

BAB 2

LANDASAN TEORI

2.1 NLP (*Natural Language Processing*)

NLP merupakan bagian dari ilmu AI yang memiliki fokus pada pemrosesan bahasa alami, dimana bahasa alami adalah bahasa yang umum digunakan oleh manusia untuk saling berkomunikasi satu sama lain, namun bahasa yang diterima oleh komputer harus diproses terlebih dahulu dan dipahami agar maksud dari pengguna dapat dipahami[6].

Prinsipnya bahasa natural merupakan suatu pesan yang ingin dikomunikasikan antar manusia. Bentuk yang sering digunakan dalam bahasa alami direpresentasikan oleh suara atau ucapan, namun sering juga melalui tulisan. Inti dari pemrosesan bahasa alami ini adalah penguraian kalimat atau disebut parser, parser berfungsi untuk membaca setiap kata dan menentukan jenis kata apa saja yang boleh mengikuti kata tersebut[6].

Dalam penelitian ini bahasa alami digunakan sebagai data masukan yang akan diproses menjadi perintah SQL yang mengandung *sub-query*.

2.2 Case Folding

Case folding adalah proses penyeragaman semua huruf ke dalam *case* yang sama menjadi huruf kecil (*lowercase*) atau huruf kapital (*uppercase*)[7]. *Case folding* merupakan proses menyeragamkan setiap huruf pada kalimat masukan dengan mengubah semua huruf menjadi kapital (*uppercase*) atau kecil (*lowercase*). Pada penelitian ini proses *case folding* yang digunakan yaitu menyeragamkan semua huruf menjadi kecil (*lowercase*).

Dalam penelitian ini proses penyeragaman kata diperlukan untuk mempermudah proses pendeteksian setiap kata dari data masukan. Contoh dari proses *case folding* dapat dilihat pada Tabel 2.1 berikut.

Tabel 2.1 Contoh Case Folding

Kalimat awal	Kalimat setelah melalui proses <i>case folding</i>
Tampilkan data transaksi!	tampilkan data transaksi!

2.3 Filtering

Filtering merupakan proses untuk memilah setiap karakter pada kalimat masukan hanya karakter tertentu. Tujuan dari proses *filtering* yaitu untuk mengurangi karakter yang tidak diperlukan oleh proses selanjutnya, sehingga mengurangi *noise* saat proses pengambilan informasi. Pada penelitian ini, karakter yang menjadi masukan hanya ‘a’ sampai ‘z’, ‘0’ sampai ‘9’.

Dalam penelitian ini proses *filtering* dilakukan agar informasi kalimat yang diproses tidak terganggu oleh karakter-karakter yang dianggap tidak penting. Contoh dari proses *filtering* dapat dilihat pada Tabel 2.2 berikut.

Tabel 2.2 Contoh Filtering

Kalimat awal	Kalimat setelah melalui proses <i>case folding</i>
tampilkan data transaksi!	tampilkan data transaksi

2.4 Tokenizing

Tokenizing adalah proses untuk memisahkan kalimat menjadi kata – kata (token) dengan parameter pemisah berupa spasi (‘ ’). Proses ini dilakukan karena pada proses selanjutnya akan dilakukan pemrosesan setiap kata. Hasil dari *tokenizing* kata yaitu berupa token – token kata.

Dalam penelitian ini proses *tokenizing* diterapkan agar sistem dapat mendeteksi setiap kata yang telah dipisahkan menjadi kumpulan token. Contoh dari proses *tokenizing* dapat dilihat pada Tabel 2.3 berikut.

Tabel 2.3 Contoh Proses Tokenizing

Kalimat awal	Hasil proses <i>tokenizing</i>
tampilkan data transaksi	tampilkan
	data
	transaksi

2.5 *Stemming*

Stemming mengacu pada proses pemotongan kata menjadi kata dasar. Kata dasar adalah bagian dari kata yang tersisa setelah dilakukan proses penghapusan awalan dan akhiran[8]. Dalam bahasa Indonesia awalan dan akhiran disebut dengan imbuhan, contohnya adalah imbuhan “-nya’ seperti dalam kata “mitranya” yang memiliki kata dasar “mitra”.

Proses *stemming* dilakukan karena kata yang akan dideteksi merupakan kata dasar sehingga perlu dilakukan proses penghapusan imbuhan adapun contoh dari proses *stemming* dapat dilihat pada Tabel 2.4 berikut.

