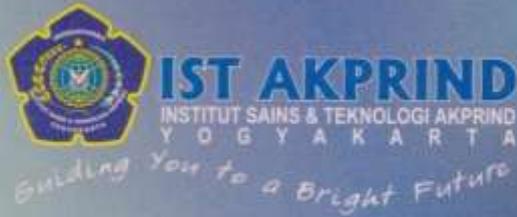


B-7  
ISSN : 1979 - 91 X



# Prosiding

*Seminar  
Nasional*

**Aplikasi Sains &**

**Teknologi**

Kampus IST AKPRIND Yogyakarta  
Sabtu, 13 Desember 2008

### Komite Pelaksana

Bambang Kusmartono, S.T., M.T.  
Ir. Risma Adelina Simanjuntak, M.T.  
Drs. Khairul Muhajir, M.T.  
Ir. Muhammad Suyanto, M.T.  
Hadi Prasetyo Suseno, S.T., M.Si.  
Ir. Dwi Indah Purnamawati, M.Si.  
Syafriyudin, S.T., M.T.  
Emy Setyaningsih, S.Si., M.Kom.  
Edhy Sutanta, S.T., M.Kom.  
Ir. Joko Susetyo, M.T.  
Endang Widuri Asih, S.T., M.T.  
Muhammad Andang Novianta, S.T., M.T.  
Imam Sodikin, S.T., M.T.  
Catur Iswahyudi, S.Kom., S.E.  
Ir. Saiful Huda, M.T.  
Ir. Gatot Santoso, M.T.  
Subandi, S.T., M.T.  
Suwanto Raharjo, S.Si., M.Kom.  
Ir. Muhammad Yusuf, M.T.  
Slamet Hani, S.T., M.T.  
Ir. Harry Wibowo, M.T.  
Drs. Yudi Setyawan, M.S., M.Sc.  
Ellyawan Setya Arbintarso, S.T., M.Sc.  
Dra. Harmastuti, M.Kom.  
Ir. Wiwik Handajadi  
Jarot Wijayanto, S.T.  
Ani Purwanti, S.T.  
RR. Yuliana Rachmawati Kusumaningsih, S.T., M.T.  
Sri Hastutiningrum, S.T., M.Si.  
Siti Imsyawati Maulidya, S.T., M.Kom.  
Sigit Priyambodo, S.T., M.T.  
I Gde Badrawada, S.T.  
Ir. Adi Purwanto, M.T.  
Purnomo Sidik  
Suryanto, B.Sc.  
Mardjuki  
Paryono  
Muhammad Feri, S.Kom.  
Salim, S.Kom.  
Arham Aminudin, S.Kom.  
Rochmad Haryanto, S.Kom.  
Djaeri

### Sekretariat

Fakultas Teknologi Industri, IST AKPRIND Yogyakarta  
Jl. Kalisahak No. 28 Komplek Balapan, Yogyakarta, 55222  
Telp. 0274-563029 Fax. 0274-563827  
website: [www.snast.akprind.ac.id](http://www.snast.akprind.ac.id);  
email: [snast@akprind.ac.id](mailto:snast@akprind.ac.id)

### Penerbit

IST AKPRIND Yogyakarta

## DAFTAR ISI BIDANG INFORMATIKA / KOMPUTER

1.	Aplikasi Natural Language Processing Dengan Metode Rule Based Pada Terjemahan Indonesia - Minang Berbasis Web <i>Dewi Soyusiawaty</i>	1-14
2.	News Agent Untuk Pencarian Dan Peringkasan Berita <i>Gunawan</i>	15-22
3.	Rancangan Sistem Pameran Online Menggunakan Metode UCD (User Centered Design) <i>Armadyah Amborowati</i>	23-30
4.	Aplikasi Algoritma <i>Dijkstra</i> Untuk Pencarian Jalur Terpendek <i>Uning Lestari</i>	31-38
5.	Perancangan Dan Pembuatan Data Warehouse Pada Perpustakaan Stmik Amikom Yogyakarta <i>Armadyah Amborowati</i>	39-52
6.	Implementasi Document Object Model Untuk Perancangan Laporan Database MySQL Berbasis Web <i>Antonius Rachmat Chrismanto</i>	53-63
7.	Implementasi JXTA Sebagai Platform Jaringan Peer-To-Peer Pada Mobile Device Untuk Aplikasi File Sharing <i>Doddy Poerwo Haryadi, Agus Virgono dan Gunawan Adi</i>	64-70
8.	Implementasi Knowledge Management Untuk Meningkatkan Kinerja Perguruan Tinggi <i>Agus Mulyanto</i>	71-78
9.	Desain Algoritma Pemecahan Operasi Dasar Aritmatika Dan Pemodelan Sistem Akuisisi Data Terdistribusi Berbasis Web <i>Diyurman Gea, Johan Setiawan dan Indrajani</i>	79-84
10.	Aplikasi Buku Elektronik (E-Book) Berbasis Web Pendukung Konversi Format Dokumen *.Doc Menjadi *.Pdf <i>Eko Handoyo, Rizal Isnanto, dan Retna Arliana</i>	85-92
11.	Implementasi Metode Function Point Untuk Mengukur Volume Software <i>M. Dedi Iskandar dan Kusri</i>	93-108
12.	E-Government : Sistem Dalam Efisiensi Dan Efektifitas Perpajakan <i>Hidayatulah Himawan</i>	109-117
13.	Analisis Faktor-Faktor Yang Mempengaruhi Pengadopsian Electronic Commerce Dan Pengaruhnya Terhadap Kinerja Perusahaan (Studi Pada Perusahaan Kecil Dan Menengah Di Indonesia) <i>Vidi Arini Yulimar dan Augustinus Setiawan</i>	118-128
14.	Teknik Animasi Inbetweening Dengan Transformasi 2D Pada Grafis Primitif <i>Pulut Suryati</i>	129-137
15.	Algoritma Companion Dan Cocitation Untuk Pencarian Halaman Web Yang Terkait <i>Gunawan</i>	138-149
16.	Penataan Ulang Software Requirement Specification (SRS) Sistem Informasi Akademik Dengan Pendekatan Reverse Engineering <i>Eko K. Budiardjo dan Yanti Andriyani</i>	150-156
17.	Aplikasi Pengolahan Citra Digital Pada Telepon Seluler <i>Herry Sofyan</i>	157-165

