

# Implementasi Metode *Top Down Parsing* pada Teknologi Bahasa Alamiah dalam Bentuk *Chatbot*

**YOULLIA INDRAWATY, YANDRI HARIANJA**

Jurusan Teknik Informatika, Fakultas Teknik Industri  
Institut Teknologi Nasional Bandung

Email: [youllia@itenas.ac.id](mailto:youllia@itenas.ac.id)

## **ABSTRAK**

*Chatbot adalah sebuah program komputer yang dirancang untuk mensimulasikan sebuah percakapan yang interaktif kepada user dalam bentuk teks. Chatbot dibangun dengan menggunakan metode top down parsing. Top down parsing digunakan untuk mencari kesimpulan dari pertanyaan user yang terdapat dalam database. Kemudian sistem akan mencari kata kunci berdasarkan pertanyaan user. Setelah kata kunci ditemukan maka akan dicocokkan dengan kata kunci yang terdapat dalam database. Kemudian sistem akan menampilkan jawaban yang sesuai dengan kata kunci yang terdapat dalam database. Hasil pengujian black box pada proses tanya jawab, sistem dapat menjawab masukan dari pengguna dengan baik sesuai dengan kata kunci yang ada di database. Hasil pengujian user acceptance test tanya jawab yang diajukan langsung kepada 10 mahasiswa, 70% chatbot mudah dioperasikan, 40% chatbot menarik bagi mahasiswa, dan 70% chatbot bermanfaat bagi mahasiswa untuk mendapatkan informasi yang diinginkan.*

## **ABSTRACT**

*Chatbot is a computer program designed to simulate an interactive conversation to the user in text form. Chatbot built using top down parsing. Top down parsing is used to find the conclusion of user questions contained in database. Then the system will search for keywords based on user questions. After the keywords are found then it will be matched with the keywords contained in database. Then the system will display the appropriate answers to the keywords contained in database. Black box testing results in the process of questioning answering system can answer with good input from the user according to the key words in database. The results of testing the user acceptance test questions and answers are submitted directly to 10 students, 70% chatbot easy to operated, 40% chatbot attractive to students, and chatbot useful for students to get needed.*

**Keyword:** *Chatbot, Top Down Parsing, Database*

## 1. PENDAHULUAN

Chatbot adalah sebuah program komputer yang dirancang untuk mensimulasikan sebuah percakapan atau komunikasi yang interaktif kepada user (manusia) melalui bentuk teks, suara, dan atau visual. Percakapan yang terjadi antara komputer dengan manusia merupakan bentuk respon dari program yang telah dideklarasikan pada database program pada komputer. Respon yang dihasilkan merupakan hasil pemindaian kata kunci pada kalimat masukan user dan menghasilkan respon balasan yang dianggap paling cocok, atau pola kata-kata yang dianggap paling mendekati pola kata-kata yang terdapat di database.

Chatbot dibangun dengan menerapkan teknologi bahasa alamiah dengan menggunakan metode top down parsing. Metode top down parsing merupakan sebuah proses yang menguraikan sebuah kalimat mulai dari constituent terbesar sampai menjadi constituent terkecil sampai semua komponen yang dihasilkan adalah constituent terkecil dalam kalimat berupa kata. Top down parsing digunakan untuk mencari kesimpulan dari pertanyaan user yang terdapat dalam database. Kemudian sistem akan mencari kata kunci berdasarkan pertanyaan user. Setelah kata kunci ditemukan maka akan dicocokkan dengan kata kunci yang terdapat dalam database. Kemudian sistem akan menampilkan jawaban yang sesuai dengan kata kunci yang terdapat dalam database.

Pada sistem ini percakapan dilakukan dalam Bahasa Indonesia yang sesuai dengan Ejaan Yang Disempurnakan (EYD), jawaban pertanyaan dari *user* hanya berupa teks kalimat. Kalimat masukan berupa pertanyaan dengan menggunakan kata tanya.