Tabel 2.4 Contoh Proses *Stemming*

Kata awal	Kata setelah melalui proses <i>stemming</i>
Mitranya	Mitra
Transaksinya	Transaksi

2.6 *Stopword Removal*

Stopword removal adalah kata yang sering muncul dan dianggap tidak penting, dalam suatu bahasa *stopword* membantu untuk membangun struktur kalimat tetapi tidak mewakili konten apapun dari isi kalimat tersebut[8], beberapa contohnya adalah kata ‘yang’, ‘dan’, ‘dengan’, ‘atau’ dan lain – lain.

Proses *stopword removal* dilakukan untuk menghapuskan kalimat-kalimat yang dianggap tidak penting yang dapat menimbulkan hasil dari proses translasi menjadi kurang baik atau keliru. Contoh dari *stopword removal* dapat dilihat pada Tabel 2.5 berikut.

Tabel 2.5 Contoh *Stopword Removal*

Kalimat awal	Kalimat setelah melalui proses <i>stopword removal</i>
Tampilkan data mitra yang memiliki saldo lebih dari 100000	Tampilkan data mitra yang memiliki saldo lebih dari 100000

2.7 Basis Data

Basis Data atau lebih sering disebut *Database* adalah sekumpulan tabel-tabel yang saling berelasi, relasi tersebut bisa ditunjukkan dengan kunci dari tiap table yang ada. Satu database menunjukkan satu kumpulan data yang dipakai dalam satu lingkup perusahaan atau instansi[1].

Database atau basis data sangat penting dalam sistem informasi karena basis data merupakan gudang penyimpanan data yang dapat diolah lebih lanjut. Peran basis data menjadi sangat penting karena dapat mengorganisir data dan juga menghindari duplikasi data atau lebih sering disebut dengan redundansi. Dalam basis data ada proses *database management system* atau disingkat DBMS yang merupakan suatu sistem yang memungkinkan *admin database* melakukan pengaksesan data, pengendalian dan juga pemeliharaan data secara efisien. Dalam penelitian ini basis data digunakan untuk menyimpan data pendukung sistem dan juga digunakan sebagai media pengetesan dari perintah hasil penerjemahan bahasa Indonesia ke bahasa SQL.

2.8 SQL (*Structured Query Language*)

SQL adalah sebuah bahasa permintaan *database* terstruktur. Bahasa SQL dibuat sebagai bahasa yang dapat merelasikan beberapa tabel dalam database maupun merelasikan antar database[2]. SQL merupakan bahasa standar yang paling banyak digunakan dalam hal manajemen basis data. SQL terbagi menjadi dua fungsi utama yaitu DDL dan DML. DDL merupakan singkatan dari *Data Manipulation Language* yang berfungsi untuk membuat atau merubah struktur dari suatu database yang memiliki fungsi seperti *create*, *alter*, *delete*, sedangkan DML merupakan bahasa yang dapat digunakan untuk manipulasi data dalam basis data yang meliputi *insert*, *update* dan *delete*, selain tiga fungsi data tersebut, ada satu fungsi lain yaitu fungsi untuk menampilkan data (*select*) yang seharusnya termasuk kedalam DQL (*Data Query Language*) namun pada prakteknya banyak yang menyebutkan bahwa fungsi *select* tersebut termasuk bagian dari DML. Dalam penelitian ini SQL merupakan bahasa permintaan *database* yang digunakan pada hasil proses translasi yang dilakukan.

2.9 Sub-Query

Jika diartikan *sub-query* sendiri adalah query dalam query. Jadi untuk mendapatkan informasi pada sebuah tabel ataupun dalam relasional tabel yang sangat kompleks, kita dapat menggunakan kondisi dengan sub permintaan tertentu[2]. *Sub-query* terbagi menjadi beberapa tipe yaitu sebagai berikut.

1. *Single row query*, merupakan *sub-query* yang hanya menghasilkan satu baris saja seperti berikut.

```
SELECT * FROM mahasiswa
WHERE tinggi_badan = (SELECT MAX(tinggi_badan) FROM
mahasiswa
```

Gambar 2.1 Sub-query select table

Dalam contoh gambar 2.1 di atas merupakan suatu contoh *query* untuk menampilkan data mahasiswa dengan tinggi badan tertinggi.

```
SELECT * FROM mitra
WHERE kode_mitra = (SELECT kode_mitra FROM transaksi WHERE
kode_produk = 'TSEL5')
```

Gambar 2.2 Sub-query select 2 tabel

Dalam contoh gambar 2.2 di atas merupakan *query* untuk menampilkan data praktek dokter yang memiliki spesialis anak.