18.	Pemakaian Program dinamik dalam pengelolaan & pengoperasian sumber daya air <i>Yeni Nuraeni</i>	166-174
19.	Analisis Keamanan Sistem Informasi Akademik Berbasis Web Di Fakultas Teknik Universitas Diponegoro <i>Kodrat Iman Satoto, R. Rizal Isnanto, dan Ahmad Masykur</i>	175-186
20.	Rekam Jejak Kebutuhan Tenaga Kerja Teknologi Informasi (Ti) Di Sumatera Selatan <i>Wijang Widhiarso</i>	187-196
21.	Pemanfaatan Metode Visualisasi Secara Tiga Dimensi Untuk Aplikasi Berbasis Web <i>Mursid W. Hananto</i>	197-202
22.	Peran Statistik Dalam Pengambilan Keputusan Pada Penentuan Sekolah Secara On-Line <i>Dina Andayati</i>	203-209
23.	Implementasi Temu Kembali Informasi Model Ruang Vektor Untuk Situs Web <i>Nurhadi dan Amir Hamzah</i>	210-219
24.	Perbandingan Kinerja Model-Model Temu Kembali Pada Sistem Temu Kembali Informasi Dokumen Berbahasa Indonesia <i>Amir Hamzah, F. Soesianto, Adhi Susanto, dan Jazi Eko Istiyanto</i>	220-233
25.	Membangun Aplikasi Web Statistik Untuk Menganalisa Trafik Web Menggunakan PHP Dan MySQL <i>Catur Iswahyudi dan Erma Susanti</i>	234-242
26.	Analisis Keamanan Sistem Aplikasi (Study Kasus Pada Aplikasi E-Learning Di IST Akprind Yogyakarta) <i>Edhy Sutanta</i>	243-253
27.	Aplikasi SMS Gateway Sebagai Sarana Informasi Indeks Prestasi Kumulatif Menggunakan Kannel 1.4.1 Pada Ubuntu 7.04 <i>Erfanti Fatkhiyah dan Adam Sekti Aji</i>	254-265
28.	Analisis Kriptografi Menggunakan Algoritma <i>Vigenere Cipher</i> Dengan Mode Operasi <i>Cipher Block Chaining (CBC)</i> <i>Erna Kumalasari Nurnawati</i>	266-272
29.	Membangun Sistem Keamanan Website <i>Iswari Susanti, Syaiful Rijal, dan Muhammad Sholeh</i>	273-278
30.	Penerapan Sistem Informasi Berbasis Short Message System <i>Muhammad Sholeh dan Agus Herryani</i>	277-282
31.	Implementasi Mikrotik Sebagai Manajemen <i>Bandwidth</i> <i>Rr. Retna Trimantaraningsih dan Istiqomatul Muarifah</i>	283-295
32.	Aplikasi Game Berbasis Java 2 Micro Edition (J2ME) <i>Trio Ardiyanto, Darwin Suhendra, dan Muhammad Sholeh</i>	296-301
33.	Sistem Informasi Akademik Berbasis SMS Gateway <i>Yudhi Irawan, Didik Amir Wahyudi, dan Bayu Ario Sumbodo</i>	302-316
34.	Desain Aplikasi Collecting Data Borang Akreditasi Program Studi (Studi Kasus Prodi Teknik Informatika IST Akprind) <i>Yuliana Rachmawati Kusumaningsih</i>	317-322
35.	Data Center Lintas Informasi Dan Integrasi Antar Satker <i>Arief Susanto, ST, Mkom</i>	323-327
36.	Metode Clustering Data Biner <i>Naniek Widayastuti</i>	328-340

# APLIKASI NATURAL LANGUAGE PROCESSING DENGAN METODE RULE BASED PADA TERJEMAHAN INDONESIA - MINANG BERBASIS WEB

Dewi Soyusiawaty  
Universitas Ahmad Dahlan  
Email: my\_soyus@yahoo.com

## INTISARI

Bahasa Minang sebagai salah satu bahasa daerah di Indonesia mulai terancam keberadaannya. Hal ini disebabkan oleh pengaruh bahasa Indonesia yang sudah menyebar jauh ke pelosok daerah di Sumatra Barat, yang dibawa oleh media audio visual seperti televisi dan radio. Berbagai usaha telah banyak dilakukan, salah satunya adalah pembuatan kamus untuk bahasa Minangkabau, namun kamus masih memiliki banyak keterbatasan, yaitu jika ingin mengetahui arti sebuah kata harus membaca satu persatu kata yang terdaftar, kata-kata yang terdapat dalam kamus juga dalam jumlah yang terbatas, padahal dalam bahasa Minang ternyata banyak memiliki kesamaan dengan bahasa Indonesia yang dapat dipolakan dengan aturan tertentu, sehingga kata-kata yang dapat dipolakan tidak harus didaftarkan dalam basis data. Tujuan dari penelitian ini adalah membangun aplikasi terjemahan kamus bahasa Indonesia – Minang yang dapat menterjemahkan kata, frase dan kalimat menggunakan metode rule based berbasis web.

Penelitian dimulai dengan mengumpulkan data berupa kosa kata bahasa Indonesia dan bahasa Minang, kemudian secara berturut-turut membuat algoritma rule based, membuat rancangan tampilan, membuat kamus, mengimplementasikan algoritma, dan langkah terakhir adalah melakukan pengujian dari aplikasi yang telah dibuat yaitu dengan Black Box Test dan Alpha Test.

Penelitian yang dibuat menghasilkan sebuah aplikasi yang dapat menerjemahkan kata, frase dan kalimat dari bahasa Indonesia-bahasa Minang. Hasil pengujian menunjukkan bahwa perangkat lunak terjemahan ini layak untuk diimplementasikan

Kata kunci : kamus, terjemahan, rule based, bahasa Indonesia, bahasa Minang

## PENDAHULUAN

Seperti halnya bahasa-bahasa daerah lainnya di Indonesia, Bahasa Minangkabau terus mengalami perubahan. Pengaruh bahasa Indonesia sudah menyebar jauh ke pelosok daerah di Sumatra Barat, dibawa oleh media audio visual seperti televisi dan radio, 'membengkokkan' lidah generasi muda Minangkabau, yang mulai terbata-bata berkomunikasi dalam bahasa ibunya sendiri [7]. Orang Minangkabau umumnya berpendapat banyak persamaan antara Bahasa Minangkabau dengan Bahasa Melayu/Indonesia. Dari perbedaan lafal yang terdapat pada bahasa Indonesia dengan bahasa Minang ditemukan beberapa *rule* yang dapat dibentuk pada bahasa Minang dengan perubahan yang terjadi pada suku kata dalam bahasa Indonesia yaitu perubahan pada suku awal, suku akhir dan gabungannya. Contoh-contoh perbedaan lafal Bahasa Melayu/Indonesia dan Bahasa Minangkabau adalah sebagai berikut : *ut menjadi uik*, contoh : rumput menjadi *rumpuik*, *at menjadi aik*, contoh : adat menjadi *adaik*, *al/ar menjadi a*, contoh : jual menjadi *jua*, kabar menjadi *kaba*, *e menjadi a*, contoh : beban menjadi *baban* dan seterusnya .

