

APLIKASI KAMUS ELEKTRONIK INGGRIS – INDONESIA - SPANYOL MENGGUNAKAN FUNGSI PEMECAH KATA DAN MANIPULASI *STRING* PADA METODE PENCARIAN

Indah Puspita*, Aghus Sofwan** Agung Budi P **

Jurusan Teknik Elektro Fakultas Teknik Universitas Diponegoro

Abstrak - Komunikasi merupakan kunci utama untuk menaklukkan era globalisasi ini. Dan tidak dapat dipungkiri bahwa bahasa adalah bagian terpenting dari komunikasi. Seseorang dapat berkomunikasi dengan baik apabila menggunakan bahasa yang sama atau saling memahami bahasa yang digunakan satu sama lain. Untuk itu dibuatlah aplikasi berupa kamus elektronik.

Aplikasi kamus elektronik ini menggunakan fungsi pemecah kata dan manipulasi *string* dalam metode pencariannya, sehingga selain dapat mencari kata dengan tepat, juga mampu memberikan referensi kata yang sedekat mungkin dengan yang dimasukkan oleh pengguna apabila kata yang dicari tidak ditemukan.

Selain digunakan sebagai kamus bahasa Inggris –Indonesia - Spanyol, aplikasi ini , lewat pengembangan metode pencarian yang sama, dapat digunakan sebagai program pencari artikel. Program pencari artikel ini dirancang sebagai penampung referensi yang dapat mendukung pembelajaran ketiga bahasa tersebut.

Kata kunci : fungsi pemecah kata, manipulasi *string*, metode pencarian.

I. PENDAHULUAN

1.1 Latar Belakang

Komunikasi merupakan kunci utama untuk menaklukkan era globalisasi ini. Sudah tidak dapat dipungkiri lagi bahwa bahasa adalah bagian terpenting dari komunikasi. Seseorang dapat berkomunikasi dengan baik apabila menggunakan bahasa yang sama atau saling memahami bahasa yang digunakan satu sama lain.

Karena mustahil bagi semua orang yang berada pada berbagai belahan dunia untuk memiliki bahasa yang sama, satu-satunya jalan adalah dengan belajar mengerti bahasa yang digunakan oleh orang lain. Hal ini akan memakan banyak waktu dan biaya, untuk itulah dalam tugas akhir ini, penulis akan membangun suatu aplikasi kamus elektronik Inggris – Indonesia – Spanyol yang dapat mengefisienkan hal tersebut.

Pemilihan bahasa Inggris dan Spanyol dalam perancangan kamus ini dikarenakan kedua bahasa tersebut merupakan bagian dari lima bahasa utama yang paling sering digunakan di seluruh dunia, sedangkan bahasa Indonesia adalah bahasa resmi yang digunakan oleh penulis.

Aplikasi kamus elektronik ini dirancang menggunakan fungsi pemecah kata dan manipulasi *string* dalam metode pencariannya, sehingga selain dapat mencari kata dengan tepat, juga mampu memberikan referensi kata yang sedekat mungkin dengan yang dimasukkan oleh pengguna apabila kata yang dicari tidak ditemukan.

Selain digunakan sebagai kamus bahasa Inggris – Indonesia - Spanyol, aplikasi ini , lewat pengembangan metode pencarian yang sama, dapat digunakan sebagai program pencari artikel. Program pencari artikel ini dirancang sebagai penampung referensi yang dapat mendukung pembelajaran ketiga bahasa tersebut.

1.2 Tujuan Tugas Akhir

Tujuan yang ingin dicapai pada tugas akhir ini adalah merancang dan membangun suatu aplikasi kamus elektronik Inggris – Indonesia – Spanyol menggunakan fungsi pemecah kata dan manipulasi *string* pada metoda pencariannya, sehingga selain dapat mencari kata dengan tepat, juga mampu memberikan referensi kata yang sedekat mungkin dengan kata yang dimasukkan oleh pengguna apabila kata yang dicari tidak ditemukan.