2. *Multiple row subquery*, merupakan *sub-query* yang menghasilkan satu atau banyak baris data, dalam mysql selain bentuk standar subquery, terdapat beberapa operator khusus untuk membantu melengkapi fungsi subquery tersebut. Beberapa operator tersebut adalah IN, ANY, SOME, EXIST, dan ALL[2]. Contohnya adalah sebagai berikut.

```
SELECT * FROM transaksi
WHERE id_pelanggan IN( SELECT id_pelanggan FROM pelanggan
WHERE kota = 'Bandung');
```

Gambar 2.3 Sub-query multiple row select in

Dalam contoh gambar 2.3 di atas merupakan *query* untuk menampilkan data dari tabel transaksi dimana *id_pelanggan* termasuk dalam *sub-query* yang menampilkan *id_pelanggan* dari tabel pelanggan dimana kotanya merupakan Bandung.

```
SELECT * FROM transaksi
WHERE id_pelanggan NOT IN(SELECT id_pelanggan FROM
pelanggan WHERE kota = 'Bandung);
```

Gambar 2.4 Sub-query multiple row select not in

Dalam contoh gambar 2.4 di atas merupakan *query* untuk menampilkan data dari tabel transaksi dimana *id_pelanggan* tidak termasuk dalam *sub-query* yang menampilkan *id_pelanggan* dari tabel pelanggan dimana kotanya adalah Bandung.

3. *nested subqueries*, merupakan *sub-query* yang berada di dalam *sub-query* lainnya seperti berikut.

```
SELECT * FROM orders
WHERE ord_amount>2000 AND ord_date<'01-SEP-08' AND
ADVANCE_AMOUNT < ANY(SELECT OUTSTANDING_AMT
FROM CUSTOMER WHERE GRADE=3 AND
CUST_COUNTRY<>'India' AND opening_amt<7000 AND EXISTS
(SELECT * FROM agents WHERE commission<.12));
```

Gambar 2.5 Nested sub-query

Dalam contoh Gambar 2.5 di atas merupakan *query* untuk menampilkan data dari tabel orders dimana *ord_amount* lebih besar dari 2000 dan

ord_date lebih kecil dari 1 september 2008 dan advance_amount lebih kecil dari semua hasil *sub-query* yang menampilkan select_outstanding_amt dari tabel customer dimana grade sama dengan tiga dan cust_country bukan India dan opening_amt lebih kecil dari 7000 dan ada pada *sub-query* yang menampilkan data tabel agents dimana comission lebih kecil dari 12.

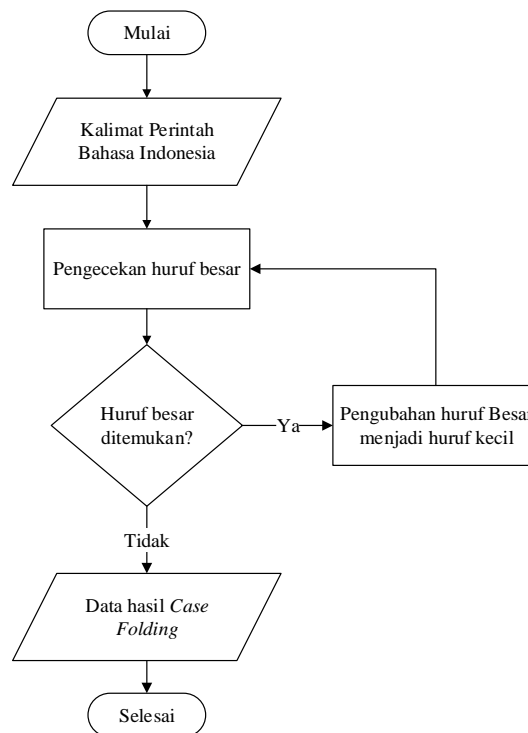
2.10 Pemodelan

Skema pemodelan data adalah metode yang memungkinkan untuk memodelkan pada aplikasi rancang bangun penerjemah text bahasa Indonesia ke *Data Manipulation Language* (DML) yang mengandung *sub-query*.

2.10.1 Flowchart

Flowchart adalah bagan-bagan yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah. *Flowchart* merupakan cara penyajian suatu algoritma. *Flowchart* adalah penggambaran secara grafik dari langkah-langkah dan urutan-urutan prosedur dari suatu program. *Flowchart* biasanya mempermudah penyelesaian suatu masalah khususnya masalah yang perlu dipelajari dan dievaluasi lebih lanjut[9].