Kamus merupakan salah satu fasilitas yang dapat membantu masyarakat untuk mengetahui dan memahami bahasa dari suatu daerah. Kamus sangat penting karena selain dapat menjadi media penerjemah, kamus juga dapat menjadi salah satu sumbangan dalam upaya memeperkaya kebudayaan dari suatu daerah. Walaupun demikian, masih terdapat beberapa kendala dalam mempelajari kamus tersebut antara lain yaitu jika pengguna ingin mencari arti dari sebuah kata, pengguna harus membaca kata per kata sampai kata yang dicari ditemukan. Jumlah kata yang terdapat dalam kamus juga sangat terbatas. Pada media publikasi *website* ada yang sudah menyediakan fasilitas penerjemah atau kamus bahasa Indonesia – Minang, contohnya pada <http://cimbuak.net>. Pada umumnya, fasilitas penerjemah kata tersebut hanya dibuat dalam bentuk *list* atau daftar kata dalam hal ini kata bahasa Indonesia dengan artinya dalam bahasa Minang, kata yang disediakanpun terbatas, selain itu kamus yang disediakan belum menyediakan fasilitas untuk meng-*input*-kan kata yang dicari dan tidak dapat mengartikan kalimat.

Beberapa kekurangan di atas dapat diselesaikan dengan adanya suatu aplikasi penterjemah bahasa Indonesia ke bahasa Minang dengan fasilitas *input* dan *output* sehingga pengguna dapat

dengan mudah mendapatkan kata atau kalimat yang diinginkan. Kata-kata dalam bahasa Indonesia yang memiliki kemiripan dalam bahasa Minangnya dikelompokkan dalam suatu aturan tertentu atau dinyatakan dalam *rule based*. Aplikasi ini diimplementasikan berbasis web dengan tujuan agar dapat dimanfaatkan oleh banyak orang. Rumusan masalah dari penelitian ini adalah : "Bagaimana membangun aplikasi terjemahan bahasa Indonesia ke bahasa Minang dengan pendekatan metode *Rule Based* sehingga dapat menterjemahkan kata atau kalimat dari *user*". Tujuan penelitian ini dilakukan adalah untuk merancang dan mengimplementasikan ke dalam suatu bahasa pemrograman berbasis web sehingga dibangun suatu aplikasi yang dapat digunakan untuk menampilkan terjemahan bahasa Indonesia ke bahasa Minang berdasarkan huruf atau penggolongan kata, menerjemahkan kata dan kalimat dari bahasa Indonesia ke bahasa Minang.

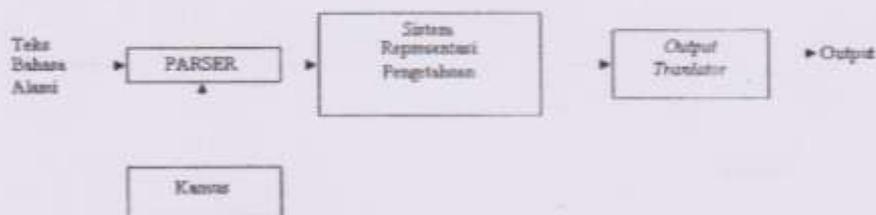
Tahapan yang dilakukan dalam penelitian ini diawali dengan pengumpulan data. Metode Observasi berdasarkan pengamatan dilakukan secara langsung pada objek penelitian yaitu Asrama Minang, Bundo Kanduang, untuk memperoleh data yang diperlukan dan gambaran nyata mengenai hal-hal yang dibutuhkan. Metode Kepustakaan dengan cara mengumpulkan, mempelajari dan memahami buku-buku referensi seperti kamus Indonesia Minang dan buku-buku tentang tata bahasa Indonesia. Tahapan analisis untuk mengetahui kebutuhan user terhadap sistem terjemahan yang akan dibangun. Selanjutnya dilakukan perancangan sistem, seperti : Perancangan Bagan Alir Sistem, dibuat dalam bentuk *Physical Model*. *Physical Model* adalah pemodelan sistem menggunakan bagan alir sistem yang menunjukkan bagaimana sistem dirancang ini nantinya dapat berjalan dan diterapkan, Perancangan Basis Data, untuk mengidentifikasi isi atau struktur dari tiap *file* yang ada, Perancangan Algoritma, digunakan untuk untuk menerjemahkan kata-kata yang tidak terdaftar di dalam database. Algoritma yang dirancang dalam sistem ini yaitu :

- a. Algoritma untuk mengambil karakter dan mengubah karakter tersebut menjadi bentuk konsonan dan vokal.
- b. Algoritma untuk mengecek suku kata depan dan suku kata belakang.
- c. Algoritma untuk mengambil suku kata tengah dari kata
- d. Algoritma untuk menggabungkan suku depan, suku tengah dan suku belakang.

Tahapan perancangan selanjutnya adalah perancangan antarmuka, berhubungan dengan tampilan program yang akan dibuat. Perancangan antarmuka ini meliputi perancangan *input* dan *output* yang diinginkan dalam program nantinya. Tahapan setelah perancangan adalah mulai mengimplementasikan sistem dengan pengkodean. Pada tahap ini rancangan yang telah dibuat diubah ke dalam sebuah program aplikasi. Rancangan yang diimplementasikan pertama kali adalah implementasi algoritma, implementasi *form*, dilanjutkan dengan pembuatan kamus. Terakhir dilakukan pengujian sistem.

Bahasa alami adalah suatu bentuk representasi dari suatu pesan yang ingin dikomunikasikan antar manusia. Bentuk utama representasinya adalah berupa suara/ucapan (*spoken language*), tetapi sering pula dinyatakan dalam bentuk tulisan[6]. Bahasa dapat dibedakan menjadi (1) Bahasa Alami, dan (2) Bahasa Buatan. Bahasa alami adalah bahasa yang biasa digunakan untuk berkomunikasi antar manusia, misalnya bahasa Indonesia, Sunda, Jawa, Inggris, Jepang, dan sebagainya. Bahasa buatan adalah bahasa yang dibuat secara khusus untuk memenuhi kebutuhan tertentu, misalnya bahasa pemodelan atau bahasa pemrograman komputer[6]. Chomsky adalah orang yang pertama kali merepresentasikan bahasa sebagai rangkaian simbol. Chomsky berhasil memperlihatkan bahwa bahasa apapun dapat direpresentasikan dengan suatu cara yang universal. Pemikiran Chomsky yang merepresentasikan bahasa sebagai kumpulan simbol-simbol dan aturan yang mengatur susunan simbol-simbol tersebut telah membuka peluang untuk melakukan pemrosesan bahasa secara simbolik dengan teknologi komputer, sehingga melahirkan bidang ilmu *Natural Language Processing (NLP)*[6].

