# BAB II

# LANDASAN TEORI DAN TINJAUAN STUDI

## 2.1 Tinjauan Studi

Penelitian yang dilakukan pada PT. RTV dengan judul “ Rancangan System Aplikasi Ticketing IT Helpdesk Berbasis Android “ bertujuan agar mempermudah proses penyelesaian masalah hardware dan software pada perusahaan.

Penyelesaian masalah hardware dan software yang terjadi pada karyawan Peruahaan Rajawali Televisi merupakan salah satu kendala yang dihadapi karyawan dalam proses pekerjaan yang dilakukan setiap hari. Penyelesaian masalah hardware dan software tidak bisa dianggap semuanya mudah, karena ada beberapa aspek yang memerlukan pertimbangan-pertimbangan.

Penyelesaian masalah hardware dan software memiliki dampak kecil ataupun besar pada proses pekerjaan karyawan perusahaan dimana ada beberapa pekerjaan karyawan yang saling berhubungan, sehingga jika ada perkerjaan yang tertunda dibagian tertentu maka bisa menghambat juga perkerjaan kelanjutannya.

Penyelesaian masalah hardware dan system memiliki beberapa aspek level yang harus ditangani lebih dahulu, dimana level dilihat dari beberapa aspek yaitu :

1. Jabatan Karyawan,
2. Perkiraan Waktu penyelesaian.

Dengan aspek-aspek diatas maka dapat dibuat bahan pertimbangan dalam sistem pembuatan keputusan, manakah diantara permasalahan pada hardware atau software yang harus didahulukan pengerjaannya.

## 2.2 Landasan Teori

### 2.2.1 Pengertian Aplikasi

Pengertian aplikasi adalah sekumpulan elemen yang saling berkaitan dan saling mempengaruhi dalam melakukan kegiatan bersama atau berinteraksi untuk mencapai tujuan tertentu. Sebagai contoh Aplikasi komputer yang terdiri dari *software*, *Hardware*, dan *Brain-ware* ketiga unsur tersebut saling berkaitan dan bertanggung jawab memproses suatu masukan (*input*) dan menghasilkan keluaran (*output*).

### 2.2.2 Pengertian *Mean Time To Repair ( MTTR )*

Pengertian *Mean Time To Repair ( MTTR )*  adalah waktu yang dibutuhkan untuk memulihkan suatu sistem dari sebuah kegagalan. Dalam hal ini juga termasuk waktu yang dibutuhkan dalam mendiagnosa masalah, waktu yang diperlukan untuk memperbaiki sistem. (Torrel & Avelar, 2010).

MTTR ini diperoleh dengan membagi total waktu yang dibutuhkan pada setiap perbaikan yang dilakukan dengan banyaknya kerusakan yang terjadi. Variable yang dibutuhkan adalah waktu-waktu lama perbaikan dan jumlah kerusakan yang terjadi. Untuk mengetahui waktu lama perbaikan, sistem akan mencatat waktu lama perbaikan dari mencatat waktu start perbaikan dan konfirmasi Staff IT yang melakukan penyelesaian perbaikan, kemudian akan dilakukan perbandingan sehingga didapat waktu lama penyelesaian.

### 2.2.3 Pengertian Web Server

Web Server adalah tempat menyimpan aplikasi web yang dibuat kemudian dalam mengaksesnya melalui web browser, juga digunakan untuk menguji apakah web aplikasi berjalan dengan baik atau tidak. ( Priyanto Hidayatullah, Jauhari Khairul Kawistara, 2014, 125 ).