Tujuan membuat *Flowchart* adalah menggambarkan suatu tahapan penyelesaian masalah secara sederhana, terurai, dan jelas menggunakan simbol-simbol standar. Contoh bagan flow chart dapat dilihat dalam Gambar 2.6 Contoh *Flowchart* berikut.



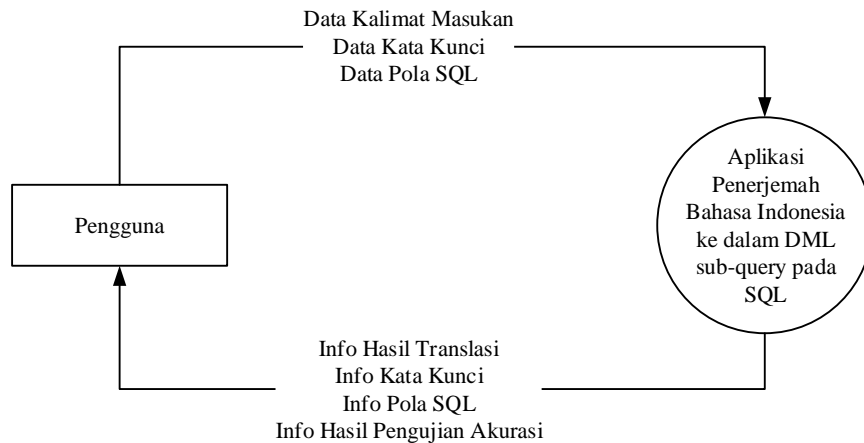
Gambar 2.6 Contoh *Flowchart*

2.10.2 Diagram Konteks

Diagram Konteks adalah sebuah diagram sederhana yang menggambarkan hubungan antara *entity* luar, masukan dan keluaran dari sistem. Diagram konteks direpresentasikan dengan lingkaran tunggal yang mewakili keseluruhan sistem [6]. Diagram konteks dimulai dengan penggambaran terminator, aliran data, aliran kontrol penyimpanan, dan proses tunggal yang menunjukkan keseluruhan sistem. Ada beberapa aturan dalam menggambarkan diagram konteks, berikut di bawah ini menunjukkan aturan tersebut.

1. Menggunakan hanya satu simbol proses.
2. Memberi label simbol proses tersebut untuk menggambar seluruh sistem, biasanya berupa kata kerja ditambah objek.
3. Tidak memberi nomor pada simbol proses.
4. Menyertakan semua terminator dari sistem.
5. Menunjukkan semua arus data antara terminator dan sistem.

Pada penelitian ini diagram konteks digunakan untuk menggambarkan sistem berbasis komputer yang dirancang secara umum. Contoh dari diagram konteks dapat dilihat pada Gambar 2.7 berikut.



Gambar 2.7 Contoh Diagram Konteks

2.10.3 DFD (*Data Flow Diagram*)

DFD adalah suatu model logika data atau proses yang dibuat untuk menggambarkan darimana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut [9].

1. Arus Data (*Data Flow*)

Menunjukkan arus dari data yang dapat berupa masukan untuk sistem atau dari proses sistem.

2. Proses

Proses adalah kegiatan yang dilakukan oleh orang, komputer dari hasil arus data yang masuk ke dalam proses untuk dihasilkan arus data yang akan keluar dari proses.

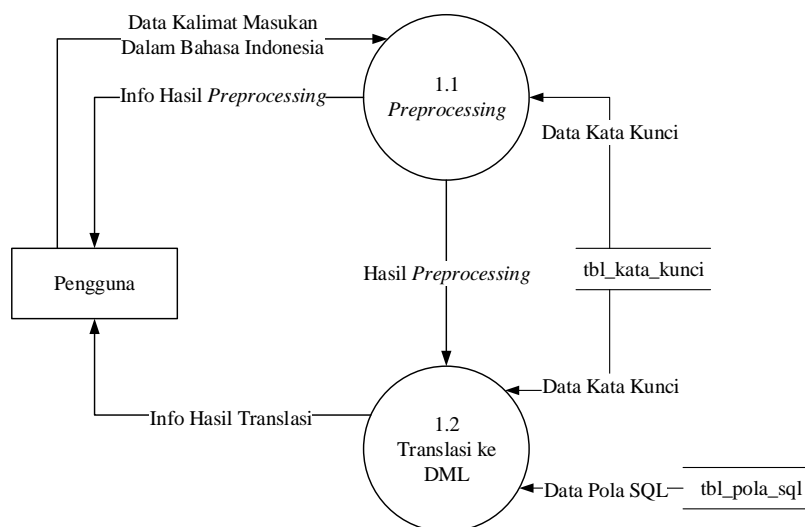
3. Kesatuan Luar (*External Entity*)

Kesatuan luar merupakan kesatuan di lingkungan luar sistem yang dapat berupa orang, organisasi atau sistem lain yang akan memberikan masukan atau menerima keluaran dari sistem.