Pengolahan bahasa alami terdiri dari dua bagian utama, yaitu : *parser*, sistem representasi pengetahuan dan pengolahan *output*.



Gambar 1. Komponen Utama Bahasa Alami

a. Parser

Sistem yang mengambil kalimat input bahasa alami dan menguraikannya ke dalam beberapa bagian gramatikal (kata benda, kata kerja, kata sifat, dan lain-lain).

b. Sistem Representasi Pengetahuan

Suatu sistem yang menganalisis output parser untuk menentukan maknanya.

c. Output Translator

Suatu terjemahan yang merepresentasikan sistem pengetahuan dan melakukan langkah-langkah yang bisa berupa jawaban atas bahasa alami atau *output* khusus yang sesuai dengan program komputer lainnya.

Bahasa merupakan fenomena yang rumit yang melibatkan proses pengenalan bunyi, sintaks kalimat serta *inferensi* semantik tingkat analisis yang berbeda untuk bahasa alamiah. Adapun tingkat analisis bahasa alamiah tersebut adalah :

a. Analisis Sintaksis

Analisis *sintaksis* adalah studi yang menyimak tentang aturan penggabungan kata menjadi yang diperkenankan dan pemakai aturan-aturan tersebut untuk membuat suatu kalimat.

b. Analisis Simantik

Analisis *simantik* adalah ilmu yang menyimak arti kata dan unsur masing-masing kata tersebut dalam suatu kalimat.

c. Analisis Pragmatik

Analisis *pragmatik* adalah studi yang menyimak tentang cara penggunaan bahasa dan pengaruhnya pada pendengar. Analisis pragmatik ini dapat digunakan untuk pemahaman suatu kalimat dengan aplikasi cerita-cerita panjang.

Penerjemahan bahasa diklasifikasikan menjadi dua jenis penerjemahan bahasa yang telah dikenal beberapa saat ini, yaitu penerjemahan bahasa alami ke dalam bahasa alami dan penerjemahan dari bahasa mesin ke dalam bahasa alami. Penjelasan lebih lanjut tentang kedua penerjemahan itu adalah :

1. Penerjemahan bahasa alami ke bahasa alami

Penerjemahan bahasa alami ke bahasa alami adalah dengan digunakannya sistem *parsing* pengenalan kata dilanjutkan dengan analisis *sintaks* pada kalimat, analisis semantik pada kalimat serta keakuratan terjemahan.

2. Penerjemahan bahasa mesin ke bahasa mesin.

Penerjemahan bahasa mesin ke bahasa mesin adalah dengan mengalihkan suatu bahasa pemrograman menjadi satu bahasa pemrograman lain.

Bahasa Minangkabau atau Baso Minang adalah salah satu anak cabang bahasa Austronesia yang dituturkan khususnya di wilayah Sumatra Barat. Orang Minangkabau umumnya berpendapat banyak persamaan antara Bahasa Minangkabau dengan Bahasa Melayu/Indonesia. Tabel di bawah berikut memberikan beberapa contoh kata dalam bahasa Minang berdasarkan penggolongan katanya, yaitu :

Tabel 1. Beberapa contoh kata Bahasa Minang ke Bahasa Indonesia

Kata (Indonesia)	Kata (Minang)	Golongan Kata
saya (Aku)	ambo, awak, denai	Kata Ganti
Kamu	Sanak, sinan, waang	Kata Ganti
Abang/mas/kakak laki-laki	Uda, tuan, tan, uwan, ajo	Kata Ganti
Mbak/kakak perempuan	Uni	Kata Ganti
Satu	Ciek	Kata Bilangan
Dua	Duo	Kata Bilangan
Tiga	Tigo	Kata Bilangan
Lima	Limo	Kata Bilangan
Tiga Puluh Lima	Tigo Puluah Limo	Kata Bilangan
Pergi	Pa'i	Kata Kerja
Berjualan/berdagang	Manggaleh/bajojo/manjua	Kata Kerja
Bercanda	Bagarah	Kata Kerja
Tanya	Tanyo	Kata Kerja
Tidur	Lalok	Kata Kerja
Berantem	Batangka/bagaduah	Kata Kerja
Cantik	Rancak, elok	Kata Sifat
Sembuh	cegak	Kata Sifat
Jujur	jujua	Kata Sifat
Jauh	Jauah	Kata Sifat
Gelas	Galeh	Kata Benda
Mobil	Oto	Kata Benda
Makanan	Panganan	Kata Benda
Piring	Piriang	Kata Benda
Mengapa/Kenapa	Manga	Kata Tanya
Dimana	Dima/Dimano	Kata Tanya

Dari beberapa kata yang dicantumkan pada tabel di atas ditemukan beberapa *rule* yang dapat dibentuk pada bahasa Minang dengan perubahan yang terjadi pada suku kata dalam bahasa Indonesia yaitu perubahan pada suku awal, suku akhir dan gabungannya, yaitu :

Tabel 2. *Rule* perubahan pada suku kata akhir

Perbedaan lafal	Kata Indonesia	Kata Minang
a – o	ada	Ado
al , ar – a	Sambal bakar	Samba baka
as – eh	balas	Baleh
at – ek	padat	padek
ik – iak	balik	padek
il, ir, is – i	kail cair baris	kai cai bari
ing – iang	anjing	anjiang
it – ik	impit	Impik
ul, ur – u	bisul kasur	bisu kasu
us – ui	ratus	ratui
uk – uak	datuk	datuak
ut – uik	rumpuk	rumpuik

Tabel 3. *Rule* perubahan pada suku kata awal

Dari	Menjadi	Kata (Indonesia)	Kata (Minang)
e	a	Beban	Baban

Tabel 4. *Rule* pada suku kata awal dan akhir :

Kata (Indonesia)	Kata (Minang)
Sebut	Sabuik
Gelas	Gale
Keras	Kare
Pedas	Pade'

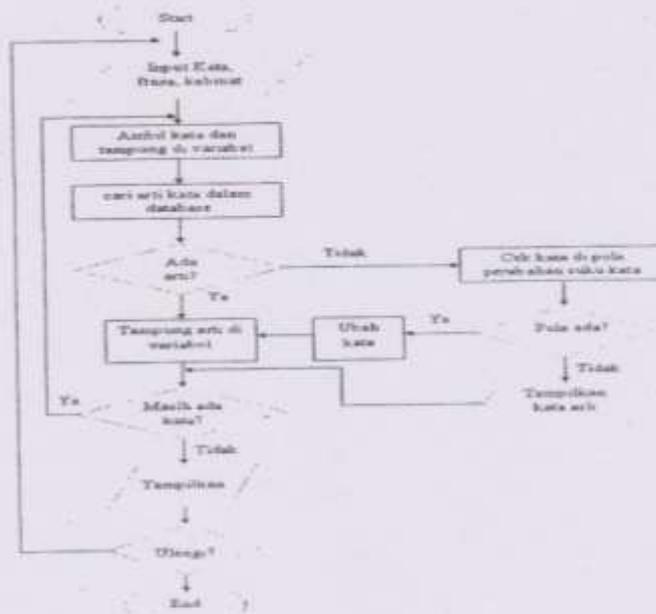
Dari beberapa *rule* perubahan suku kata pada bahasa Indonesia ke bahasa Minang di atas, maka dapat dinyatakan suatu fungsi dalam pemrograman yang dapat mengubah secara otomatis inputan dalam bahasa Indonesia yang mengandung suku kata yang dinyatakan dalam tabel di atas dan diubah menjadi suku kata yang seharusnya dalam bahasa Minang.