## 2. METODOLOGI PENELITIAN

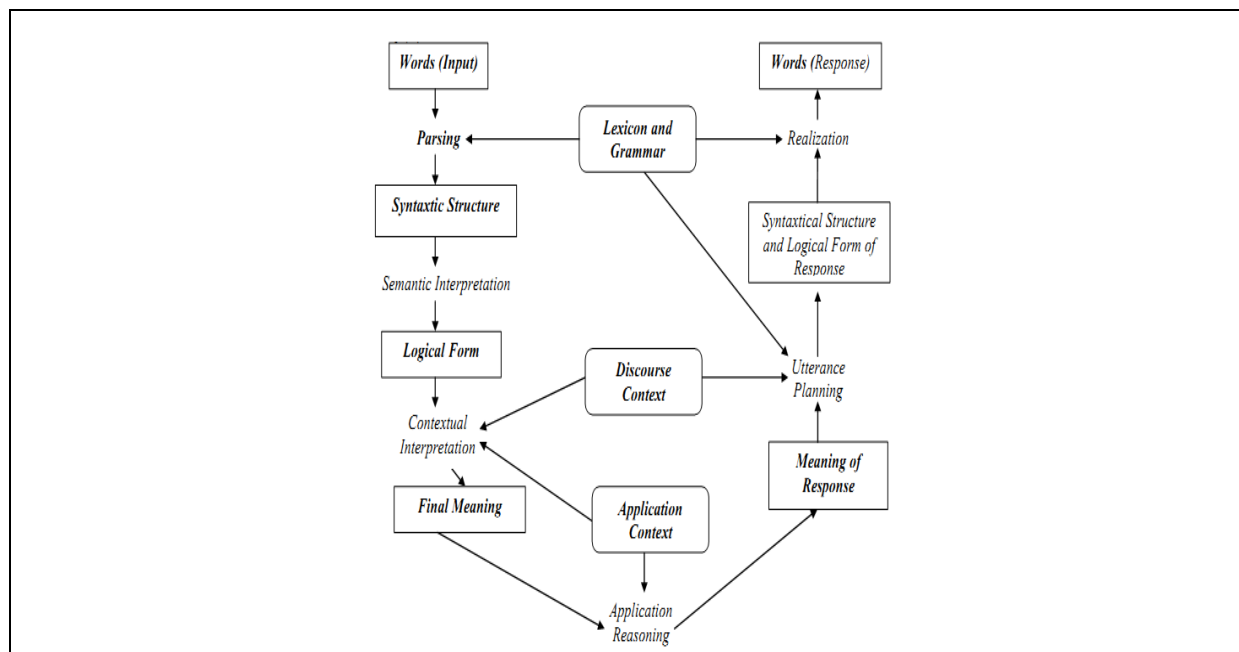
### 2.1 Chatbot

Chatbot adalah sebuah program komputer yang dirancang untuk mensimulasikan sebuah percakapan atau komunikasi yang interaktif kepada user (manusia) melalui bentuk teks, suara, dan atau visual. Percakapan yang terjadi antara komputer dengan manusia merupakan bentuk respon dari program yang telah dideklarasikan pada database program pada komputer. Respon yang dihasilkan merupakan hasil pemindaian kata kunci pada masukan user dan menghasilkan respon balasan yang dianggap paling cocok, atau pola kata-kata yang dianggap paling mendekati dengan kata kunci yang terdapat di database. Chatbot mampu meniru komunikasi manusia. Jika manusia sedang melakukan percakapan dengan program ini, seakan-akan ada 2 pribadi manusia yang saling berkomunikasi. Nyatanya, manusia berkomunikasi dengan Bot. Bot adalah program yang dirancang untuk merespon segala jenis pertanyaan dan pernyataan berdasarkan kalimat masukan dari user. Hal ini dapat dilakukan karena sebelumnya sudah dideklarasikan pada database, berupa entitas-entitas kata, pola kalimat, dan berbagai jenis pernyataan dan pertanyaan [1].

### 2.2 Bahasa Alamiah (*Natural Language*)

Bahasa alami adalah suatu bentuk representasi dari suatu pesan yang ingin dikomunikasikan antar manusia. Bentuk utama representasinya adalah berupa suara atau ucapan (*spoken language*), tetapi sering pula dinyatakan dalam bentuk tulisan. 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

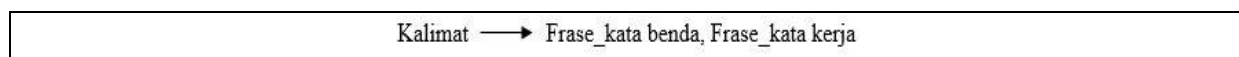
komputer [2]. Gambaran organisasi dari sebuah sistem Natural Processing Language yang lengkap ditunjukkan pada Gambar 1.



