
                tmptoken :='';
        end;
      end;
';' : begin
        if flag then tmptoken := tmpToken + tmpString[i]
        else begin
        if TokenValue='' then tokenvalue := tmpToken;
        if (tokenIf<>'') and (tokenSign<>'') and
```

```
                (tokenValue<>'')
            then
                AddVarList(tmpRule^.Output,tokenif,
                       tokenSign,tokenValue,1);
                tokenif := '';
                tokensign := '';
                tokenValue := '';
            end;
          end;
      ' ' : begin
            if (tmptoken = '=') or (tmptoken = '<=') or
               (tmptoken = '>=') or (tmptoken = '<') or
               (tmptoken = '>') or (tmptoken = '<>')
            then
              tokensign:= tmptoken;
              if flag then tmptoken := tmptoken + tmpstring[i]
               else tmptoken := '';
          end;
    else
     MessageDlg('Invalid Karakter pada rule No
          '+inttostr(RuleNo),MtWarning,[mbOk],1);
    end;
    inc(i);
  end;
  if statusrule then
  begin
    if headRule = nil then
    begin
     HeadRule := TmpRule;
     TailRule := tmpRule;
    end else begin
            TailRule^.next := tmpRule;
            tailRule := tailRule^.next;
          end;
  end;
 end
End;

Begin
 DestroyAll;
 assignfile(FileInput,Filename);
 try
 reset(FileInput);
 except
 end;
 while not eof(FileInput) do
 begin
  readln(FileInput,Strbuff);
  if (StrBuff[1]<>'!') and (StrBuff<>'')then AddRule(StrBuff);
 end;
 Closefile(FileInput);
 MakeRuleConnection;
End;
```

Prosedur MakeRuleConnection;

```
Procedure TNewChain.MakeRuleConnection;
var tmpRule1,tmpRule2 : PRuleType;
begin
 tmpRule1 := HeadRule;
 while tmpRule1<>nil do
 begin
   SortVarList(tmpRule1);
   tmpRule2 := tmpRule1^.next;
   while tmpRule2 <> nil do
   begin
     CreateConnection(TmpRule1,TmpRule2);
     CreateConnection(TmpRule2,TmpRule1);
     tmpRule2 := tmpRule2^.next;
   end;
   MakeLogicConnect(tmpRule1);
   tmpRule1 := tmpRule1^.next;
 end;
end;
```

Procedure GetInput

```
procedure TNewChain.GetInput(StrInput : string);
var i : byte;
    hurufke : byte;
    tmp : byte;
begin
  ClearPremis;
  TextInput := strinput;
  texthuruf := '';
  hurufke := 1;
  tmp := length(StrInput);
  i:= 1;
  while i<=tmp do
  begin
    case Ord(StrInput[i]) of
     199 : begin
             Addpremise(0,'huruf-'+inttostr(hurufke),'alif','=',1);
             inc(hurufke);
             texthuruf := texthuruf + StrInput[i];
           end;
     200 : begin
             Addpremise(0,'huruf-'+inttostr(hurufke),'ba','=',1);
             inc(hurufke);
             texthuruf := texthuruf + StrInput[i];
           end;
............
............
............

     242 : begin
             Addpremise(0,'harokat-'+inttostr(hurufke-1),
             'kasroh tain','=',1);
           end;
```

```
    250 : begin
              Addpremise(0,'harokat-'+inttostr(hurufke-1),
              'sukun','=',1);
          end;
    else begin
          MessageDlg('Input karakter ke-'+inttostr(i)+'
          salah',mtWarning,[mbOk],0);
         end;
   end;
   inc(i);
  end;
  Addpremise(0,'jumlah',inttostr(hurufke-1),'=',1);
end;
```

Fungsi Asal Kata

```
function TNewChain.asal_kata(inputpos,textasal : string):string;
var// tmpinput : string;
     posisi : array [1..10] of byte;
     jml_huruf : byte;
     index,i : byte;
     tmpstr : string;
     tmpresult : string;
begin
 if (inputpos <> '') and (textasal <> '') then
  begin
    index := 1;
    i := 1;
    tmpstr := '';
    while i < length(inputpos)+1 do
    begin
      case inputpos[i] of
      '0'..'9' : tmpstr := tmpstr + inputpos[i];
      '-' : begin
              if tmpstr<>'' then
              begin
               posisi[index] := strtoint(tmpstr);
               tmpstr := '';
               inc(index);
              end;
            end;
       'a','w','y','Y','h','s' :
            begin
              Posisi[index] := Ord(InputPos[i]);
              tmpstr := '';
              inc(index);
            end;
      end;
      inc(i);
    end;

    jml_huruf := index-1;

    tmpresult := '';
    for i:= 1 to Jml_huruf do
    begin
```

```
    if posisi[i] > 50 then
    begin
      case Chr(posisi[i]) of
        'a' : tmpresult := tmpresult + Chr(199);
        'w' : tmpresult := tmpresult + Chr(230);
        'y' : tmpresult := tmpresult + Chr(237);
        'Y' : tmpresult := tmpresult + Chr(236);
        'h' : tmpresult := tmpresult + Chr(193);
        's' : tmpresult := tmpresult + Chr(248);
      end;
    end
    else
     tmpresult := tmpresult + textasal[posisi[i]];
   end;
   Result := tmpresult;
  end;
end;
```

Prosedur Forward Chaining

```
Procedure TNewChain.ForwardChaining;
 function findValidRule:PRuletype;
 var handleRule : PRuletype;
     ketemu : boolean;
 begin
  { mencari rule yang belum dieksekusi
    proses dimulai dengan pengecekan rule yang pertama sampai rule
       yang terakhir }
   handleRule := HeadRule;
  ketemu := false;
  while (handleRule <> nil) and (not ketemu) do
  begin
   if handleRule^.status then
   begin
     ketemu := true;
     result := handleRule;
   end
   else handleRule := handleRule^.next;
  end;
  if not ketemu then result := nil;
 end;

var tmpRuleType : PRuleType;
    handle1     : PRuleType;
    HandleList : PRuleList;
begin
if HeadRule = nil then MessageDlg('Rule Tidak
       Ada',mtError,[mbOk],0)
else
begin
  FChainStatus := C_Forward;
  FormArti.MaknaKata.init;
  tmpRuleType := HeadRule;
  akhir := false;
```