## HASIL DAN PEMBAHASAN

Kebutuhan sistem yang dapat disimpulkan adalah :

1. Program mampu menerjemah kata atau kalimat dari bahasa Indonesia ke bahasa Minang.
2. Program menyediakan fasilitas untuk menampilkan daftar kamus kata berdasarkan jenis katanya.
3. Program menyediakan fasilitas untuk membuka file yang akan diterjemah.
4. Program menyediakan fasilitas untuk menyimpan hasil terjemahan.
5. Program menyediakan form admin yang dapat digunakan untuk meng-*update* data kamus.

Tahapan penerjemahan dinyatakan dengan bagan alir terjemahan. Program akan membaca masukan yang berupa kata, frasa atau kalimat, kemudian program mengambil per kata dan menampung kata dalam variabel kata, selanjutnya program mencari arti kata di dalam database, apabila kata terdapat dalam database maka program akan mengambil arti kata dan menampung arti di variabel tempKalimat, jika tidak terdapat dalam database maka kata akan dicek susunan katanya kemudian diproses dengan algoritma *rule based* untuk perubahan suku kata. Kata yang polanya dapat diubah akan diubah sesuai dengan aturan perubahan pola dan ditampung di variabel tempKalimat. Jika kata tidak memenuhi aturan perubahan pola kata, maka program akan menampilkan kata aslinya(kata yang di-*input*-kan). Program akan mengecek apakah masih ada kata dalam *input text*. Jika ada program akan mengulangi membaca masukan, jika tidak maka program akan menampilkan hasil terjemahan.



Gambar 2. Bagan Alir Terjemahan

Berikut ini adalah struktur file basisdata yang digunakan, disusun dalam bentuk kamus data untuk tabel yang diperlukan oleh sistem.

Tabel 5. Tabel KamusKata

bahasa_Indonesia	bahasa_Minang1	bahasa_Minang2	Jenis_kata
Abang	Uda	Udo	kata ganti
Bagaimana	Baa		kata tanya
Cambang	Gurambéh		kata benda
Daftar	Daftar	Ranji	kata kerja
Engkau	Waang		Kata ganti
.....	.....	.....	.....

Perancangan Algoritma *Rule Based* dalam aplikasi ini terdiri atas beberapa algoritma, yaitu :

1. Algoritma 1 : Procedure GetCh()

Hal pertama yang harus dilakukan dalam pembuatan aplikasi penerjemah adalah membuat prosedur untuk memindah pita karakter dan membaca isi dari pita karakter tersebut serta mengubahnya menjadi bentuk konsonan dan vokal. Pita karakter adalah kumpulan dari karakter-karakter yang dibaca oleh mesin karakter secara karakter per karakter sampai menemui akhir dari pita tersebut. Dalam hal ini teks berperan sebagai pita karakter. Teks akan terus dibaca sampai pita menemui akhir dari file (EOF).

```

KAMUS GLOBAL
Index : integer { indeks dari pita karakter saat ini }
CC : char { isi dari pita karakter saat ini }
k_EOF : boolean { bendera status End Of File }
Kalimat : string { pita karakter }
tempPola : string { peubah untuk pola konsonan dan vokal }

Procedure GetChar()
{ pita karakter maju satu karakter, kemudian membaca karakter tersebut }

DEKLARASI
function Mid(Str:string, posisi:integer, pjg:integer) → string
{ mengambil beberapa huruf dalam Str dari index posisi sebanyak pjg }
function tandabaca(input tanda: string) → boolean
{ mengembalikan nilai True jika variabel tanda merupakan tanda baca,
False jika bukan }
Function Lcase(Str:string) → string
{ mengubah Str menjadi berhuruf kecil }

DESKRIPSI
if CC ≠ Chr(3) then
    Index ← Index + 1
    CC ← Mid(Kalimat, Index, 1)
    if CC ≠ Chr(3) then
        k_EOF ← false
    else
        k_EOF ← true
    endif
    if Lcase(CC) = "a" or LCase(CC) = "i" or _
    LCase(CC) = "u" or LCase(CC) = "e" or _
    LCase(CC) = "o" then
        tempPola ← tempPola + "V"
    elseif CC = "#" or tandabaca(CC) or CC = " " then
        exit sub
    else
        tempPola ← tempPola + "K"
    endif
else
    k_EOF ← true
endif

```

Algoritma 1. Procedure GetChar

Kondisi `if LCase(CC) = "a" or LCase(CC) = "i" or LCase(CC) = "u" or LCase(CC) = "e" or LCase(CC) = "o" then tempPola ← tempPola + "V"` adalah kondisi dimana apabila karakter yang dibaca berupa huruf vokal (a, i, u, e, o) maka karakter tersebut akan diubah menjadi karakter huruf "V" (Vokal) dan ditampung di variabel tempPola. Namun, apabila karakter yang dibaca tidak berupa huruf vokal karakter tersebut akan diubah menjadi karakter huruf "K" (Konsonan).

2. Algoritma 2: Procedure AmbilKata()

<p>KAMUS GLOBAL</p> <p>CC : <u>char</u> { isi dari pita karakter saat ini }</p> <p>k_EOF : <u>boolean</u> { bendera status End Of File }</p> <p>kata : <u>string</u> { peubah untuk kata yang diambil }</p>
<p>Procedure AmbilKata()</p> <p>{ mengambil karakter per karakter dari input text , kemudian menampungnya di variabel kata }</p>
<p>DEKLARASI</p> <p><u>function</u> tandabaca(<u>input</u> tanda: <u>string</u>) → <u>boolean</u></p> <p>{mengembalikan nilai True jika variabel tanda merupakan tanda baca, False jika bukan }</p> <p><u>procedure</u> GetChar()</p> <p>{membaca satu karakter pada pita karakter}</p> <p>DESKRIPSI</p> <p><u>If</u> kata ← "" <u>then</u></p> <p>  <u>repeat</u></p> <p>    kata ← kata + CC</p> <p>    GetCh</p> <p>  <u>until</u> CC = " " <u>and</u> CC = Chr(10) <u>and</u> tandabaca(CC) _</p> <p>  <u>and</u> k_EOF</p> <p><u>endif</u></p>

Algoritma 2. Procedure untuk mengambil kata dalam input text

Algoritma digunakan untuk mengenali text yang di-input. Program akan mengambil karakter per karakter dan berhenti ketika bertemu dengan spasi, tandabaca dan akhir dari file.