1.3 Pembatasan Masalah

Dalam penyusunan tugas akhir ini, penulis membatasi permasalahan sebagai berikut:

1. Aplikasi ini , form utamanya terdiri dari empat menu utama yaitu menu file, menu aplikasi, menu window dan menu help.
2. Inti dari kamus elektronik ini terdapat pada menu aplikasi, yang didalamnya terdapat program kamus elektronik Inggris –Indonesia-Spanyol dan program pencari artikel.
3. Pada menu yang lain terdapat form – form yang berfungsi sebagai form pembantu atau form pendukung dari kedua program diatas.
4. Program pencari artikel diadakan untuk menampilkan fungsi lain dari dasar pengembangan metoda pencarian yang sama, juga berfungsi untuk menampung referensi artikel yang dapat membantu pembelajaran dari ketiga bahasa yang diangkat.
5. Kamus elektronik ini dapat menampilkan file gambar, file suara berekstensi *.wav dan pada program pencari kumpulan artikel dapat menampilkan file berekstensi *.pdf.
6. Aplikasi kamus elektronik ini menggunakan empat metoda area pencarian yang didalamnya menggunakan fungsi pemecah kata dan manipulasi *string*
7. Aplikasi ini diuji menggunakan Sistem Operasi Windows 98.
8. Bahasa pemrograman yang digunakan adalah Visual Basic 6.0.
9. Basis Data yang digunakan adalah Microsoft Access 2002.

II. LANDASAN TEORI

2.1 Kamus

Kamus adalah kumpulan dari berbagai kata - kata, istilah, sebutan ataupun peribahasa, baik dalam bentuk buku maupun aplikasi program yang dapat digunakan sebagai acuan untuk menerangkan makna dari suatu kata. Selain menerangkan maksud suatu perkataan, kamus juga dapat digunakan sebagai pedoman bagaimana suatu kata digunakan.

Berikut ini adalah jenis - jenis kamus yang banyak ditemui :

1. Berdasarkan penggunaan bahasa

Berdasarkan penggunaan bahasa, kamus dibagi menjadi tiga jenis yaitu :

a. Kamus Ekabahasa

Kamus ini hanya menggunakan satu bahasa, baik kata - kata yang dijelaskan maupun penjelasannya terdiri dari bahasa yang sama. Perbedaan kamus ini dengan kamus dwibahasa terletak pada penyusunannya dibuat berdasarkan pembuktian data, ini berarti bahwa definisi makna kata - kata berdasarkan makna yang diberikan dalam contoh kalimat yang mengandung kata - kata yang berhubungan. Contoh : Kamus Besar Bahasa Indonesia.

b. Kamus Dwibahasa

Kamus ini menggunakan dua bahasa, yang berarti kata - kata yang dijelaskan diberi penjelasan dalam bahasa yang berbeda. Tujuan dari kamus ini adalah untuk mempelajari dua bahasa yang berbeda. Contoh : Kamus Oxford Inggris - Indonesia.

c. Kamus Aneka Bahasa

Kamus ini, minimal menggunakan tiga bahasa atau lebih. Contoh : Kamus Inggris - Indonesia - Spanyol.

2. Berdasarkan isi

a. Kamus Saku / Kamus Mini

Kamus ini berukuran sangat kecil, sehingga dapat disimpan dalam saku dan memiliki tebal kurang dari 2 cm.

b. Kamus Kecil

Kamus ini berukuran kecil walaupun tidak sekecil kamus saku, namun ia dengan mudah dapat dibawa kemana - mana.

c. Kamus Besar

Kamus ini berisi arti kata - kata dalam suatu bahasa secara lengkap, berukuran besar sehingga tidak mudah untuk dibawa kemana - mana.

3. Berdasarkan Bentuk Mediana

a. Kamus Konvensional

Kamus konvensional menggunakan media yang masih konvensional sehingga memiliki bentuk berupa buku atau kumpulan kertas.

b. Kamus Elektronik

Kamus elektronik menggunakan media seperti komputer sehingga memiliki bentuk berupa suatu program aplikasi di komputer.

4. Kamus Khusus / Kamus Istimewa

Kamus Khusus adalah kamus yang memiliki fungsi khusus seperti :

a. Kamus Istilah

Input Kamus ini berupa istilah khusus bagi bidang tertentu, fungsinya adalah untuk pembelajaran ilmiah.

b. Kamus Etimologi

Kamus yang menerangkan asal - usul suatu perkataan

dan maksud asalnya.

c. Kamus tesaurus

Kamus yang menerangkan makna suatu perkataan dengan memperhatikan penggunaan diksi.

d. Kamus Peribahasa

Kamus yang menerangkan arti dari suatu peribahasa.

e. Kamus Kamus Kata Nama Khas

Kamus yang menyimpan kata nama unuk atau khas seperti nama tempat, nama tokoh dan nama - nama institusi.

f. Kamus Terjemahan

Kamus yang menyediakan terjemahan dari suatu bahasa asing.

2.2 Dasar Aplikasi Kamus Elektronik

Aplikasi kamus elektronik digolongkan ke dalam jenis perangkat lunak komputer personal. Aplikasi ini pada dasarnya merupakan pemindahan dari kamus konvensional ke dalam media komputer dengan menggunakan sistem pencarian otomatis.