**Gambar 1. Organisasi Sebuah Sistem NLP**  
(Sumber : Fendy Ridwan, 2012)

### 2.3 Gramatika

Tata bahasa dalam bahasa alami menspesifikasikan struktur kalimat yang berfungsi untuk menentukan kategori sintaksis dasar seperti kata benda dan kata kerja, dan menentukan struktur dari kalimat dan tata bahasa. Dalam bahasa alami, kalimat tidaklah sering di parsing untuk mengetahui apakah kalimat tersebut benar atau tidak. Parsing digunakan untuk menentukan struktur dan membantu mencari arti dari sebuah kalimat. Tetapi kebanyakan tata bahasa mengacu pada pengaturan bahasa yang resmi, bila menggunakan kalimat dengan bahasa yang telah dimodifikasi, parsing akan menjadi semakin rumit [2]. Berikut merupakan aturan tata bahasa seperti pada Gambar 2.



**Gambar 2. Aturan Parsing**  
Sumber : (Astari kartika hadinata dan Novita, 2007)

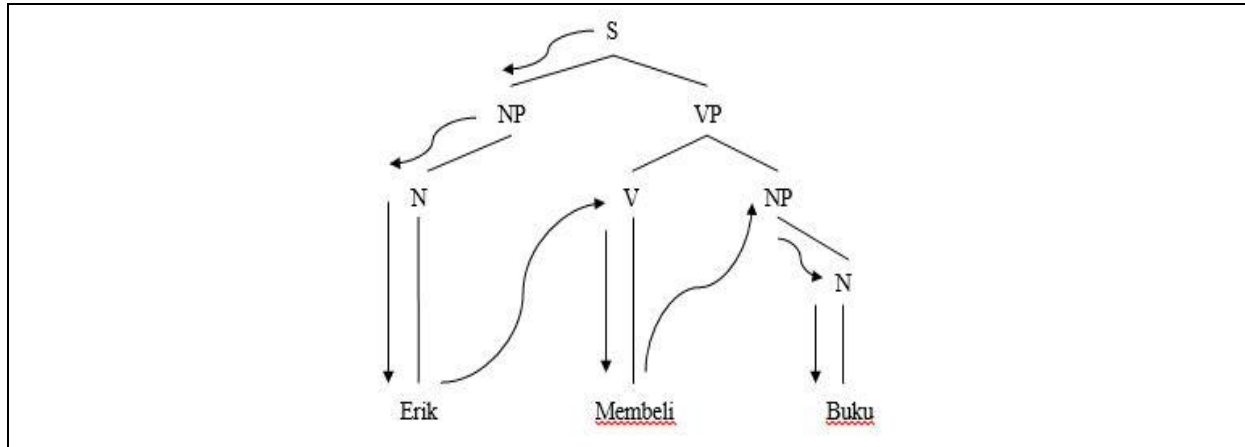
Dari keterangan Gambar 2 dapat dijelaskan bahwa jika kita memiliki sebuah frase\_kata benda, yang diikuti oleh sebuah frase\_kata kerja maka kita memiliki sebuah kalimat

### 2.4 Parsing

*Parsing* atau *syntactic analyzer* pada kompilator bahasa pemrograman berfungsi untuk memeriksa kebenaran kemunculan setiap kalimat. Proses *parsing* membagi-bagi kalimat menjadi kata per kata. Pada sistem tanya jawab, fungsi dari *parsing* berbeda karena kalimat yang akan diolah memiliki tipe yang sama yaitu berupa kata [3]. Urutan kemunculan kata akan diolah dengan mengacu pada *database* agar didapatkan makna kalimat yang sesungguhnya. Proses *parsing* menentukan pembuatan frase untuk kata, seperti kata\_kerja, kata\_keterangan, kata\_benda, dan seterusnya.