3. Algoritma 3: Procedure SukuDepanMng()

<p>KAMUS GLOBAL</p> <p>KataDasar : <u>string</u> { peubah untuk kata dasar }</p> <p>SukuAwal : <u>string</u> {peubah untuk suku kata bagian awal }</p>
<p>Procedure SukuDepanMng()</p> <p>{ mengambil 2 huruf awal dari kata, mengecek apakah kata tersebut memiliki huruf vokal "e" atau tidak, jika ada maka vokal tersebut diubah menjadi "a" }</p>
<p>DEKLARASI</p> <p><u>function</u> Mid(Str:<u>string</u>, posisi:<u>integer</u>, pjl:<u>integer</u>) → <u>string</u></p> <p>{mengambil beberapa huruf dalam Str dari index posisi sebanyak pjl}</p> <p><u>Function</u> Right(Str:<u>string</u>, pjl:<u>integer</u>) → <u>string</u></p> <p>{mengambil beberapa string dalam Str dari kanan sebanyak pjl }</p> <p><u>Function</u> Left(Str:<u>string</u>, pjl:<u>integer</u>) → <u>string</u></p> <p>{mengambil beberapa string dalam Str dari kiri sebanyak pjl }</p> <p>DESKRIPSI</p> <p>SukuAwal ← Mid(KataDasar,1,2)</p> <p><u>if</u> Right(SukuAwal, 1) = "e" <u>then</u></p> <p>  SukuAwal ← Left(SukuAwal,1) + "a"</p> <p><u>else</u></p> <p>  SukuAwal ← ""</p> <p><u>exitsub</u></p> <p><u>endif</u></p>

Algoritma 3. Procedure SukuDepanMng

Algoritma di atas adalah algoritma untuk memeriksa suku kata yang terletak pada bagian awal kata. Fungsi  $SukuAwal \leftarrow Mid(KataDasar, 1, 2)$  merupakan fungsi untuk mengambil dua karakter dari KataDasar dimulai dari posisi index ke-1. Kondisi  $if\ Right(SukuAwal, 1) = "e"$  then  $SukuAwal \leftarrow Left(SukuAwal, 1) + "a"$  adalah kondisi untuk mengenali karakter pertama dari kanan pada SukuAwal. Jika karakter tersebut adalah "e" maka karakter tersebut diubah menjadi "a" dan ditampung di variabel suku awal.

4. Algoritma 4. Procedure SukuBlkgMng()

<p>KAMUS GLOBAL</p> <p>KataDasar : <u>string</u> { peubah untuk kata dasar }</p> <p>SukuAkhir : <u>string</u> { peubah untuk suku kata bagian akhir }</p> <p>tempPola : <u>string</u> { peubah untuk pola konsonan dan vokal }</p> <p>Akhir1 : <u>string</u> { peubah untuk beberapa string yang diambil dari kanan dari KataDasar }</p>
<p>Procedure SukuDepanMng()          { mengambil suku kata bagian akhir dari KataDasar kemudian diubah menjadi suku akhir pada bahasa tujuan }</p>
<p>DEKLARASI</p> <p>Function <u>Right</u>(Str:<u>string</u>, pjg:<u>integer</u>) → string          {mengambil beberapa string dalam Str dari kanan sebanyak pjg }</p> <p>Function <u>Left</u>(Str:<u>string</u>, pjg:<u>integer</u>) → string          {mengambil beberapa string dalam Str dari kiri sebanyak pjg }</p> <p>DESKRIPSI</p> <p><u>if</u> <u>Right</u>(tempPola, 4) = "KVKK" <u>then</u>              Akhir1 ← <u>Right</u>(KataDasar, 4)              akhir2 ← <u>Right</u>(Akhir1, 3)              <u>case</u> (akhir2)                  akhir2 = "ung" : ubah = "uang"                  akhir2 = "ing" : ubah = "iang"                  akhir2 = "eng" : ubah = "eang"                  akhir2 = "ong" : ubah = "oang"              <u>endcase</u></p> <p>    <u>if</u> ubah = "" <u>then</u>                  Akhir1 ← ""                  akhir2 ← ""              <u>exitsub</u>              <u>else</u>                  SukuAkhir ← <u>Left</u>(Akhir1, 1) + ubah              <u>endif</u></p> <p>    <u>elseif</u> <u>Right</u>(tempPola, 3) = "KVK" <u>then</u>                  Akhir1 ← <u>Right</u>(KataDasar, 3)                  akhir2 ← <u>Right</u>(Akhir1, 2)                  <u>case</u> (akhir2)                      akhir2 = "al" : ubah = "a"                      akhir2 = "ad" : ubah = "aik"                      akhir2 = "ar" : ubah = "a"                      .....                      .....                  <u>endcase</u></p> <p>    <u>if</u> ubah = "" <u>then</u>                  Akhir1 ← ""                  akhir2 ← ""              <u>exitsub</u></p>

```

else
  SukuAkhir ← Left(Akhir1, 1) + ubah
endif

elseif Right(tempPola, 2) = "VK" Then
  Akhir1 ← Right(KataDasar, 2)
  case (akhir2)
    akhir2 = "al" : ubah = "a"
    akhir2 = "ad" : ubah = "aik"
    akhir2 = "ar" : ubah = "a"
    akhir2 = "as" : ubah = "eh"
    .....
  endcase

  if ubah = "" then
    Akhir1 ← ""
    akhir2 ← ""
  exitsub
  else
    SukuAkhir ← ubah
  endif

elseif Right(tempPola, 2) = "KV" then
  Akhir1 ← Right(KataDasar, 2)
  if Right(Akhir1, 1) = "a" then
    SukuAkhir ← Left(Akhir1, 1) + "o"
  else
    Akhir1 ← ""
  exitsub
  endif

elseif Right(tempPola, 2) = "VV" then
  Akhir1 ← Right(KataDasar, 2)
  if Right(Akhir1, 1) = "a" then
    SukuAkhir ← Left(Akhir1, 1) + "o"
  else
    Akhir1 ← ""
  exitsub
  endif
endif

```

Algoritma 4. Procedure SukuBlkgMng()