Dasar dari sebuah kamus elektronik terletak pada kelengkapan basis datanya dan kehandalan *search engine*-nya dalam mencari suatu kata. *Search engine* yang baik sangat jarang menggunakan satu algoritma pencarian saja, karena itu dalam kamus elektronik biasanya menggunakan beberapa algoritma pencarian yang disatukan untuk menghasilkan proses pencarian yang lebih akurat.

Berikut ini adalah beberapa algoritma pencarian kesamaan string, yang secara sebagian atau penuh sering digunakan sebagai dasar pada proses pencarian kata dalam suatu kamus elektronik :

1. Algoritma Boyer Moore

Pada algoritma ini pemeriksaan string dilakukan dari kanan ke kiri.

Contoh :

Langkah - langkah untuk mencari kata artist pada the-greatest-artist

a. Langkah pertama menyamakan posisi huruf terakhir dari input dengan kata yang dicari

N :

a	r	t	i	s	t
---	---	---	---	---	---

M :

t	h	e	-	g	r	e	a	t	e	s	t	-	a	r	t	i	s	t
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

*

b. Kemudian diamati pada posisi paling akhir input, karena posisi ke 6 N tidak sama dengan posisi ke 6 M, maka posisi artist digeser ke kanan sebanyak 4 posisi, sampai huruf r dari kata artist sesuai dengan huruf r pada kata greatest.

N :

a	r	t	i	s	t
---	---	---	---	---	---

M :

t	h	e	-	g	r	e	a	t	e	s	t	-	a	r	t	i	s	t
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

X

c. Karena pada posisi ke 10 M (huruf e) tidak sama dengan huruf akhir kata artist, dan huruf tersebut tidak sama dengan semua huruf yang ada pada kata artist maka kata artist digeser ke kanan sebanyak 5 posisi.

a	r	t	i	s	t
---	---	---	---	---	---

N :

M : t h e - g r e a t e s t - a r t i s t

*

d. Pada posisi baru ini ini posisi terakhir dari kata artist sama dengan posisi urutannya, namun posisi kedua dari akhir tidak sama , maka kata artist digeser 3 posisi lagi. Akhirnya semua huruf yang ada pada kata artist sesuai dengan huruf pada teks.

N :

a r t i s t

M : t h e - g r e a t e s t - a r t i s t

2. Algoritma Knuth-Morris-Pratt

Algoritma ini mencocokkan string dari kiri ke kanan.

Langkah pertama :

N :

a r t i s t

M : t h e - g r e a t e s t - a r t i s t

Langkah kedua :

N :

a r t i s t

M : t h e - g r e a t e s t - a r t i s t

Langkah Ketiga :

N :

a r t i s t

M : t h e - g r e a t e s t - a r t i s t

Langkah Keempat :

N :

a r t i s t

M : t h e - g r e a t e s t - a r t i s t

Langkah Kelima :

N :

a r t i s t

M : t h e - g r e a t e s t - a r t i s t

2.3 Visual Basic 6.0 Sebagai Bahasa Pemrograman

Fungsi pemecah kata adalah suatu fungsi yang digunakan untuk memecah suatu *string* menjadi beberapa array baru berdasarkan salah satu karakter yang terdapat pada *string* tersebut.

Misal : suatu string yang terdiri dari 'aku ingin makan jeruk'

Jika ingin dipecah menggunakan fungsi pemecah kata berdasarkan spasi yang ada maka menggunakan perintah :

S = "aku ingin makan jeruk"

Split(s, " ")

Ini akan menghasilkan array :

S(0) =aku

S(1) =ingin

S(2) =makan

S(3) =jeruk

Lewat fungsi `ubound` atau fungsi `join` (fungsi penggabungan kata) maka array tersebut dapat ditangkap menjadi karakter *string* kembali.

Visual Basic menyediakan fungsi `split` sebagai fungsi pemecah kata dalam *defaultnya* namun apabila fungsi tersebut tidak ada , maka dapat diciptakan sendiri, lewat perintah berikut :

```
Dim i As Long
Dim lNextPos As Long
Dim sText As String
Dim lCount As Long
Dim varTemp() As String
For i = 1 To Len(Expression)
    lNextPos = InStr(i + 1, Expression, +_
Delimiter)
    If lNextPos = 0 Then
        lNextPos = Len(Expression) + 1
    End If
    sText = Mid$(Expression, i, lNextPos - i)
    ReDim Preserve varTemp(lCount)
    varTemp(lCount) = sText
    lCount = lCount + 1
    i = lNextPos + Len(Delimiter) - 1
Next
Split = varTemp
End Function
```