### 2.5 Top Down Parsing

*Top down parsing* bekerja dengan cara menguraikan sebuah kalimat mulai dari *constituent* yang terbesar sampai menjadi *constituent* yang terkecil. Hal ini dilakukan terus-menerus sampai semua komponen yang dihasilkan ialah *constituent* terkecil dalam kalimat yaitu kata [3]. Berikut merupakan contoh proses *top down parsing* seperti pada Gambar 3.



**Gambar 3. Top down parsing**  
**Sumber: (James Suciadi, 2001)**

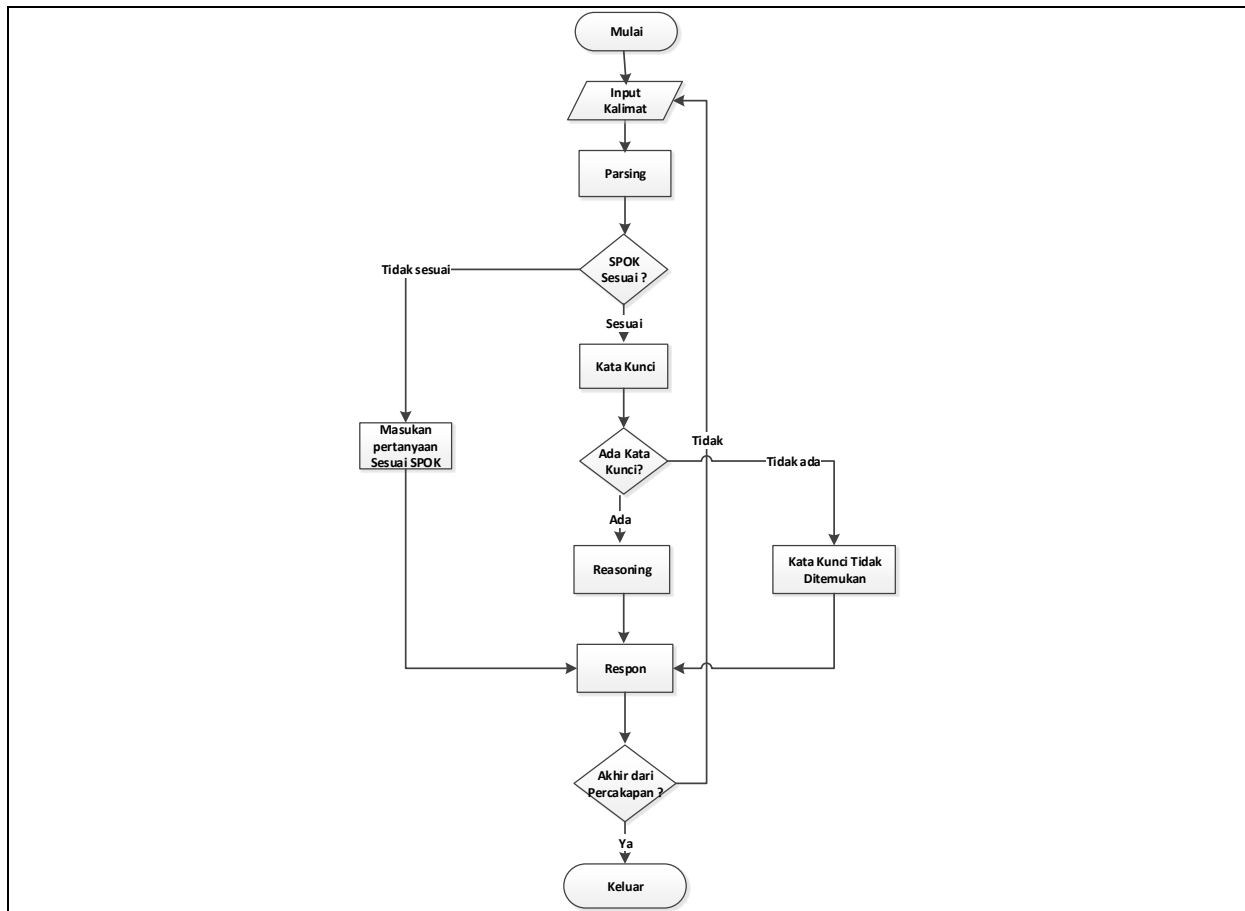
Proses top down parsing pada Gambar 3 dapat dijelaskan sebagai berikut:

- Keterangan: S = Kalimat (proses awal)  
 NP = Frase\_kata benda  
 VP = Frase\_kata kerja  
 S = Subjek  
 V = Kata kerja  
 N = Kata benda

## 3. ANALISIS DAN PEMBAHASAN

### 3.1 Perancangan aplikasi *chatbot*

Proses flowchart aplikasi *chatbot* dimulai dengan *user* memasukan kalimat pertanyaan. Kemudian kalimat pertanyaan tersebut dibagi menjadi beberapa kata dalam proses *parsing*. Setelah kalimat pertanyaan di *parsing* akan dilakukan proses SPOK. Dalam proses SPOK akan dilakukan pengecekan jenis kata dari kalimat pertanyaan hasil *parsing*. Kemudian hasil dari SPOK akan dicari kata kunci yang sesuai dengan kata kunci yang ada pada *database chatbot*. Apabila kata kunci ditemukan maka sistem akan melakukan proses *reasoning* sedangkan jika sistem tidak dapat menemukan kata kunci maka jawaban yang muncul tidak berhubungan dengan pertanyaan *user*. Dalam proses *reasoning* sistem akan menjawab pertanyaan *user* dengan jawaban *reasoning*. Berikut merupakan proses flowchart aplikasi *chatbot* seperti pada Gambar 4.



**Gambar 4. Flowchart aplikasi chatbot**  
**Sumber : (Astari kartika hadinata dan Novita, 2007)**

### 3.2 Proses Kalimat Masukan

Proses pendeteksian kalimat masukan yaitu dengan mengenali *input* yang dimasukan oleh *user* dan memasukkannya kedalam variabel penampung untuk kemudian dibagi-bagi ke dalam kelompok variable kata.

```

publicstring InputKalimat
{
get { return inputKalimat; }
set { inputKalimat = value; }
}
    
```

Pada proses ini kalimat masukan dari *user* adalah kalimat pertanyaan yaitu: "Siapa yang mengajar matakuliah logika fuzzy?". Kemudian kalimat pertanyaan diproses secara *parsing*.

### 3.3 Proses *Parsing*

Pada proses ini, kalimat masukan yang diterima diubah menjadi huruf kecil dan semua tanda baca (titik, koma, kutip, garis miring, dan tanda seru) diubah menjadi spasi. Kemudian kalimat masukan dipecah menjadi kata per kata dan dimasukan kedalam satu variabel penampung kata untuk selanjutnya menentukan jenis kata dalam proses SPOK. Proses *parsing* akan dijelaskan sebagai berikut:

```

publicstring[] Parsing(string kalimat)
{
string pureKalimat = String.Join(" ", kalimat.Split(newstring[] { " " },
StringSplitOptions.RemoveEmptyEntries)).Trim();
string[] kata = pureKalimat.Split(new[]
{' ', '*', '&', '$', '@', '(', ')', '!', '#', '%', ',', '/', '?'},StringSplitOptions.RemoveEmptyEntries);
string[] penyimpananKata = newstring[kata.Length];
for (int i = 0; i < kata.Length; i++)
{
penyimpananKata[i] = kata[i];
}
return penyimpananKata;
}

```

Proses ini menjelaskan jika ada kalimat pertanyaan yang dimulai dengan *multiple spaced* antara kata, sebelum dan sesudah kalimat. Maka dilakukan proses penghilangan *multiple space* menjadi *normal space*. Setelah itu proses dilanjutkan dengan menghilangkan karakter khusus seperti: `\*`, `@`, `\$`, `#`, dll. Kemudian akan dilakukan proses pemecahan kalimat menjadi kata dan menyimpan kata ke dalam variabel penyimpanan kata.

```

publicvoid TampilkanParsing(string[] kata)
{
for (int i = 0; i < kata.Length; i++)
{
Console.WriteLine("kata ke-" + (i + 1) + " adalah: " + kata[i]);
}
Console.WriteLine(Environment.NewLine);
}