Ada beberapa web server yang bisa menjadi pilihan. Pada windows ada *Internet Informasi Services ( IIS )* dikeluarkan oleh Microsoft. Aplikasi web server ini sudah support asp.net dan php,

Web server lainnya yang sangat terkenal dan sering digunakan adalah Apache, Apache merupakan project salah satu yayasan yang memiliki banyak project software gratis untuk digunakan oleh banyak orang. Yayasan itu bernama the *Apache Software Foundation.*

Aplikasi web server Apache bisa digunakan bukan hanya di windows, tetapi bisa juga digunakan pada linux dan Mac OS. Webserver Apache terus dikembangkan hingga senantiasa kompatibel dengan teknologi web terkini yang sangat tinggi. Maka agar sebuah web dinamis bisa diakses secara local menggunakan web server local, maka harus di install PHP ( untuk Script ), Apache ( Web Server ), dan MY SQL ( Database ) satu demi satu. Namun sekarang ada beberapa komunitas programer yang menyediakan solusi praktis dalam menginstall aplikasi-aplikasi tersebut, diantaranya adalah :

1. XAMPP
2. WAMPSERVER
3. WAMP ( Mancintosh, Apache, MYSQL, dan PHP )

Namun saat ini yang akan digunakan adalah XAMPP, karena XAMPP support untuk banyak siste operasi seperti Windows, Linux, Mac, dan Solaris sehingga tidak masalah ketika berpindah-pindah sistem operasi.

Kata XAMPP berasal dari :

1. ‘X’ yang berarti *cross platform*  karena XAMPP bisa dijalankan pada Windows, Linux, Mac, dsb.
2. ‘A’ yang berarti Apache sebagai web servernya.
3. ‘M’ yang berarti MYSQL sebagai Database Management System ( DBMS).
4. ‘PP’ yang berarti PHP dan Perl sebagai bahasa yang mendukungnya.

### 2.2.4 Pengertian Bootstrap Twitter

Bootstrap adalah sebuah frameword yang terdiri dari HTML, CSS, dan Javascript yang berfungsi untuk mendesain sebuah website secara mudah dan cepat dimana hasilnya sangat responsive diberbagai layar monitor baik layar komputer ataupun layar smartphone atau tablet

Bootstrap dirilis pertama kali oleh twitter pada bulan februari 2012 dengan project yang bernama project *Github*. Bootstrap memiliki dukungan terhadap HTML 5 dan CSS 3 dan kompatibel dengan berbagau web browser sejak Bootstrap versi 2.0 dirilis.(Yosef Murya & Rizki Hesananda, 2014 : 33).

Beberapa manfaat dari framework Bootstrap Twitter bagi programer dapat dilihat, antara lain sebagai berikut :

1. Efisien Waktu :
2. Para programer tidak perlu lagi membuang waktu untuk menuliskan script program, hanya dengan menggunakan potongan script yang tepat dan sesuai dengan kebutuhan programer dan mencocokan dengan struktur website yang sedang dibangun.
3. Fleksibel :
4. Dapat digunakan sesuai keinginan, karena Bottstrap Twitter bisa memodifikasi script yang ada didalamnya. Tidak ada ketentuan dalam memanfaatkan Bootstarp Twitter.
5. Desain :
6. Dapat mendesain tampilan untuk komputer, smartphone, dan table hanya menggunakan sedikit script.
7. Web yang dibuat akan dapat direspon atau ditampilkan dengan baik diberbagai web browser seperti Google Chrome, Firefox Mozila, Opera, dan Lain-lain. (Yosef Murya & Rizki Hesananda, 2014 : 33 ).

### 2.2.5Pengertian Framework Codeigniter

* + - 1. Framework adalahsuatu struktur konsetual dasar yang digunakan utuk memecahkan atau menangani suatu masalah kompleks, dimana istilah ini sering digunakan dalam bidang perangkat lunak. Framework juga dapat diartikan sekumpulan perintah/program dasar dimana perintah dasar tersebut dapat digunakan lagi untuk menyelesaikan masalah yang lebih kompleks sehingga dapat digunakan untuk membantu membuat aplikasi baru atau aplikasi kompleks tanpa barus membuat program dari awal.
      2. Codeigniter adalah salahsatu Framework yang cukup baik untuk saat ini karena didalamnya terdapat fitur lengkap web dimana fitur-fitur tersebut sudah dikemas menjadi satu. Codeigniter juga banyak digunakan khususnya bagi development web untuk mengembangkan aplikasi berbasis webnya tersebut.( Priyanto Hidayatullah, Jauhari Khairul K., 2014: 295)
      3. Beberapa Keuntungan yang didapat dalam penggunaan framework adalah :