```
  initRule;
  tmpRuleType := findvalidRule;
  while (tmpRuleType <>nil) and (not akhir) and (not
       cancelProcess)do
  begin
   if tmpRuleType^.status then
   begin
    handle1 := tmpRuleType;
     while (executeRule(Handle1) > -1) and (not akhir) and
           (Handle1^.Output^.RuleRef<> nil)  do      // Proses
       terus kekanan
     begin
       if Handle1^.Output^.RuleRef<> nil then
       begin
         handlelist := Handle1^.Output^.RuleRef;
         if  handlelist^.status then handle1 := handlelist^.Item;
       end
     end;
 end;
 tmpRuleType := findvalidRule; /// Proses diulang
end;
if FormArti.MaknaKata.Count <> 0 then
begin
 FormArti.MaknaKata.first;
 FormArti.Show;
end else
 begin
   MessageDlg('Kata     dan     pola     tidak     terdapat     dalam
kamus',mtInformation,[mbOK],0);
 end;
end;
end;
```

Sedangkan modul untuk menampilkan arti kata mempunyai kelas seperti dibawah ini :

```
PMakna = ^Makna;
  Makna = Record
           RuleRef   : Pointer;
           Jenis     : string[50];
           textInput : string[30];
           wazan     : string[20];
           bina      : string[20];
           asal      : string[20];
           fail      : string[50];
           maful     : string[50];
          end;

  TMakna = Class
   private
    Items : TList;
    FIndex : Integer;
    FCount : Integer;
    Item  : PMakna;
```

```
public
 Jenis    : string[50];
 textInput : string[30];
 Ruleref : Pointer;
 wazan    : string[20];
 bina     : string[20];
 asal     : string[20];
 fail     : string[50];
 maful    : string[50];
 constructor create;
 destructor destroy;
 procedure init;
 procedure add(ref:Pointer; sjenis, stextinput, swazan, sbina,
      sasal, sfail, smaful:string);
 procedure first;
 procedure last;
 procedure next;
 procedure prev;
 property count :integer read FCount;
 Property index : integer read FIndex;
end;
```

Apabila bentuk dasar ditemukan, akan tetapi kosakata tidak ditemukan dalam kamus, otomatis akan ditanyakan apakah kata tersebut akan ditambahkan ke kamus, apabila dijawab ya. Selanjutnya akan ditampilkan form untuk menambah data kosakata.

4.3.2 Pembuatan Form

Dalam program Kamus Bahasa Arab Indonesia ini dibagi menjadi lima form tampilan yang masing-masing memiliki fungsi sendiri-sendiri. Form-form tersebut adalah :

1. Form Utama, merupakan form dialog untuk menerima data masukan dari user. Form ini juga berfungsi untuk menampilkan form menambah data dalam basis data.
2. Form Penampilan Hasil, form ini digunakan untuk menampilkan hasil dari proses penelusuran pola kata dan pencarian arti kata. Form ini otomatis ditampilkan apabila data yang diinputkan sesuai dengan pola kata yang ada dalam basis aturan.

3. Form Penampilan Aturan-aturan yang digunakan dalam basis pengetahuan, form ini berfungsi sebagai informasi bagi user mengenai aturan-aturan yang dipakai dalam basis pengetahuan.
4. Form Editor Basis data Kosakata, form ini berfungsi sebagai antarmuka bagi user apabila ingin menambahkan, merubah atau mengurangi data arti kata yang ada dalam basis data.
5. Form Editor Basis data Wazan, form ini memiliki fungsi seperti form basis data kosakata yaitu untuk menambah, merubah atau mengurangi data pola kata yang ada dalam basis data.

Sutruktur project dari program tugas Akhir ini adalah sebagai berikut:

```
program Kamus;
uses
  Forms,
  main in 'main.pas' {MainForm},
  DM in 'DM.pas' {Data_Kamus: TDataModule},
  Tampil in 'Tampil.pas' {FTampilRule},
  AddData in 'AddData.pas' {DataForm},
  Newchain in 'Newchain.pas', {mesin inferensi}
  Arti in 'Arti.pas' {FormArti},
  EditWazan in 'EditWazan.pas' {DataWazan};