Fungsi manipulasi *string* adalah berbagai fungsi yang dapat mengubah *string* masukan menjadi suatu keluaran yang baru. Dengan fungsi- fungsi ini *string* dapat diubah ke dalam bentuk lain , atau dapat ditangkap berdasarkan beberapa susunan karakternya. Untuk memanipulasi *string*, Visual Basic menyediakan banyak fungsi - fungsi standar dalam *default* referensinya

Tiga diantaranya adalah :

1. `String ()` , untuk menangkap karakter paling depan pada suatu string, dan mengulangi karakter itu sebanyak konstanta yang diinginkan.
2. `Right ()` , untuk menangkap karakter dari posisi paling kanan sampai urutan yang diinginkan.
3. `Left ()` , untuk menangkap karakter dari posisi paling kiri sampai urutan yang diinginkan.

III. PERANCANGAN PERANGKAT LUNAK

Perumusan Masalah

Kamus elektronik ini dirancang dengan menggunakan empat metoda pencarian, yaitu :

1. Match case

Dalam metoda ini, akan dicari kata yang sama persis dengan kata yang dimasukkan oleh pengguna. Contoh : jika pengguna memasukkan kata 'apa kabar'(yang diantara kata apa dan kata kabar dibatasi hanya oleh satu spasi), maka program akan mencari dalam basis data kata 'apa kabar'(dengan satu spasi di tengah kata), perbedaan jumlah spasi di tengah kata dalam metoda ini akan sangat berpengaruh.

2. Large case

Pada metoda large case, akan dicari kata yang didepannya mengandung kata yang dimasukkan oleh pengguna. Contoh : jika pengguna memasukkan kata 'abad', maka program akan mencari dalam basis data kata yang didepannya terkandung kata 'abad' (abad%) . Sehingga bila kata 'abad' tidak terdapat dalam basis data,

namun terdapat kata 'abadi' dan kata 'abadilah', maka kedua kata ini yang akan ditampilkan dalam hasil pencarian.

3. Wider case

Metoda ini akan mencari kata yang mengandung kata yang dimasukkan oleh pengguna (baik itu didepan, ditengah maupun dibelakang). Contoh : jika pengguna memasukkan kata 'abad', maka program akan mencari kata yang mengandung kata 'abad' (%abad%). Sehingga bila kata 'abad' tidak terdapat dalam basis data, namun terdapat kata 'perabadian' dan 'abadilah', maka kedua kata itulah yang akan ditampilkan dalam hasil pencarian.

4. Nearest Case

Metoda ini lewat fungsi manipulasi *string* akan menghilangkan dari belakang, huruf/karakter dari kata yang dicari satu demi satu, sampai ditemukan kata dalam basis data yang didepannya mengandung kata tersebut. Contoh : apabila mencari kata 'mildest' dan dalam basis data terdapat kata 'milder', 'mild', 'milderania' maka proses pencarian akan seperti berikut : Akan dicari dalam basis data kata yang didepannya mengandung kata 'mildest'(mildest%), apabila tidak ditemukann maka kata 'mildest' hurufnya yang paling belakang (yaitu huruf 't') akan dihilangkan, sehingga menjadi kata 'mildes', Selanjutnya akan dicari kata yang didepannya mengandung kata mildes (mildes%) , bila tidak ditemukan maka huruf paling belakang dari kata baru itu (yaitu huruf 's') akan dihilangkan lagi sehingga menjadi kata 'milde', kemudian akan dilanjutkan dengan mencari kata yang didepannya mengandung kata milde(milde%) , pengurangan atau penghilangan kata tersebut akan terus dilakukan sampai menemukan kata yang dicari atau sampai tertinggal satu huruf terakhir. Pada pencarian ini akan menampilkan kata 'milder' dan 'milderania' sebagai hasil akhir.

3.1.1 Permasalahan Pada Large dan Wider Case

Permasalahan selanjutnya adalah ketika kata yang dicari terdiri lebih dari satu patah kata, contoh : selamat siang dan pagi – pagi buta . Kedua kata ini bila dicari dengan metoda *match case* yang berprinsip pencarian identik, tidak akan menimbulkan perbedaan yang besar. Namun bila dicari dengan metoda *large case* atau *wider case* akan memberikan dampak perbedaan pencarian yang sangat besar bila dibandingkan dengan pencarian satu patah kata saja. Hal itu dikarenakan kata yang terdiri dari beberapa kata tersebut akan tetap dicari sebagai satu patah kata tunggal, ini menyebabkan tujuan penggunaan metoda tersebut untuk melebarkan area pencarian tidak akan tercapai.