4. File

Kumpulan data yang disimpan dengan cara tertentu. Data yang mengalir disimpan dalam file. Aliran data di-update atau ditambahkan ke dalam file.

Pada penelitian ini *Data Flow Diagram* (DFD) digunakan untuk untuk penggambaran analisis maupun rancangan sistem yang mudah dikomunikasikan kepada pemakai maupun pembuat program. Contoh dari DFD dapat dilihat pada Gambar 2.8 berikut.



Gambar 2.8 Contoh DFD

2.10.4 Kamus Data

Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum [9]. Kamus data mendefinisikan elemen data dengan menjelaskan arti aliran data dan penyimpanan data dalam DFD, mendeskripsikan komposisi paket data yang bergerak melalui aliran, mendeskripsikan komposisi penyimpanan data, menspesifikasikan nilai dan satuan yang relevan bagi penyimpanan dan aliran, mendeskripsikan hubungan detail antar penyimpanan.

Kamus data berfungsi untuk membantu pelaku sistem untuk mengartikan alokasi secara detail dan mengorganisasikan semua elemen data yang digunakan dalam sistem secara persis sehingga baik pemakai atau penganalisis sistem

mempunyai dasar pengertian yang sama tentang masukan, keluaran, penyimpanan dan proses.

Kamus data biasanya berisi:

- a) Nama-nama dari data
- b) Digunakan pada – merupakan proses-proses yang terkait data
- c) Deskripsi – merupakan deskripsi data
- d) Informasi tambahan – seperti tipe data, nilai data, batas nilai data, dan komponen yang membentuk data.

Kamus data memiliki simbol untuk menjelaskan informasi tambahan seperti pada Tabel 2.6 berikut.

Tabel 2.6 Kamus Data

Simbol	Keterangan
=	disusun atau terdiri dari
+	Dan
[]	baik ... atau....
{ } ⁿ	n kali diulang/ bernilai banyak
()	data opsional
...	batas komentar

Pada penelitian ini kamus data digunakan untuk memberikan informasi mengenai aliran data pada DFD.

2.10.5 Pengujian

Untuk menghitung akurasi dari hasil query dilakukan perhitungan dengan rumus berikut [10].

$$Akurasi = \frac{Jumlah\ hasil\ query\ sistem\ benar}{Jumlah\ sampel\ data\ uji} \times 100\% \quad (2.3)$$

Proses pengujian pada penelitian ini bertujuan untuk menghitung akurasi yang digunakan untuk mengetahui tingkat keberhasilan dari proses translasi yang

dilakukan oleh sistem dengan membandingkan *query* harapan dan *query* hasil translasi.

2.11 Bahasa Pemrograman

Pemrograman untuk komputer seringkali disingkat dengan istilah *coding*, yaitu sebuah proses menulis dan menguji sebuah kode sumber. Kode sumber ini ditulis dengan suatu bahasa pemrograman. Bahasa pemrograman merupakan kumpulan aturan yang disusun sedemikian rupa sehingga memungkinkan pengguna komputer membuat program yang dapat dijalankan dengan aturan tersebut[11].

Bahasa pemrograman secara tradisional dilihat dari tiga segi aspek[12] :

1. *Syntax* sebagai struktur dari bahasa pemrograman.
2. *Semantic* sebagai makna dari bahasa pemrograman.
3. *Pragmatic* sebagai implementasi dari bahasa pemrograman

Dalam penelitian ini bahasa pemrograman digunakan untuk mengimplementasikan perancangan sistem yang sudah dilakukan menjadi sebuah program sehingga dapat dilakukan pengujian terhadap perancangan yang telah dilakukan.

2.11.1 PHP

PHP(PHP:Hypertext Preprocessor) adalah sebuah bahasa pemrograman yang berbentuk *scripting*. Sistem kerja dari program ini adalah sebagai *interpreter* bukan sebagai *compiler*. Perbedaan antara bahasa Compiler dengan bahasa yang berupa Interpreter[13].