{$R *.RES}

begin
  Application.Initialize;
  Application.CreateForm(TMainForm, MainForm);
  Application.CreateForm(TData_Kamus, Data_Kamus);
  Application.CreateForm(TFTampilRule, FTampilRule);
  Application.CreateForm(TDataForm, DataForm);
  Application.CreateForm(TFormArti, FormArti);
  Application.CreateForm(TDataWazan, DataWazan);
  Application.Run;
end.
```

MILIK PERPUSTAKAAN
INSTITUT TEKNOLOGI
SEPULUH - NOPEMBER

4.3.3 Pembuatan Basis Data

Pembuatan basis data dari program Kamus Bahasa Arab Indonesia ini dibagi menjadi dua bagian, yaitu basis data untuk menyimpan data kosakata dan yang kedua adalah basis data untuk menyimpan data makna wazan.

Format dari tabel untuk menyimpan kosakata kamus adalah sebagai berikut :

Tabel 4.2 Format tabel Basis data Kosakata

Nama Field	Jenis	Ukuran
Kata	Char	30
Wazan	Char	2
Arti	Char	50

Sedangkan untuk basis data wazan memiliki format tabel sebagai berikut :

Tabel 4.3 Format Tabel Basis data Wazan

Nama Field	Jenis	Ukuran
Wazan	Char	20
Makna	Memo	

BAB V

UJI COBA DAN EVALUASI

BAB V

UJI COBA DAN EVALUASI PERANGKAT LUNAK

Perangkat lunak yang dibangun pada tugas akhir ini dirancang untuk dapat mengenali kata-kata dalam bahasa arab yang bukan merupakan bentuk dasar, baik dengan menggunakan harokat maupun tanpa menggunakan harokat. Jenis kata yang bisa diartikan meliputi semua jenis kata yaitu kata benda, kata kerja dan huruf dengan syarat kata tersebut bisa dicari susunan asalnya, dan arti kata tersebut terdapat dalam basis data. Untuk arti kata yang diketahui polanya akan tetapi kata tersebut tidak terdapat dalam basis data, maka akan ditampilkan susunan huruf asalnya yang merupakan kata kunci untuk pencarian arti kata dalam kamus yang lain.

Sebelumnya diuraikan tentang hasil uji coba perangkat lunak dengan berbagai masukan yang berasal dari turunan kata. Akan dijelaskan mengenai lingkungan kerja tempat perangkat lunak ini dijalankan. Kemudian baru diadakan evaluasi terhadap perangkat lunak.

Lingkungan system yang bisa digunakan untuk menjalankan perangkat lunak ini disarankan mempunyai spesifikasi seperti di bawah ini :

- Prosessor Pentium 100 atau yang lebih tinggi
- Memory 16 MB atau lebih.
- VGA Memory 1 MB atau lebih
- Sistem Operasi Windows 9.X Arabic

5.1 UJI COBA PERANGKAT LUNAK

Berikut ini dijelaskan uji coba perangkat lunak dengan menggunakan beberapa jenis input.

Uji coba tersebut dilakuknan pada komputer dengan spesifikasi sebagai berikut

- Prosessor AMD-K5 133
- Memory 64 MB
- VGA Memory 2 MB