Untuk mengatasi permasalahan tersebut digunakan fungsi pemecah kata atau yang disebut dengan fungsi *split()*, untuk memecah tiap patahan kata yang dibatasi oleh spasi, sehingga kata yang terdiri dari dua patah kata akan dicari sebagai dua patah kata dan kata yang terdiri dari tiga patah kata akan dicari sebagai tiga patah kata.

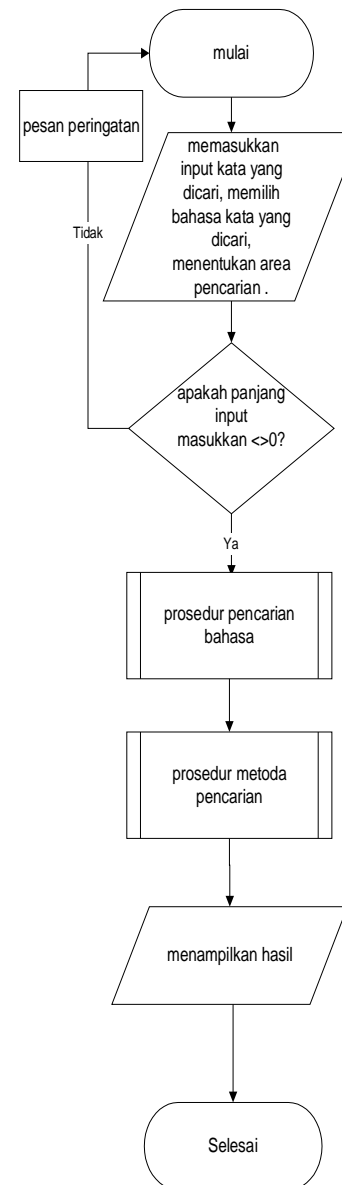
3.1.2 Permasalahan Pada Nearest Case

Dengan menggunakan metoda *nearest case* apabila kata yang dimasukkan oleh penggunanya terlalu

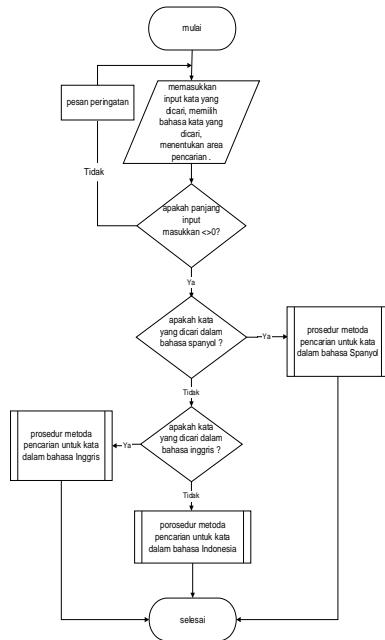
panjang, maka proses pencarian katanya akan memakan waktu yang lebih lama, oleh karena itu perlu ditentukan batas pencarian maksimal untuk metoda ini. Contoh: apabila pengguna memasukkan kata yang memiliki 1000 panjang kata , maka bila batas maksimal pencarian = 15, kata tersebut akan mulai dicari dari 15 huruf/karakter terdepan.

Pemodelan Aplikasi Kamus Elektronik Inggris-Indonesia- Spanyol

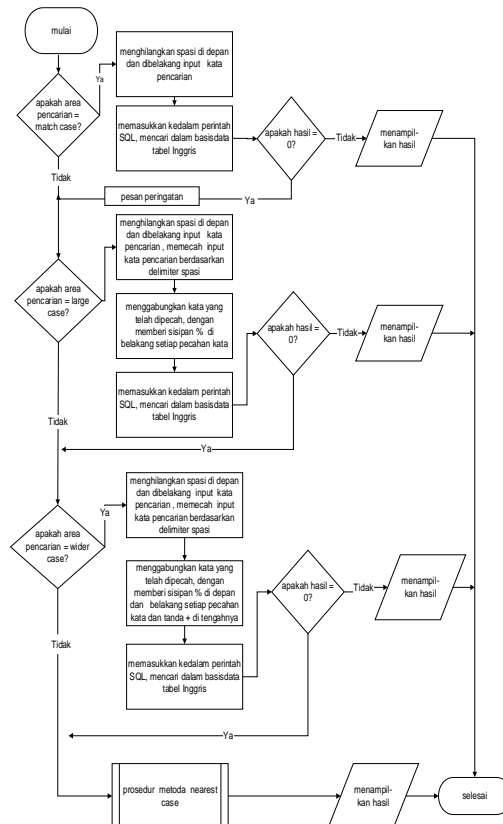
Pemodelan aplikasi ini digambarkan dalam diagram alir berikut ini :