Algoritma di atas merupakan algoritma untuk mengubah suku kata pada bagian akhir dari bahasa sumber menjadi suku kata bentuk bahasa tujuan. Proses yang berjalan pada algoritma di atas yaitu program membaca string pada variabel tempPola kemudian mengambil beberapa karakter dari kanan untuk diperiksa apakah sesuai dengan kondisi yang ada. Kondisi-kondisi tersebut antara lain  $\text{Right}(\text{tempPola}, 4) = \text{"KVKK"}$ ,  $\text{Right}(\text{tempPola}, 3) = \text{"KVK"}$ ,  $\text{Right}(\text{tempPola}, 2) = \text{"VK"}$ ,  $\text{Right}(\text{tempPola}, 2) = \text{"KV"}$  dan  $\text{Right}(\text{tempPola}, 2) = \text{"VV"}$ . Setelah ditemukan pola yang sesuai dengan kondisi tersebut, kemudian program akan membaca string KataDasar untuk mengambil beberapa karakter dari string tersebut kemudian ditampung dalam variabel Akhir1. Dari string Akhir1 diambil lagi beberapa karakter untuk diperiksa apakah sesuai dengan aturan yang ada, jika ada maka string tersebut diubah menjadi pola bentuk bahasa tujuan. Adapun aturan-aturan tersebut dapat dilihat pada table dibawah ini : Kondisi  $\text{Right}(\text{tempPola}, 4) = \text{"KVKK"}$

Tabel 6. Tabel *rule* untuk Suku Kata Akhir yang memiliki Pola "KVKK"

Kondisi akhiran pada bahasa sumber	Diubah menjadi
Ing	iang
Eng	eang
Ong	oang
Ung	uang

Kondisi Right(tempPola, 3) = "KVK"

Tabel 7. Tabel *rule* untuk Suku Kata Akhir yang memiliki Pola "KVK"

Kondisi akhiran pada bahasa sumber	Diubah menjadi
Ad	aik
Al	a
Ap	ok
Ar	a
As	eh
At	aik
Ik	iak
Il	ia
Ir	ia
Is	ih
It	ik
Uh	uah
Uk	uak
Us	ui
Ut	uik

Untuk kondisi Right(tempPola, 2) = "VK" *rule*-nya sama dengan *rule* pada kondisi di atas. Kondisi Right(tempPola, 2) = "KV" jika akhiran pada bahasa sumber adalah "a" maka pada bahasa tujuan diubah menjadi "o". Sama halnya pada kondisi Right(tempPola, 2) = "VV".

5. Algoritma 5. Prosedur Tengah()

<p>KAMUS GLOBAL</p> <p>KataDasar : <u>string</u> { <i>peubah untuk kata dasar</i> }</p> <p>SukuAwal : <u>string</u> { <i>peubah untuk suku kata bagian Awal</i> }</p> <p>SukuTengah : <u>string</u> { <i>peubah untuk suku kata bagian tengah</i> }</p> <p>Akhir1 : <u>string</u> { <i>peubah untuk beberapa string yang diambil dari kanan dari KataDasar</i> }</p>
<p>Procedure Tengah()</p> <p>{ <i>mengambil beberapa huruf dari KataDasar dari posisi panjang SukuAwal+1 sebanyak hasil pengurangan dari panjang KataDasar dengan panjang SukuAwal dan panjang Akhir1</i> }</p>
<p>DEKLARASI</p> <p><u>Function</u> Mid(Str:<u>string</u>, index:<u>integer</u>, n:<u>integer</u>) → <u>string</u>  <i>{mengambil beberapa huruf dalam Str dari posisi ke index sebanyak n}</i></p> <p><u>Function</u> Len(Str:<u>string</u>) → <u>integer</u>  <i>{mengembalikan nilai panjang Str}</i></p> <p>DESKRIPSI</p> <p><u>if</u> Len(KataDasar) - Len(SukuAwal) - Len(Akhir1) = 0 <u>then</u>              SukuTengah = ""  <u>else</u>              SukuTengah ← Mid(KataDasar, Len(SukuAwal) + 1, ...                  Len(KataDasar) - Len(SukuAwal) - Len(Akhir1))  <u>endif</u></p>

Algoritma 5. Prosedur Tengah()

Algoritma di atas digunakan untuk mengambil suku tengah dari KataDasar yaitu dengan cara mengambil beberapa huruf dari KataDasar dari posisi panjang SukuAwal+1 sebanyak hasil pengurangan dari panjang KataDasar dengan panjang SukuAwal dan panjang Akhir1.

6. Algoritma 6. Prosedur digabung()

<p><b>KAMUS GLOBAL</b>          SukuAwal : <i>string</i> (peubah untuk suku kata bagian Awal)          SukuTengah : <i>string</i> (peubah untuk suku kata bagian tengah)          SukuAkhir : <i>string</i> (peubah untuk suku kata bagian akhir)          gabung : <i>string</i> (peubah untuk hasil penggabungan)</p>
<p><b>Procedure Tengah()</b>          { menggabungkan SukuAwal, SukuTengah, dan SukuAkhir }</p>
<p><b>DEKLARASI</b>  <b>DESKRIPSI</b>          gabung ← SukuAwal + SukuTengah + SukuAkhir</p>

Algoritma 6. Prosedur untuk menggabungkan setiap suku kata

Berikut ditampilkan implementasi dari aplikasi terjemahan. Halaman utama dari aplikasi ini memiliki tampilan seperti pada gambar berikut :



Gambar 3. Tampilan Halaman Utama Aplikasi

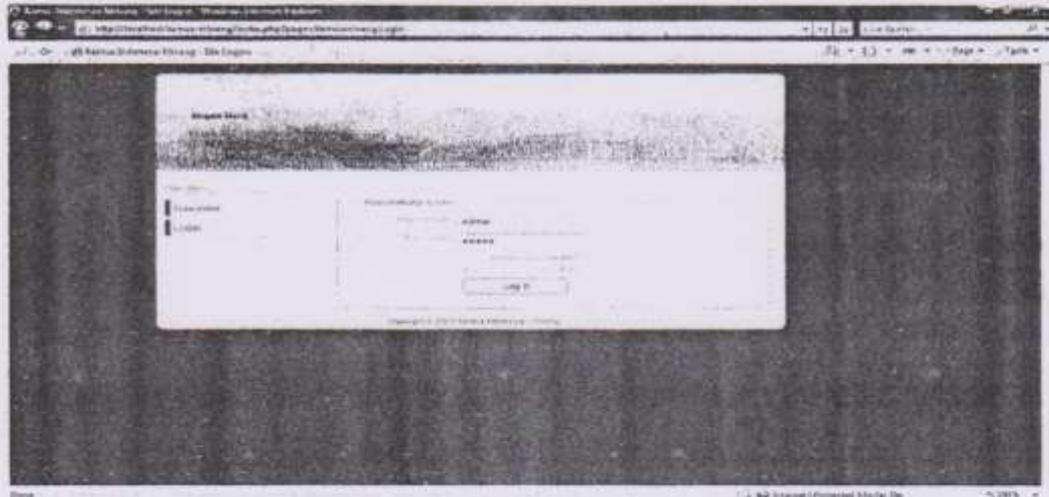
Pada halaman ini terdapat menu Translator dan Login. Bila dipilih menu Translator maka akan tampil gambar berikut :



Gambar 4. Tampilan Halaman Translator

Pada halaman *Translator* ini pengguna dapat melakukan proses penerjemahan teks dari bahasa Indonesia ke bahasa Minang. Pengguna dapat langsung mengetikkan teks yang akan diterjemahkan lewat *keyboard* atau dapat pula membuka *file* yang tersimpan di *harddisk* pengguna dengan menekan tombol "Browse" dan "Open File".