- Sistem Operasi Windows 98 Arabic

Dibawah ini merupakan contoh-contoh dari hasil uji coba perangkat lunak yang dibuat dengan memberikan masukan berupa kata-kata yang merupakan bukan merupakan asal kata. Kata-kata yang ditampilkan pada uji coba kali ini dicarikan kata-kata yang sudah sering kita dengar seperti kata masjid, majelis, 'aalim, mu'miniin dan mu'minat.

Pada uji coba yang pertama diberikan masukan berupa kata

مَسْجِدٌ (masjidun) berasal dari kata سَجَدَ (sajada)

masjidun merupakan bentuk isim makan (keterangan tempat) dari arti kata asal, sehingga kata tersebut memiliki arti tempat sujud.

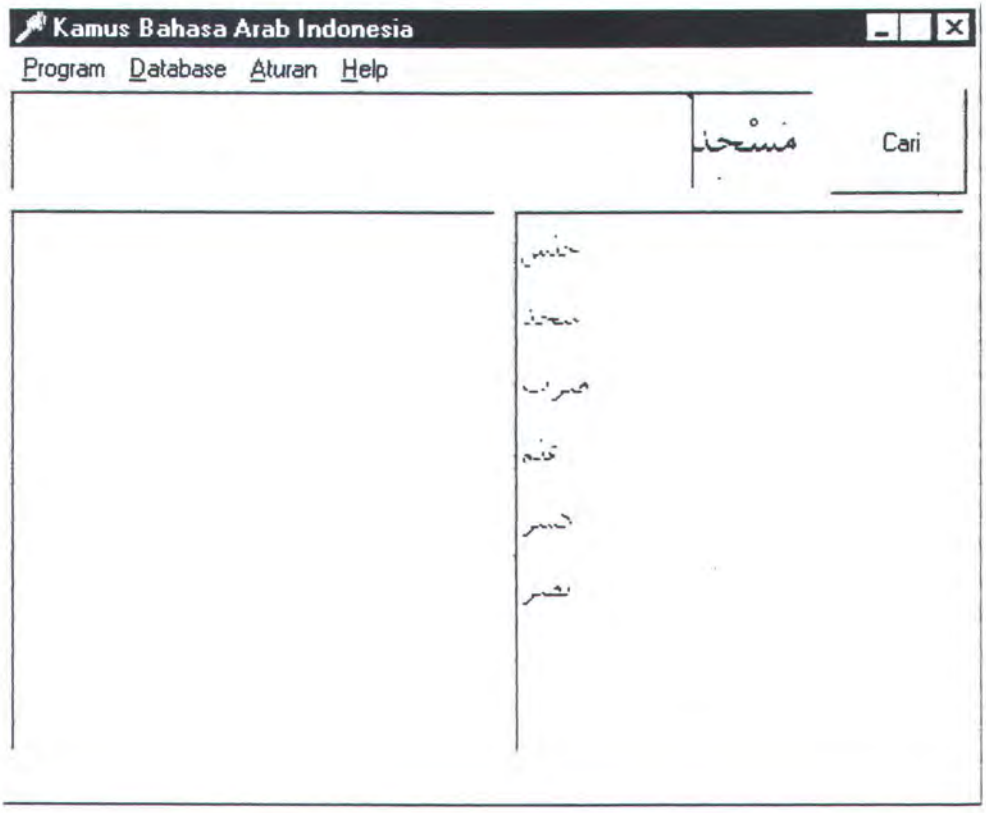
Proses yang dilakukan untuk mencari asal kata masjidun adalah dengan mencocokkan kata tersebut dengan kaidah-kaidah yang ada, dari input kata tersebut akan didapatkan token-token premis awal sebagai berikut :

huruf-1=mim, huruf-2=sin, huruf-3=jim, huruf-4=dal, harokat-1=fathah,

harokat-2 = sukun, harokat-3=kasroh, harokat-4=dhammah tain

pada prosesnya akan sesuai dengan aturan nomor 41 dan 42, kemudian kata tersebut disusun bentuk dasarnya dan didapatkan asal kata sajada yang artinya

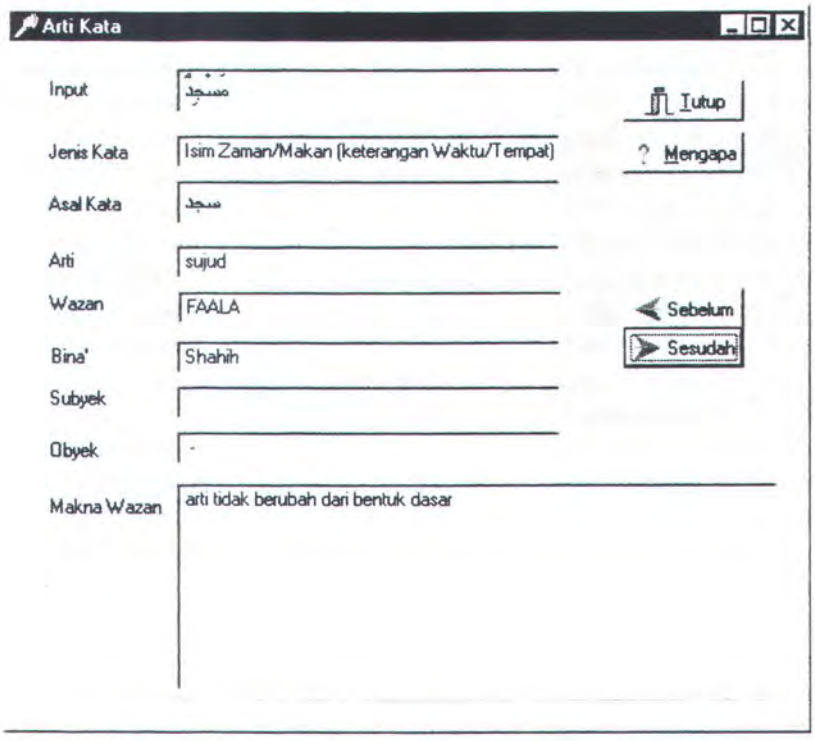
sujud. Tampilan input dari perangkat lunak tersebut bisa dilihat pada **Gambar 5.1.** Hasilnya bisa dilihat pada **Gambar 5.2**



Gambar 5.1 Uji coba dengan masukan kata masjid

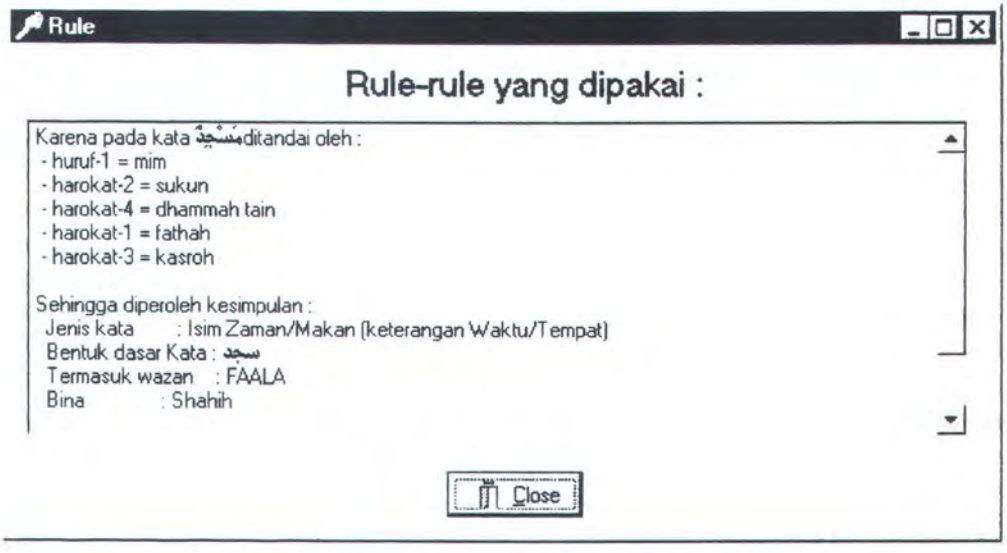
Adapun artinya sesuai dengan informasi diatas adalah tempat / waktu yang digunakan untuk melakukan perbuatan sesuai dengan bentuk dasar. Sedangkan makna wazan tidak berubah karena masih merupakan wazan tsulastsy mujarrad. Makna wazan akan berubah jika termasuk wazan tsulastsy mazied atau termasuk wazan ruba'i mazied.

Tampilan hasilnya adalah sebagai berikut :



Gambar 5.2 Makna dari kata Masjid

Apabila user ingin mengetahui mengapa bisa ketemu asal kata tersebut, maka bisa ditampilkan tanda-tanda yang digunakan untuk mengenali kata tersebut :



Gambar 5.3 Tampilan penjelasan

MILIK PERPUSTAK
INSTITUT TEKNOL
SEPULUH - NOPEMB

Sedangkan uji coba yang kedua menggunakan masukkan berupa kata :

مَجْلِس (majlisun) yang berasal dari kata جَلَسَ (jalasa)

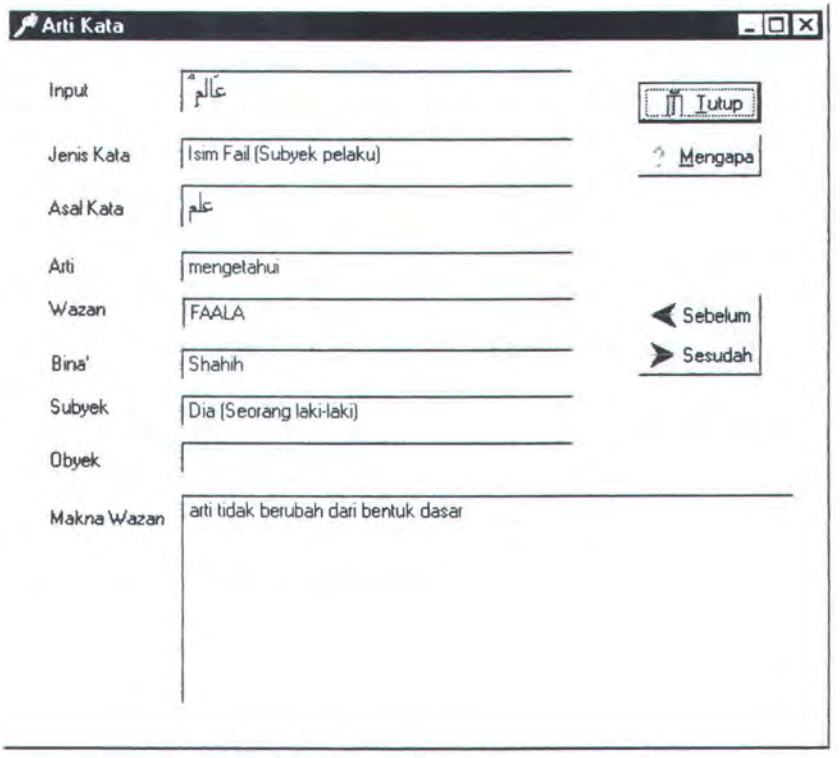
Seperti kata *masjidun* kata *majlis* merupakan bentuk isim makan dari kata *jalasa* yang artinya duduk sehingga maknanya menjadi tempat yang dipakai untuk duduk. Disini kata yang disimpan adalah kata *jalasa*.

Sedangkan uji coba yang ketiga adalah menggunakan masukkan berupa kata :

عالِمٌ ('aalimun) yang berasal dari kata ('alama).