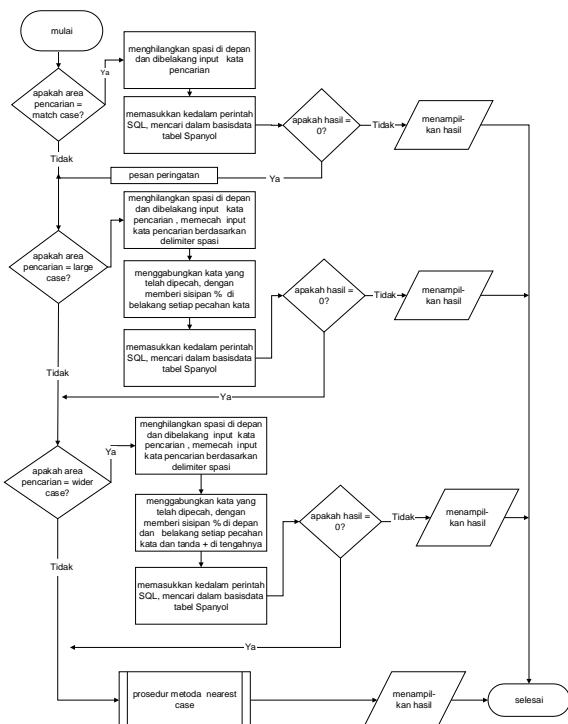
Gambar 3.1 Diagram Alir Proses Pencarian Kata Pada Kamus Secara Umum



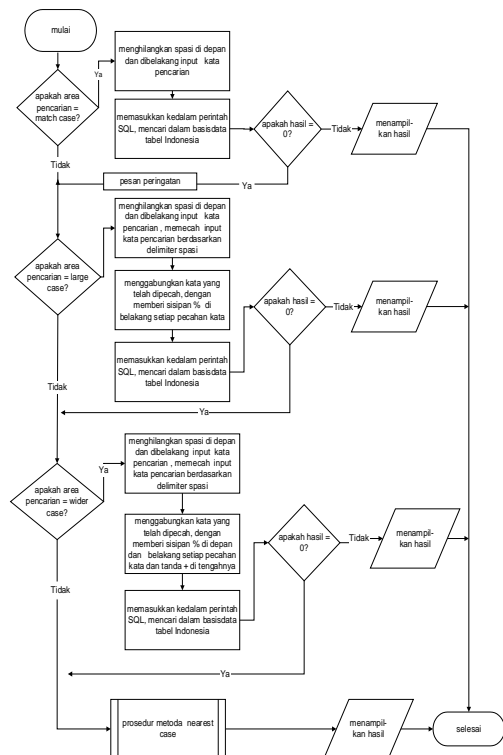
Gambar 3.2 Diagram Alir Penentuan Bahasa Pada Prosedur Pencarian Bahasa



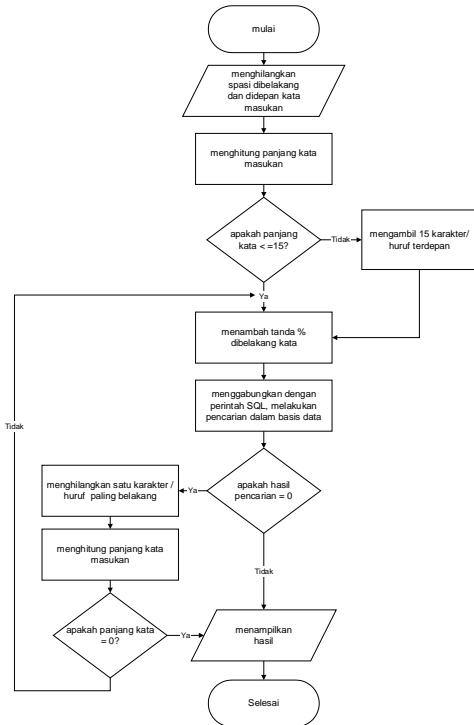
Gambar 3.4 Diagram Alir Prosedur Pencarian Kata Inggris Pada Kamus



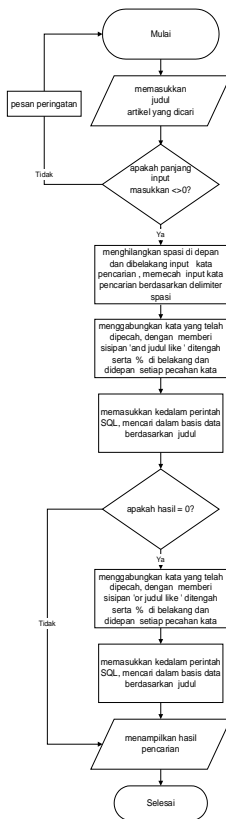
Gambar 3.3 Diagram Alir Prosedur Pencarian Kata Spanyol Pada Kamus