```

Proses ini menampilkan hasil *parsing* kalimat pertanyaan berupa kata. Kata yang ditampilkan dari proses ini sebagai berikut:

siapayangmengajarmatakuliah

logikafuzzy

### 3.4 Proses SPOK

Pada proses SPOK dilakukan pengecekan jenis kata yang merupakan hasil dari proses *parsing*. Berikut proses SPOK dari hasil proses *parsing* kalimat masukan pengguna.

```

classDatabaseKata
{
ListVariable lv = newListVariable();
privatestring[] dbKataKerjaIntrans;
privatestring[] dbKataKerjaTran;
privatestring[] dbKataTanya;
privatestring[] dbKataSubjek;
privatestring[] dbKataSifat;
privatestring[] dbKataKetWaktu;
privatestring[] dbKataKetTempat;
}

```

```
publicstring CekKata(string s)
{
string jenisKata = "";

if (KetersediaanKataKerjaIntransitif(s))
{
jenisKata = "kerja intransitif";
}
else
{
if (KetersediaanKataKerjaTransitif(s))
{
jenisKata = "kerja transitif";
}
else
{
if (KetersediaanKataTanya(s))
{
jenisKata = "tanya";
}
else
{
if (KetersediaanKataSubjek(s))
{
jenisKata = "subjek";
}
else
{
if (KetersediaanKataKeteranganWaktu(s))
{
jenisKata = "ket. waktu";
}
else
{
if (KetersediaanKataKeteranganTempat(s))
{
jenisKata = "ket. tempat";
}
}
}
}
}
}
}
}
```

Pada proses ini dilakukan pengecekan jenis kata dari hasil *parsing* seperti kata kerja transitif, kata kerja intransitif, kata tanya, kata subjek, kata keterangan waktu, kata keterangan tempat, dan disesuaikan dengan jenis kata yang terdapat dalam *database chatbot*. Sehingga didapatkan dari proses SPOK sebagai berikut :

```
Siapa = Kata tanya
yang = Kata sambung
mengajar = Kata kerja transitif
matakuliah = Kata benda
logika fuzzy = Kata benda
```

### 3.5 Proses Kata Kunci

Kata kunci digunakan untuk mengidentifikasi topik yang sedang dibicarakan oleh *user*. Proses kata kunci ditujukan untuk menemukan jawaban atas masukan yang diberikan pengguna.

Setelah kata kunci diketahui, sistem dapat langsung memberikan respon terhadap masukan dari *user*. Apabila kata kunci tidak ditemukan program akan memberikan respon kata kunci tidak ditemukan.

```
class DatabaseKata
{
    ListVariable lv = newListVariable();
    private string[] dbKataKEYWORD;
    public string CekKata(string s)
    {
        string jenisKata = "";
        if (KetersediaanKataKeyword(s))
        {
            jenisKata = "keyword";
        }
    }
}
```

Pada kalimat masukan "Siapa yang mengajar matakuliah logika fuzzy?" ditemukan kata kunci yaitu : "mengajar matakuliah nama\_mk". Kata kunci tersebut terdapat dalam *database chatbot*.

### 3.6 Proses Respon Jawaban

Proses *reasoning* dilakukan apabila kata kunci yang terdapat dalam *knowledge base chatbot* diketahui. Kemudian dilakukan proses pengembalian respon jawaban kepada pengguna.

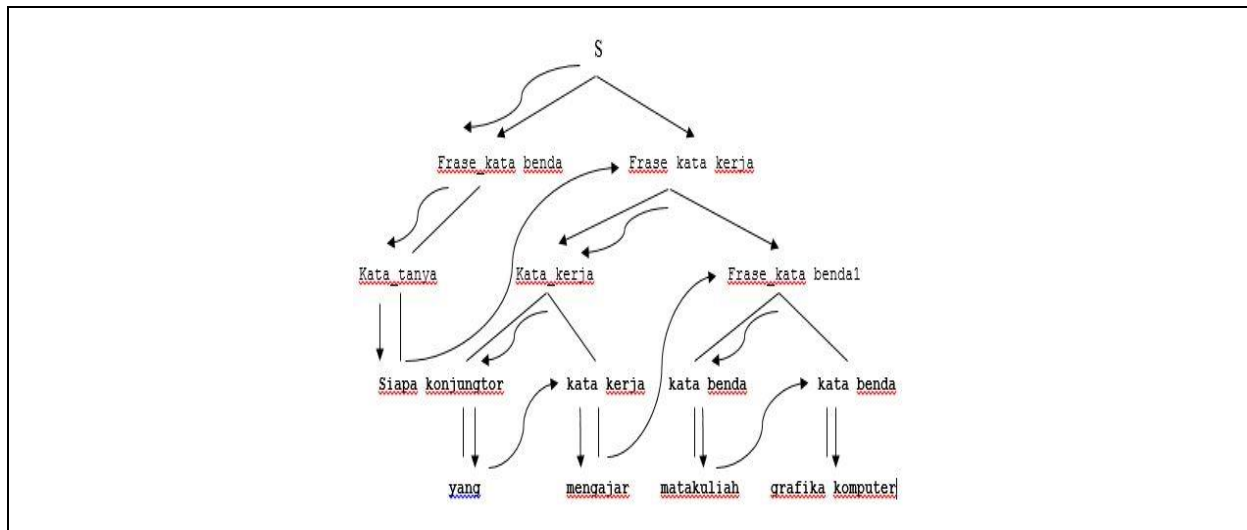
```
public string mkNamaDosen
{
    get
    {
        string tempMKdosen = this.mk;
        resultDosen =
        new WebClient().DownloadString("http://127.0.0.1/chatbot/dosen_info_perkuliahan.php?mk="
        tempMKdosen);
        return resultDosen;
    }
    set
    {
        resultDosen = value;
    }
}
```