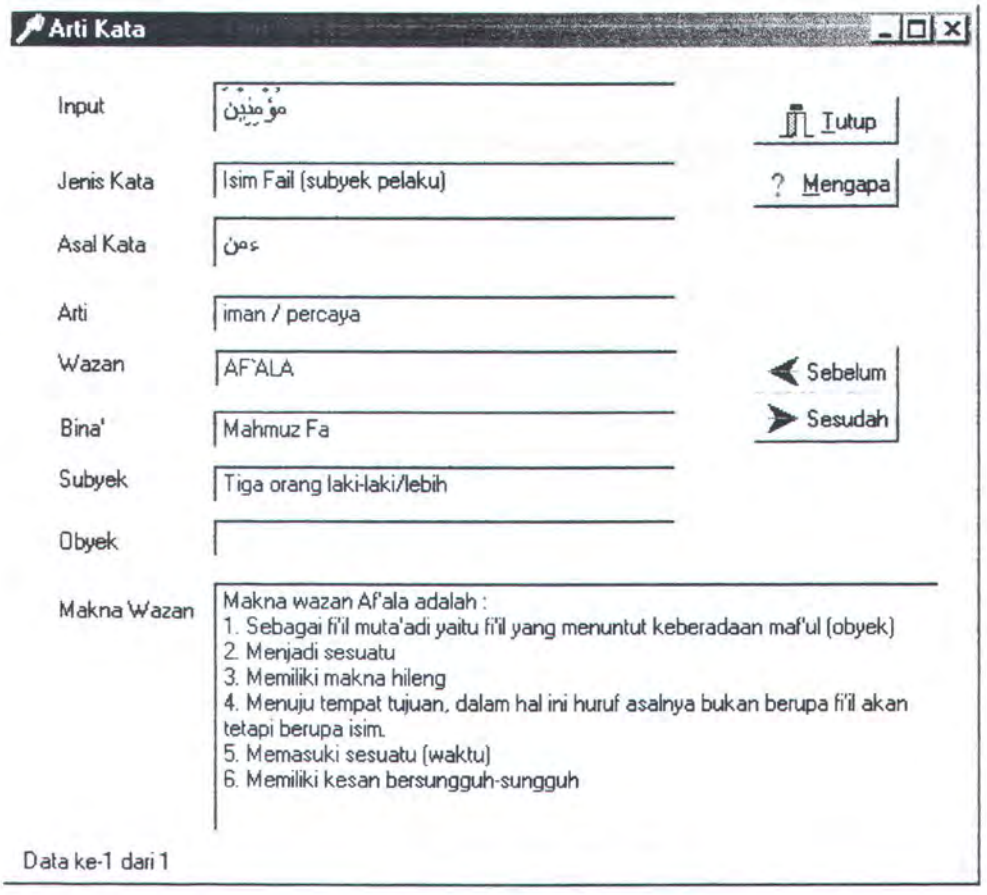
Kata 'aalimun merupakan bentuk isim fail (subyek pelaku) yang berasal dari kata 'alama kemudian diikutkan wazan fa'ilun yang merupakan bentuk isim fail dari wazan fa'ala. Kata 'aalimun ini memiliki arti orang yang berbuat 'alama atau dengan kata lain orang yang mengetahui.

Berikut ini tampilan hasil output dari input 'aalimun



Gambar 5.4 Tampilan Hasil input 'aalimun

Pada uji coba yang keempat menggunakan input kata mu'minin, kata ini apabila kita cari didalam kamus tidak ada karena merupakan bentuk jamak dari kata mu'min, sedangkan kata mu'min sendiri merupakan bentuk isim fa'il dari kata amana yang diikutkan kepada wazan af'ala. Tampilan output dari makna kata mu'minin adalah sebagai berikut :

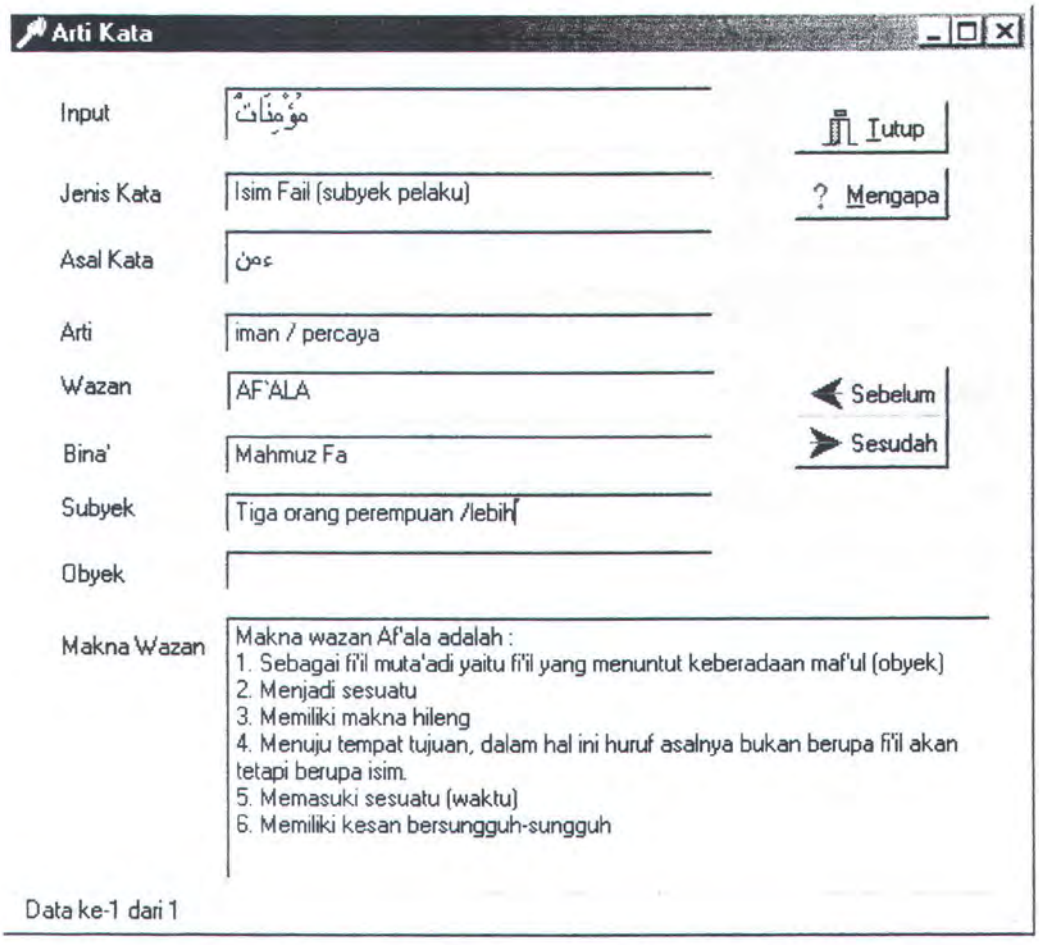


Gambar5.5 Output kata Mu'miniin

Arti yang tepat untuk kata diatas adalah orang-orang laki-laki yang menjadi beriman, atau orang-orang laki-laki yang telah beriman.

Sedangkan uji coba yang terakhir adalah menggunakan kata mu'minat. Kata ini merupakan bentuk jamak jenis perempuan *(jamak muanats salim)* dari kata mu'min. Sehingga arti yang tepat untuk susunan ini adalah orang-orang wanita yang telah beriman.

Tampilan dari input kata mu'minat adalah sebagai berikut :



Gambar 5.6 Tampilan arti kata mu'minat

5.2. EVALUASI PERANGKAT LUNAK

Metode penyimpanan yang digunakan dalam perangkat lunak Kamus Bahasa Arab Indonesia ini untuk kata-kata yang merupakan turunan dari kata kerja yang disimpan hanyalah bentuk dasarnya saja, sedangkan apabila kata tersebut memang merupakan asli berupa kata benda atau huruf maka kata tersebut langsung disimpan kedalam basis data, juga apabila kata tersebut merupakan kata kerja yang tidak bisa di kias maka kata tersebut harus disimpan langsung kedalam basisdata. Kosa-kata yang ada dalam kamus ini hanya

sekedar contoh saja. Untuk melengkapi data kosa-kata bisa dilakukan melalui form editor kosakata.

Progam ini juga dilengkapi dengan penanganan variabel yang tidak diketahui pada mesin inferensi, sehingga meskipun data masukkan tidak menggunakan harokat yang lengkap, alternatif-alternatif pola kata yang sesuai masih tetap bisa ditemukan. Agar pemakai tidak kesulitan dalam membandingkan antar alternatif makna, maka program akan menelusuri harokat-harokat yang sesuai untuk pola tersebut.