Gambar 3.5 Diagram Alir Prosedur Pencarian Kata Indonesia Pada Kamus



Gambar 3.6 Diagram Alir Prosedur Metoda Nearest Case
Diagram alir program pencari artikel :



Gambar 3.7 Diagram Alir Proses Pencarian Pada Program Pencari Artikel

IV. IMPLEMENTASI DAN PENGUJIAN

Pengujian aplikasi ini terdiri dari pengujian pernyataan kondisional, pengujian output dan pengujian waktu pencarian.

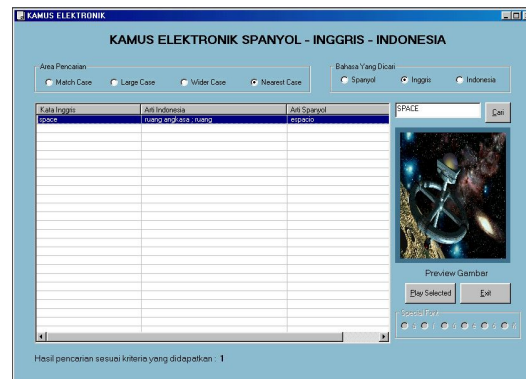
4.1 Konfigurasi Sistem

Pada pengujian kali ini digunakan satu komputer, dengan konfigurasi perangkat keras dan perangkat lunak sebagai berikut :

1. Prosesor Pentium II 400 MHz
2. RAM SDRAM 192 MB
3. Harddisk 40 GB Maxtor ATA 100
4. Keyboard & Mouse PS/2
5. Driver audio Yamaha opl3-sax wdw
6. Driver VGA Rendition Verite 2200 agp
7. Speaker Sound Blaster sebagai perangkat audio
8. Motherboard Intel BD100P
9. Sistem operasi Windows 98.
10. Microsoft Acces 2002 sebagai basis data
11. Acrobat Reader 4 sebagai pdf reader

4.2 Form Kamus Elektronik Inggris-Indonesia-Spanyol

Pada form kamus elektronik Inggris-Indonesia-Spanyol apabila salah satu metode tersebut tidak menemukan hasil akan diteruskan ke metode selanjutnya sampai metode terakhir atau sampai kata yang dicari ditemukan sesuai pendekatan metodenya.

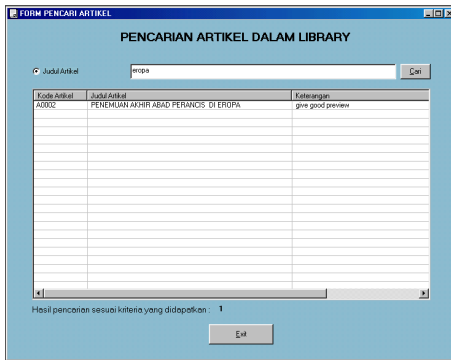


Gambar 4.1 Tampilan Form Kamus Elektronik Inggris - Indonesia - Spanyol

4.3 Form Pencari Artikel

Pada form Pencari Artikel dapat mencari judul artikel dengan mencari kata yang terkandung dalam masukan. Misal : pada contoh dibawah kata yang dicari adalah kata 'eropa perancis', dan hasil yang dikeluarkan adalah 'penemuan akhir abad perancis di eropa '

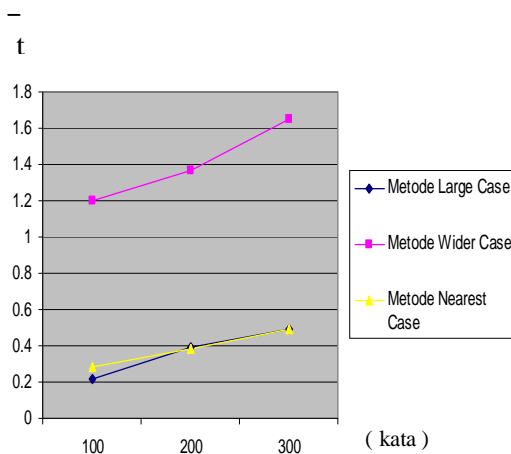
Jadi program mencari kata yang mengandung kata eropa dan mengandung kata perancis tanpa memperhatikan urutan patahan kata, yang berarti bahwa pada hasilnya nanti, kata eropa tidak harus terletak setelah kata perancis, hal ini berbeda dengan metoda pencarian wider case pada kamus