Untuk menjalankan php diperlukan sebuah perangkat lunak berupa web server dan web browser. Web server berperan sebagai penerjemah dan juga pemroses sintaks PHP yang dituliskan oleh *coder* dan mengirimkan hasil penerjemahan dan pemrosesan yang telah dilakukan pada web browser.

Pada penelitian ini PHP digunakan karena dapat membuat program yang dibuat menjadi dinamis. Dibandingkan dengan HTML yang hanya dapat menampilkan saja, PHP dapat memproses suatu perintah yang kita sisipkan dalam sintak PHP. Dalam PHP kita dapat menyisipkan kode untuk memproses seperti perhitungan, perulangan, dan juga logika seperti IF-Else. Hal lain yang dapat

dilakukan PHP adalah dapat mengintegrasikan antara website dengan *database* untuk menyimpan maupun mengakses data. Sistem yang dibangun pada penelitian ini menggunakan PHP versi 7.2.0.

2.11.2 Javascript

Javascript merupakan salah satu bahasa pemrograman yang pertama kali dikembangkan oleh Netscape yang digunakan dalam sebuah browser atau perambah web [14]. JavaScript dibuat agar mudah diintegrasikan ke dalam program dan aplikasi berbasis web. Javascript pada aplikasi ini digunakan untuk menghitung nilai performansi akurasi.

2.12 Perangkat Lunak Pendukung

Perangkat lunak adalah istilah khusus untuk data yang diformat, dan disimpan secara digital, termasuk program komputer, dokumentasinya, dan berbagai informasi yang bisa dibaca, dan ditulis oleh komputer. Dengan kata lain bagian sistem komputer yang tidak berwujud.

Dalam pembangunan dan pengujian aplikasi penerjemah dari bahasa alami bahasa Indonesia ke dalam *query SQL*, peneliti menggunakan Xampp sebagai *web server* untuk pembangunan dan pengujian aplikasi penerjemah dari bahasa alami bahasa Indonesia ke dalam *query SQL* yang dibangun, *web browser*, dengan PhpStorm 10 sebagai penyunting kode bahasa pemrograman.

2.12.1 MySQL

MySQL merupakan *database server open source* yang cukup populer keberadaannya. Dengan berbagai keunggulan yang dimiliki, membuat software database ini banyak digunakan oleh para praktisi untuk membangun suatu project. Adanya fasilitas API (*Application Programming Interface*) yang dimiliki oleh MySQL, memungkinkan bermacam-macam aplikasi komputer yang ditulis dengan berbagai bahasa pemrograman dapat mengakses basis data MySQL.

MySQL pertama kali dirintis oleh seorang programmer *database* bernama Michael Widenius. MySQL adalah program database yang mampu mengirim dan menerima data dengan sangat cepat dan *multi user*. MySQL memiliki dua bentuk lisensi, yaitu *free software* dan *shareware*.

Keunggulan MySQL adalah kemampuannya dalam menyediakan berbagai fasilitas atau fitur-fitur yang dapat digunakan oleh bermacam-macam user. *User* ini sendiri termasuk administrator *database*, programmer aplikasi, manager, sampai *end user* (pemakai akhir)[15].

Dalam penelitian ini MySQL digunakan untuk menyimpan data-data pendukung sistem seperti data kata kunci dan data pola.

2.12.2 XAMPP

XAMPP adalah sebuah aplikasi yang dapat menjadikan komputer kita menjadi sebuah server. Kegunaan XAMPP ini untuk membuat jaringan local sendiri dalam artian kita dapat membuat website secara offline, Jadi fungsi dari XAMPP itu sendiri merupakan server website kita untuk cara memakainya. Mengapa harus menjadi server, karena dalam hal ini komputer yang akan kita pakai harus memberikan pelayanan untuk pengaksesan web, untuk itu komputer kita harus menjadi server. XAMPP yang digunakan pada penelitian ini adalah versi 3.1.0 64 bit.

2.12.3 Apache Server

Apache HTTP Server adalah server web yang dapat dijalankan di banyak sistem operasi (Unix, BSD, Linux, Microsoft Windows dan Novell Netware serta platform lainnya) yang berguna untuk melayani dan memfungsikan situs web. Protokol yang digunakan untuk melayani fasilitas web/www ini menggunakan HTTP Apache server. Dalam penelitian ini apache digunakan sebagai server untuk menjalankan sistem yang dibuat dengan menggunakan bahasa pemrograman PHP.