Perangkat lunak ini dibuat dengan menggunakan system operasi Windows 98 Arabic, dengan tujuan agar perangkat lunak mudah untuk diaplikasikan dengan aplikasi-aplikasi dibawah lingkungan sistem operasi Windows Arabic yang lain. Dimana file-file text arab biasanya dibuat dengan menggunakan perangkat lunak dalam Windows Arabic. Sehingga bisa dilakukan copy-paste dari dokumen-dokumen text arab yang ada.

Sedangkan hasil tampilan memuat jenis kata, asal kata, wazan, bina' dan pelakunya. Dengan model tampilan seperti diatas diharapkan dapat sekaligus untuk mempelajari makna-makna lain yang terkandung dalam kata tersebut.

Untuk saat ini basis pengetahuan yang dibuat masih untuk wazan-wazan umum dan sering dipakai. Akan tetapi tidak menutup kemungkinan untuk menambahkannya lagi tanpa harus mengkompile ulang program.

Dari uraian diatas dapat dilihat kelebihan perangkat lunak ini bila dibandingkan dengan jenis kamus bahasa Arab yang lain. Kelebihan itu antara lain adalah :

1. Penyimpanan data lebih hemat, karena daftar kata yang disimpan hanya bentuk dasarnya saja.

2. Jenis kata yang diterima juga bisa lebih banyak, karena mencakup turunan turunan kata dari bentuk dasar.
3. Program dibuat dilingkungan Sistem Operasi Windows Arabic, sehingga mudah diaplikasikan dengan aplikasi Arabic yang lain.
4. User tidak harus mengetahui tanda bacanya, dan akan ditampilkan alternatif-alternatif cara membaca kata input.

Sedangkan keterbatasan program ini adalah :

1. Makna kata yang ditampilkan bukan merupakan, makna kata dari bahasa indonesia secara langsung. Sehingga pengguna harus mencari kata-kata dalam bahasa Indonesia yang lebih tepat sesuai dengan makna kata yang dimaksud.
2. Jenis pola kata yang diterima jumlahnya masih terbatas, dan perlu ditambahkan.
3. Jumlah kosa kata yang diinputkan masih terbatas, akan tetapi tidak menutup kemungkinan untuk menambahkannya.

MILIK PERPUSTAKAAN
INSTITUT TEKNOLOGI
SEPULUH - NOPEMBER

BAB VI

PENUTUP

BAB VI

PENUTUP

Demikian buku tugas akhir ini disusun, selama proses pembuatan perangkat lunak dan penyusunan buku tugas akhir ini telah diusahakan semaksimal mungkin agar bisa diambil kegunaan dan manfaatnya. Namun demikian tentunya tidak luput dari kekurangan.

Berikut ini diuraikan beberapa kesimpulan yang dapat diambil dari tugas akhir ini dan saran-saran yang bisa diberikan oleh penulis untuk penelitian yang dapat dilakukan dimasa mendatang.

6.1 KESIMPULAN

Beberapa kesimpulan yang didapatkan dari Tugas Akhir ini adalah sebagai berikut :

1. Pembuatan suatu kamus bahasa Arab yang hemat dalam penyimpanan data dapat dibuat dengan cara hanya menyimpan bentuk dasarnya saja, kemudian ditambahkan suatu proses agar dapat menerima input kata yang bukan merupakan kata dasar.