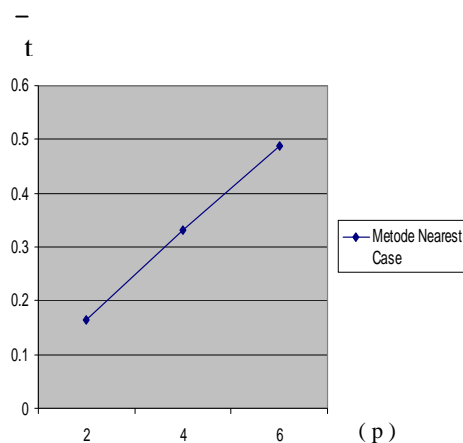
Gambar 4.2 Tampilan Form Pencari Artikel

4.4 Pengujian Waktu Pencarian

Selanjutnya diadakan pengujian terhadap waktu pencarian kata, pada metode nearest case, large case, dan wider case pada basis data yang memiliki 2000 perbendaharaan kata, untuk mengetahui hubungan antara waktu pencarian dengan proses pencarian.



Gambar 4.3 Grafik Rata - Rata Waktu Pencarian (t) Terhadap Banyak Kata yang memiliki kesamaan string



Gambar 4.4 Grafik Rata - Rata Waktu Pencarian (t) Terhadap Banyaknya Proses Pengurangan Karakter (p) Pada Metode Nearest Case

V. PENUTUP

5.1 Kesimpulan

1. Berdasarkan analisa grafik rata - rata waktu pencarian didapatkan bahwa pada metode large case wider case dan nearest case waktu pencariannya berbanding lurus dengan banyak kata yang memiliki kesamaan string.
2. Pada metode nearest case, waktu pencariannya juga berbanding lurus dengan banyaknya karakter yang harus dihilangkan sebelum menemukan kata yang dicari.
3. Aplikasi Kamus Elektronik Inggris - Indonesia - Spanyol yang dibangun sudah memenuhi tujuan perancangan yaitu selain dapat mencari kata dengan tepat, juga mampu memberikan referensi kata yang sedekat mungkin dengan kata yang dimasukkan oleh pengguna apabila kata yang dicari tidak ditemukan.

5.2 Saran

1. Diharapkan untuk selanjutnya dapat dikembangkan penelitian tentang teknik kompresi file pdf, file gambar dan file suara yang lebih baik sehingga dapat mengefisienkan kapasitas penyimpanan hard disk yang akan sangat membantu dalam pengembangan aplikasi ini.
2. Diharapkan agar fitur - fitur yang dapat ditampilkan di aplikasi ini dapat lebih dikembangkan .

DAFTAR PUSTAKA

1. Donald Knuth, James H. Morris, Jr, Vaughan Pratt (1977). "Fast pattern matching in strings". *SIAM Journal on Computing* 6 (2): 323-350.
2. Ibrahim bin Ahmad, *Perkamusan Melayu: Suatu Pengenalan*, Dewan Bahasa dan Pustaka, Kuala Lumpur, 2002.
3. Kadir, Abdul, *Konsep dan Tuntunan Praktis Basis Data*, Pt. Andi Yogyakarta, 1999.
4. Simanjuntak, Herpius, *Kamus Inggris - Spanyol - Indonesia*, Kesaint Blanc, Jakarta, 2004.
5. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, "Section 32.4: The Knuth-Morris-Pratt algorithm", *Introduction to Algorithms*, Second edition, MIT Press and McGraw-Hill, 2001.
6. Tim Penelitian dan Pengembangan Wahana Komputer, *Tutorial Membuat Program Dengan Visual Basic*, Salemba Infotek, Jakarta, 2004.
7. Wirth, Niklaus, *Algoritma + Struktur Data = Program*, Andi, Yogyakarta, 1997.

BIODATA MAHASISWA**Indah Puspita****L2F 002 587**

Lahir pada tanggal 21 April 1983 di Semarang, Menempuh pendidikan di SDN Krapyak 02 Semarang lulus tahun 1995, kemudian melanjutkan ke SMP Negeri 1 Semarang lulus tahun 1998. Lalu melanjutkan ke SMU Negeri 3 Semarang lulus tahun 2001, dan sekarang sedang menyelesaikan studi S1 pada Jurusan Teknik Elektro konsentrasi Informatika dan Komputer, Fakultas Teknik, Universitas Diponegoro Semarang.

Menyetujui dan mengesahkan

Pembimbing I

Aghus Sofwan, ST, MT

NIP.132 163 757

Pembimbing II

Agung Budi Prasetijo, ST, MIT

NIP. 132 137 932