Proses kata kunci "mengajar matakuliah nama\_mk" mengacu pada respon jawaban tentang nama dosen matakuliah. Karena kata kunci tersebut berhubungan dengan nama dosen matakuliah yang terdapat pada *database chatbot*. Sehingga respon jawaban yang dihasilkan adalah "dosen yang mengajar matakuliah logika fuzzy adalah nama\_dosen".

### 3.7 Kesimpulan Top Down Parsing

*Top down parsing* memulai pemeriksaan dari simbol awal kalimat dan mencoba untuk mencari bentuk simbol terminal berikutnya yang sesuai dengan jenis kata dari kalimat masukan. Proses *top down parsing* dapat dilihat pada Gambar 5.





**Gambar 5. Top down parsing**

Pada Gambar 5 ini dijelaskan proses *top down parsing* dengan menggunakan aturan tata bahasa Indonesia.

Kalimat awal:

“siapa yang mengajar matakuliah grafika komputer ?”

Tahapan selanjutnya adalah menentukan jenis kata dari kalimat dan menghasilkan kata. Adapun jenis kata yang dihasilkan adalah sebagai berikut :

siapa = kata tanya  
 yang = kata sambung  
 mengajar = kata kerja transitif  
 matakuliah = kata benda  
 grafika komputer = kata benda

Kemudian akan dilakukan proses *top down parsing* dengan aturan tata bahasa indonesia.

Kalimat → Frase\_kata benda, Frase\_kata kerja

S = Frase\_kata benda, Frase\_kata kerja  
 Frase\_kata benda = kata\_tanya  
 Kata tanya = Siapa  
 Frase\_kata kerja = kata kerja, Frase\_kata benda  
 Kata kerja = konjungtor, kata kerja  
 Konjungtor = yang  
 Kata kerja = mengajar  
 Frase\_kata benda = kata benda, pelengkap  
 Kata benda = matakuliah  
 Pelengkap = grafika komputer