2. Di dalam kamus bahasa Arab Indonesia ini, sistem berbasis pengetahuan digunakan untuk melakukan pencarian bentuk dasar suatu kata, dengan cara memberikan aturan-aturan tentang wazan (neraca) suatu turunan kata ke dalam basis pengetahuan. Sehingga program bisa mengetahui asal kata dari suatu turunan kata.
3. Dengan adanya perangkat lunak ini pengguna dapat mencari suatu arti kata bahasa Arab tanpa harus mengetahui bentuk dasarnya. Sehingga

akan sangat membantu bagi mereka yang belum paham benar mengenai perubahan-perubahan yang ada pada suatu kata dalam Bahasa Arab.

6.2 SARAN-SARAN

Dengan adanya beberapa kelebihan dan kekurangan yang dimiliki oleh perangkat lunak ini, berikut ini diberikan beberapa saran untuk pengembangan selanjutnya :

1. Arti kata yang ditampilkan bukan merupakan arti secara langsung, melainkan arti menurut bentuk dasar dan pola kata, sehingga perlu ditambahkan proses untuk mengolah arti tersebut sehingga diperoleh arti yang lebih mudah dipahami.
2. Data aturan-aturan dan kosakata perlu ditambah sehingga bisa menjadi kamus yang benar-benar lengkap.
3. Ditambahkannya proses untuk mengecek grammar suatu kalimat, sehingga bisa dikembangkan menjadi program penterjemah dari bahasa Arab ke Bahasa Indonesia.

MILIK PERPUSTAKAAN
INSTITUT TEKNOLOGI
SEPULUH - NOPEMBER

DAFTAR PUSTAKA

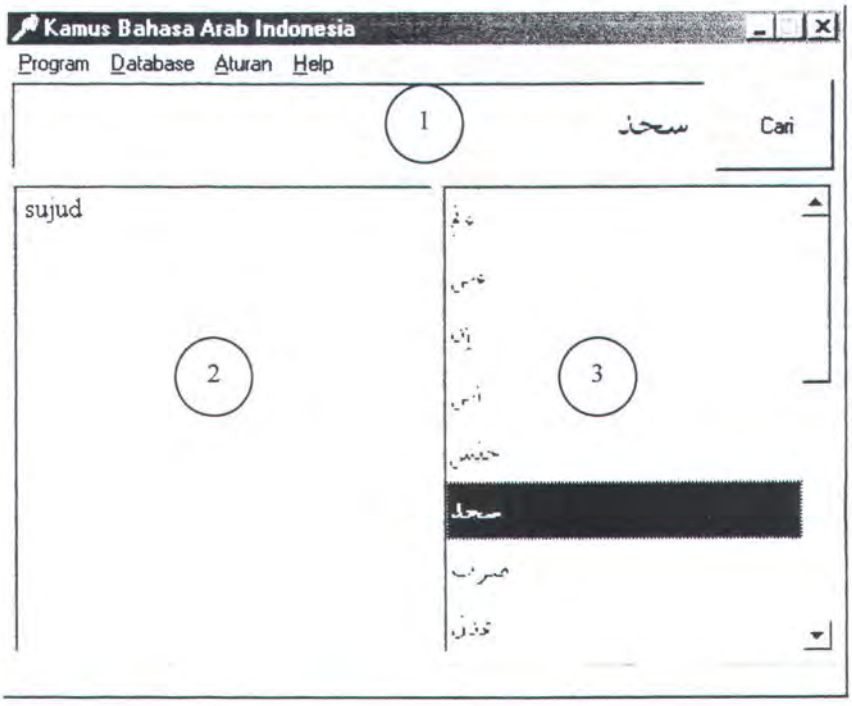
DAFTAR PUSTAKA

1. Anwar, Moch,KH, *Ilmu Nahwu*, Sinar Baru Algesindo, Bandung, 1995.
2. Anwar, Moch, *Ilmu Sharaf*, Sinar Baru Algesindo, Bandung, 1989.
3. Durkin, John, *Expert System Design and Development*, MacMillan Publishing, New York, 1994
4. Gonzales Avelino J and Dankel Douglas D, *The Engineering of Knowledge-Based System*, Prentice Hall International Edition, 1993
5. Lauriere, Jean louis,*Problem Solving and Artificial Inteligence*, Prentice Hall International (UK), Great Britain, 1987
6. Munawwir, Ahmad Warson, *Al-Munawwir Kamus Arab-Indonesia*, Pustaka Progressif, 1997
7. Ma'luf, Lutfi, *Kamus Al-Munjid, Darul Masyrik,* Beirut, Lebanon, 1987
8. Nadwi, Abdullah Abbas, Dr., *Belajar Mudah Bahasa Al-Quran*, Mizan, 1999
9. Nadwi, M. Maftuhin Sholeh, *Ilmu Bahasa Arab Shorof*, Putra Jaya, Surabaya, 1982.
10. Yunus, Mahmud, Prof. H, *Kamus Arab Indonesia*,Yayasan Penyelenggara Penterjemah Pentafsiran Al-Quran Jakarta, 1973

LAMPIRAN

LAMPIRAN
Petunjuk Penggunaan Program

L1. Tampilan Awal

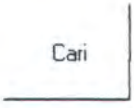


Keterangan :