Halaman admin ini digunakan oleh admin untuk meng-*update*.



Gambar 5. Halaman Login

Pada halaman admin terdapat beberapa fasilitas, antara lain, yaitu : Daftar Kata, Tambah Kata dan Rule. Bila dipilih Daftar Kata akan ditampilkan halaman yang menampilkan seluruh kata yang ada dalam database. Disediakan beberapa pilihan daftar kata, seperti yang terlihat di gambar bawah :



Gambar 6. Halaman Daftar Kata

Berikut halaman daftar kata semua jenis kata :



Gambar 7. Halaman Daftar Kata Semua Jenis Kata

Berikut halaman daftar kata berdasarkan jenis kata yang dipilih :



Gambar 8. Halaman Daftar Kata Jenis Kata Benda

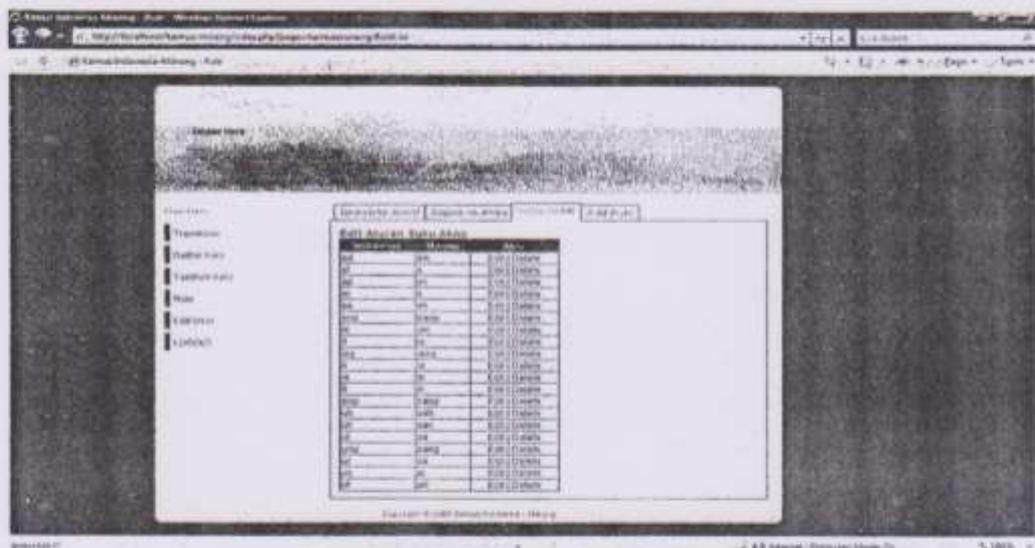
Pada halaman admin juga terdapat fasilitas Tambah Kata seperti berikut :



Gambar 9. Tambah Kata

Fasilitas lainnya yang terdapat pada halaman admin yaitu untuk rule atau aturan yang terdapat pada terjemahan ini, sehingga isi dari aplikasi tidak tergantung pada basis data, namun

dibuat dalam satu aturan tertentu yang dapat dengan mudah ditambah, diubah atau dihapus oleh admin.



Gambar 10. Halaman Rule Perubahan Terjemahan

## KESIMPULAN

Dari penelitian yang dilakukan maka telah dibuat aplikasi terjemahan bahasa Indonesia – Minang menggunakan pendekatan metode *rule based* berbasis web. Metode *rule based* dipakai bertujuan untuk mengantisipasi keterbatasan jumlah kata yang terdapat dalam basis data, yang mana kata-kata dalam bahasa Indonesia yang memiliki kemiripan dalam bahasa Minangnya dibuatkan dalam pola aturan tertentu. Aplikasi terjemahan bahasa Indonesia – Minang ini dapat menerjemahkan kata, frase dan kalimat baik yang diinputkan langsung oleh *user* ataupun berupa file.

Demi kesempurnaan dari penelitian ini, maka aplikasi ini dapat dikembangkan dengan fasilitas terjemahan dari bahasa Minang ke bahasa Indonesia, lebih lanjut dengan menambah kemampuan untuk menerjemahkan kalimat-kalimat yang kompleks yang memiliki imbuhan di awal dan di akhir kata. Aplikasi juga dapat dikembangkan sebagai program untuk belajar *vocabulary* bahasa Minang, misalnya dengan menambahkan fasilitas *text to voice converter* untuk belajar pengucapan yang benar dari kata-kata bahasa Minang.

## DAFTAR PUSTAKA

- [1] Arman, Arry Akhmad, 2004, *Teknologi Pemrosesan Bahasa Alami sebagai Teknologi Kunci untuk Meningkatkan Cara Interaksi antara Manusia dengan Mesin*, <http://www.itb.ac.id>.
- [2] Gorys Keraf, Dr, 1984, *Tata Bahasa Indonesia*, Flores:Nusa Indah.
- [3] Gouzali Saydam, Drs, 2004, *Kamus Lengkap Bahasa Minang*, Pusat Pengkajian Islam dan Minangkabau, Padang-Sumatra Barat.
- [4] Mulyanto, 2003, *Perangkat Lunak Penerjemah Bahasa Indonesia-Inggris Menggunakan Pencocokan Tata Bahasa Bagi Pelajar Tingkat Dasar*, Tugas Akhir-S1, Universitas Gadjah Mada, Yogyakarta.
- [5] Mustikawati, Tri Atmi, 2004, *Program Bantu Penerjemah Kata Bahasa Indonesia Ke Dalam Kata Bahasa Jawa*, Skripsi-S1, Universitas Ahmad Dahlan, Yogyakarta.
- [6] Sutopo, Hadi Ariesto, 2002, *Analisis dan Desain Berorientasi Objek*, J&J Learning, Yogyakarta.
- [7] Suryadi, 2006, *Menyelamatkan Kekayaan Bahasa Minangkabau: Tantangan Ahli Bahasa dan Pekamus*, <http://www.Ranah-Minang.com>.
- [8] ..... 2006, *Belajar Loghat/Bahasa Minangkabau*. <http://www.sriandalas.multiply.com>
- [9] ..... 2008. *Bahasa Minangkabau*. <http://www.wikipedia.com>