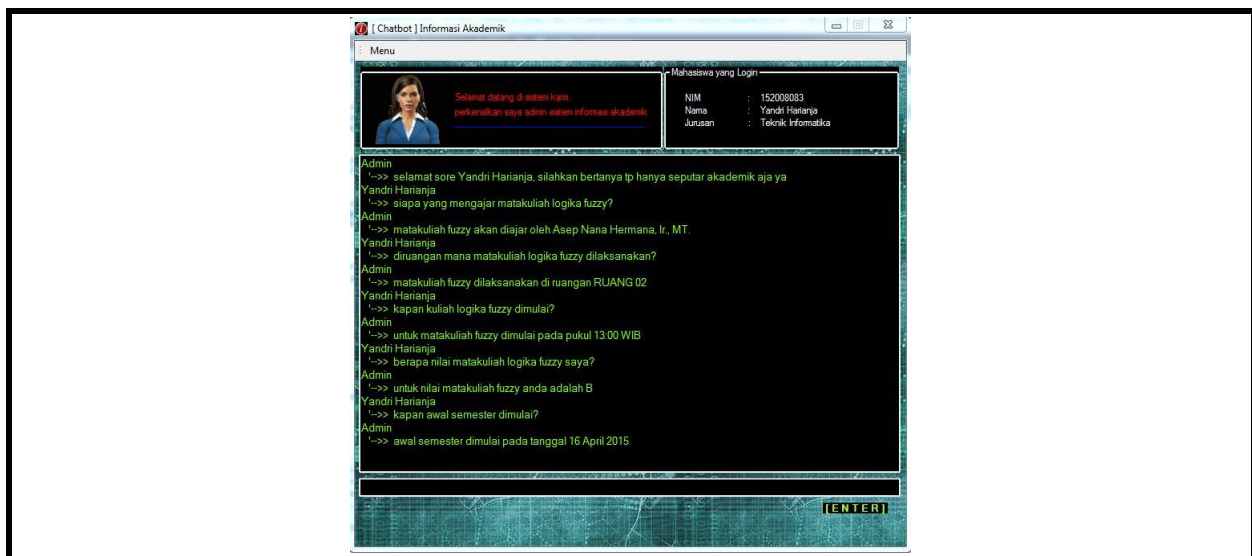
Dari proses *top down parsing* diperoleh struktur kalimat yang benar yaitu :

“Siapa yang megajar matakuliah grafika komputer”

Setelah struktur kalimat diketahui, sistem mencari kata kunci dari setiap masukan. Apabila sistem menemukan kata kunci tersebut maka kata kunci dicocokkan dengan kata kunci yang terdapat dalam *database*. Kemudian sistem merespon kalimat masukan pengguna berdasarkan kata kunci yang terdapat dalam *database* dan menampilkan jawaban yang sesuai dengan kata kunci tersebut.

### 3.7 Percakapan

Pengujian ini dilakukan untuk menampilkan halaman percakapan. Kemudian di halaman percakapan kita dapat melakukan tanya jawab seputar informasi akademik. Pengujian percakapan dapat dilihat pada Gambar 6.



Gambar 6. Pengujian percakapan

### 3.8 Pengujian beta

Pengujian ini menggunakan kuisisioner yang terdiri dari 5 pertanyaan yang ditanyakan kepada 10 responden. Berdasarkan hasil dari jawaban yang diberikan oleh responden yang telah menggunakan aplikasi *chatbot* dapat ditarik kesimpulan bahwa aplikasi *chatbot* sebesar 70% mudah dioperasikan, 60% aplikasi *chatbot* mempermudah mahasiswa dalam mencari informasi seputar jurusan informatika, 40% aplikasi *chatbot* menarik bagi mahasiswa, dan 70% aplikasi chatbot bermanfaat bagi mahasiswa untuk mendapatkan informasi yang diinginkan.

## 4. KESIMPULAN

Berdasarkan hasil pengujian black box pada proses tanya jawab sistem dapat menjawab masukan dari pengguna dengan baik sesuai dengan kata kunci yang ada di *knowledge base*. Dan hasil pengujian *user acceptance test* tanya jawab yang diajukan langsung kepada mahasiswa, 70% aplikasi chatbot mudah dioperasikan, 60% aplikasi chatbot mempermudah mahasiswa dalam mencari informasi seputar jurusan informatika, 40% aplikasi chatbot menarik bagi mahasiswa, dan 70% aplikasi chatbot bermanfaat bagi mahasiswa untuk mendapatkan informasi seputar jurusan teknik informatika.

### DAFTAR RUJUKAN

- [1] Kristanto,Andri (2005). Kecerdasan Buatan. Graha Ilmu, Yogyakarta.
- [2] Hasugian, Jonner (2006). Penggunaan Bahasa Alamiah dan Kosa kata Terkendali dalam
- [3] Sistem Temu Balik Informasi Berbasis Teks. Universitas Sumatera Utara, Sumatera Utara.
- [4] Suciadi, James (2001). Studi analisis Metode-metode *Parsing* dan Interpretasi Semantik pada *Natural Language Processing*. Universitas Kristen Petra. Surabaya.
- [5] Fatima, Choirunnisa (2013). Pencocokan String pada Aplikasi Chatbot Sederhana. Institut Teknologi Bandung. Bandung.