1 : Kotak isian untuk memasukkan kata yang akan dicari artinya.

2 : kotak yang akan menampilkan arti dari kata yang disimpan dalam basisdata.

3 : Kotak pilihan yang menampilkan kata-kata yang disimpan dalam basisdata

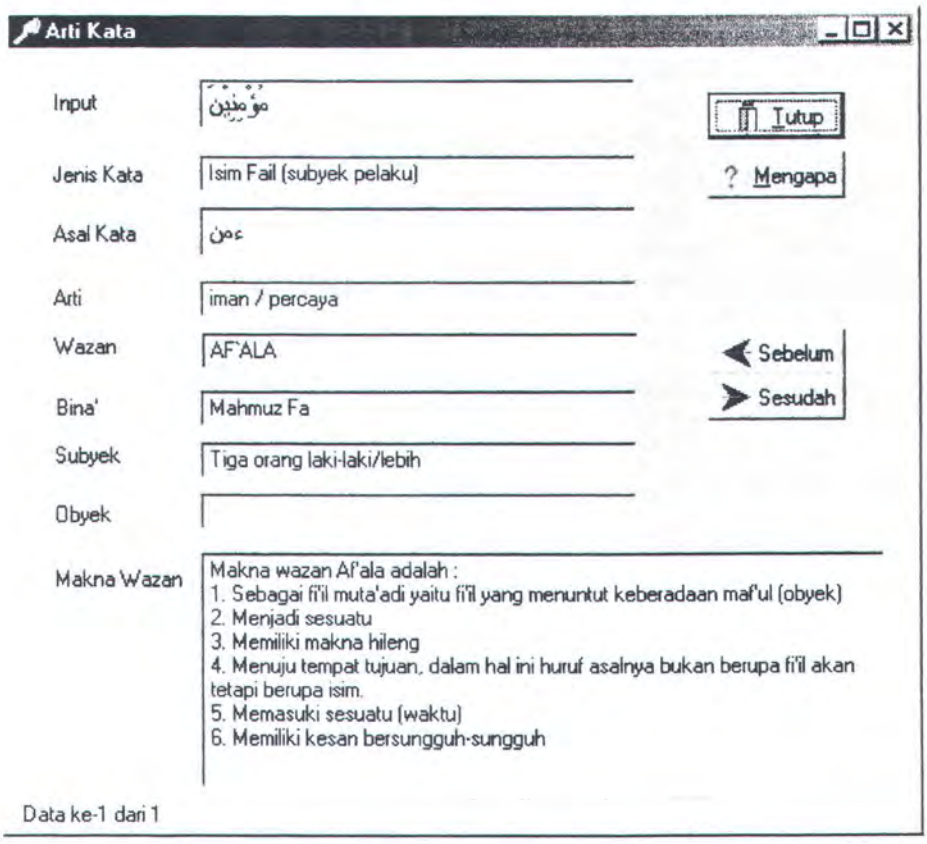


: Tombol untuk memulai pencarian arti kata.

Langkah pencarian :

Masukkan kata yang akan dicari artinya kedalam kotak no. 1, kemudian click tombol Cari. Apabila kata tersebut termasuk kata asli dan terdapat dalam basisdata, otomatis akan ditampilkan artinya pada kolom no. 2, sedangkan apabila merupakan kata turunan atau tidak terdapat dalam basisdata maka akan dicari bentuk dasarnya dan akan ditampilkan dialog tampilan makna kata.

L.2 TAMPILAN MAKNA KATA



Keterangan :

Input : Berisikan kata input yang dicari artinya.

Jenis kata : Menampilkan jenis kata dari kata yang dicari

Asal kata : Menampilkan kata asal dari kata yang dimasukkan

Arti : Menampilkan arti dari kata asal.

Wazan : Menampilkan jenis wazan dari kata yang dimasukkan.

Bina' : Menampilkan bina' kata.

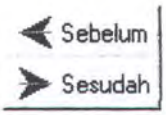
Subyek : Menampilkan jenis pelaku dari kata yang dimasukkan *(fail)*.

Obyek : Menampilkan jenis penderita dari kata yang dimasukkan *(maful)*

Makna Wazan : menjelaskan makna kata menurut wazannya, disini memungkinkan makna yang lebih dari satu.

: Tombol untuk menutup tampilan

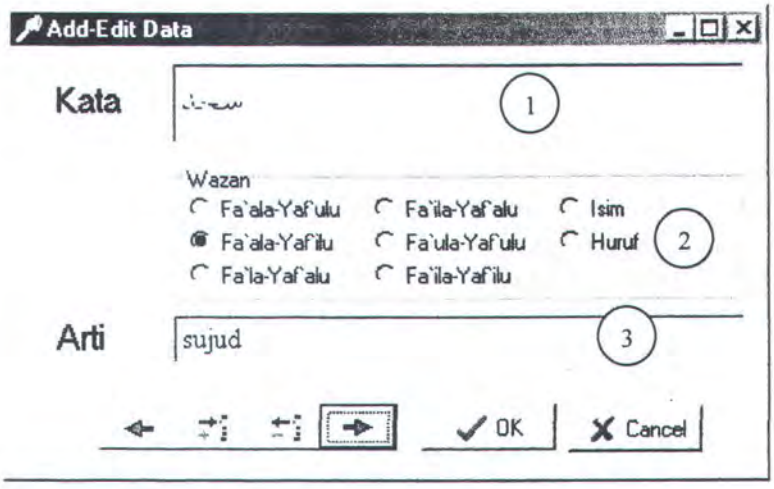
: Tombol untuk menampilkan alasan didapatkannya asal kata.



: Tombol untuk menampilan data sebelum /sesudah jika didapatkan pola lebih dari satu.

: Menampilkan status posisi data hasil pencarian.

L.3 TAMPILAN DIALOG TAMBAH DATA KOSAKATA



Keterangan :

1 : Kotak untuk kata bahasa arab.

2 : Pilihan jenis wazan.

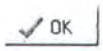
3 : Kotak untuk menampilkan arti kata.

: Tombol untuk menampilkan data sebelumnya

: Tombol untuk menambah data

: Tombol untuk menghapus data.

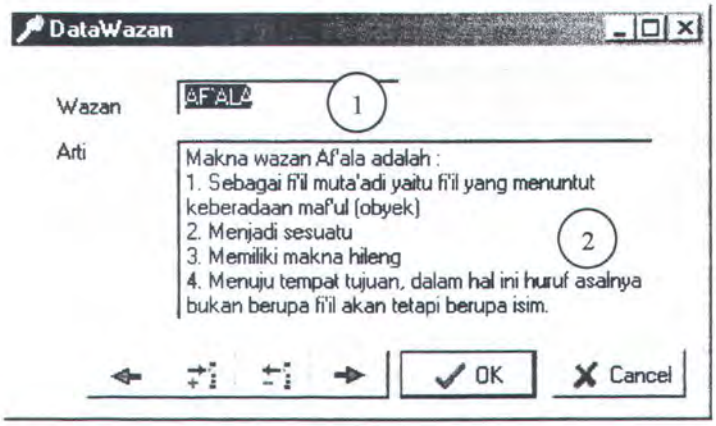
: Tombol untuk menampilkan data selanjutnya.



: Tombol untuk menyelesaikan penambahan data.

: Tombol untuk membatalkan penambahan data.

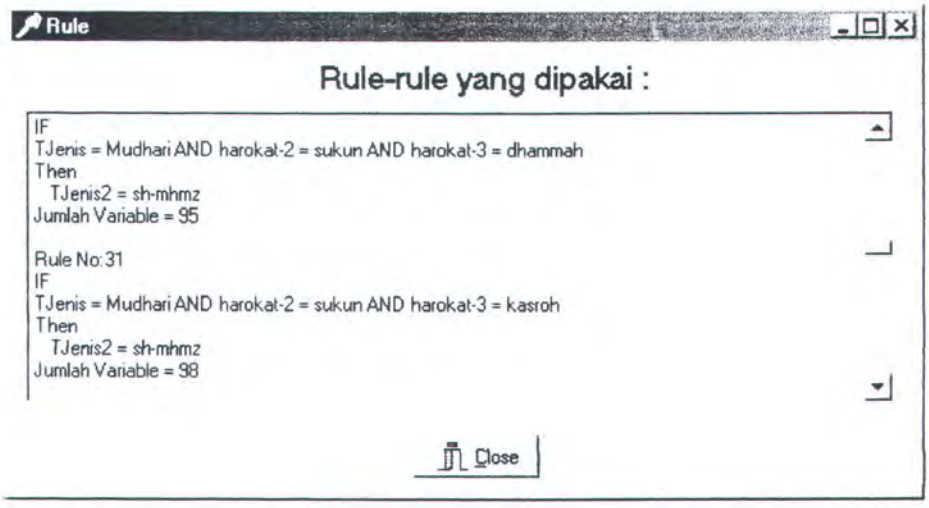
L.4 TAMPILAN DIALOG PENAMBAHAN / PERUBAHAN DATA WAZAN



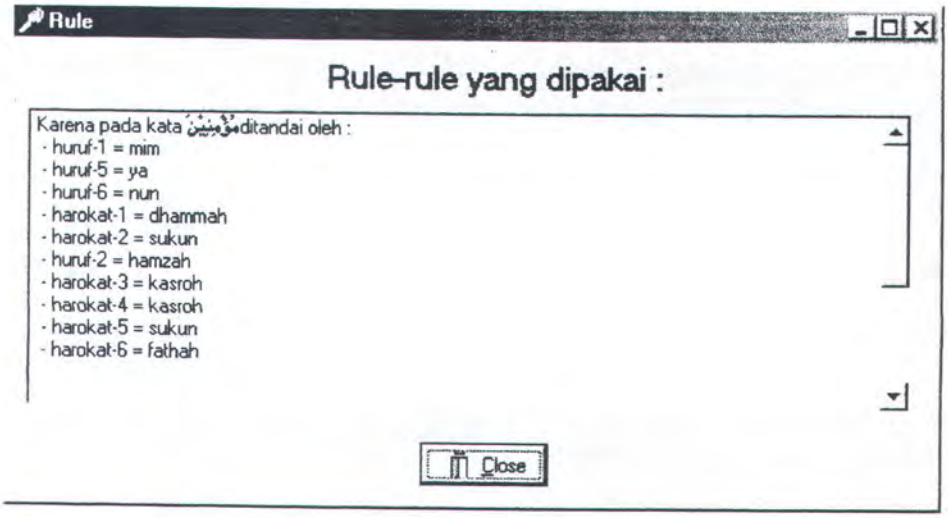
Keterangan :
1 : Berisi nama wazan.
2 : Berisikan arti/makna dari wazan tersebut.

Keterangan tombol sama seperti pada dialog penambahan data kosakata.

L.5 TAMPILAN ATURAN YANG DIGUNAKAN



L.6 TAMPILAN PENJELASAN ALASAN TERPILIHNYA POLA



MILIK PERPUSTAKAAN
INSTITUT TEKNOLOGI
SEPULUH - NOPEMBER