1. **Menghemat waktu pengembangan :** dengan library yang telah disediakan oleh framework maka tidak perlu lagi memikirkan hal-hal dasar atau hal-hal umum yang sudah dibuat pada library, jadi bisa hanya fokus ke proses bisnis yang akan dikerjakan.
2. **Penggunaan ulang program/kode :**  dengan menggunakan framework maka pekerjaan akan memiliki struktur yang baku, sehingga dapat menggunakan program/kode itu kembali untuk pekerjaan lain.
3. **Bantuan komunitas :**pada umumnya setiap framework yang besar mempunyai komunitas dimana komunitas inilah yang siap membantu jika ada permasalahan, selain itu bisa berbagi ilmu sehingga dapat meningkatkan kemampuan pemrograman.
4. **Kumpulan prigram terbaik :** sebuah framework merupakan kumpulan program terbaik yang sudah teruji, sehingga dapat meningkatkan kualitas program. ( Priyanto Hidayatullah, Jauhari Khairul K., 2014, 295)

### 2.2.6 Pengertian Android

Android adalah sistem operasi open source yang dirancang untuk perangkat selular layar sentuh smartphone, tablet. Dan Google merilis kodenya dibawah lisensi Apache. Kode open source dan lisensi perizinan pada Android memungkinkan perangkat lunak untuk memodifikasi secara bebas dan didistribusikan oleh para pembuat perangkat, operator nirkabel, dan pengembang aplikasi. Android juga memiliki sejumlah besar komunitas pengembang aplikasi (apps) yang memperluas fungsional perangkat, umumnya ditulis dalam versi kustomisasi bahasa pemrograman Java. (Gregorius Agung .2015:1)

#### 2.2.6.1 Sejarah dan Perkembangan Android

Pada mulanya terdapat berbagai macam sistem operasi pada perangkat seluler diantaranya sistem Symbian, Microsoft Windows Mobile, Mobile Linux, iPhone, dan sistem operasi lainnya. Namun diantara sistem operasi yang ada belum mendukung standar dan penerbitan API yang dapat dimanfaatkan secara keseluruhan dan dengan biaya yang murah. Kemudian Google ikut berkecimpung didalamnya dengan *platform* android, yang menjanjikan keterbukaan, keterjangkauan, *opensource*, dan *framework* berkualitas.

Pada tahun 2005 Google mengakuisisi perusahaan Android Inc, untuk memulai pengembangan *platform* Android. Dimana terlibat dalam pengembangan ini Andy Rubin, Rich Miner, Nick Sears, dan Chris White. Pada pertengahan 2007 sekelompok pemimpin industri bersama-sama membentuk aliansi perangkat seluler terbuka, *Open Handset Alliance* (OHA). Bagian dari tujuan aliansi ini adalah berinovasi dengan cepat dan menaggapi kebutuhan konsumen dengan lebih baik, dengan produk awalnya adalah *platform* Android. Dimana android dirancang untuk melayani kebutuhan operator telekomunikasi, manufaktur handset dan pengembangan aplikasi. OHA berkomitmen untuk membuat Android *opensource* dengan lisensi Apache versi 2.0.

Android pertama kali diluncurkan pada 5 November 2007, dan *smartphone*pertama yang digunakan sistem operasi Android dikeluarkan oleh T-Mobile dengan sebutan G1 pada bulan September 2008. Hingga saat ini Android telah merilis beberapa versi Android untuk menyempurnakan versi sebelumnya. Selain berdasarkan penomoran, pada setiap versi android terdapat kode nama berdasarkan nama-nama makanan. Hingga saat ini sudah terdapat beberapa versi yang telah diluncurkan, diantaranya: versi 1.5 dirilis pada 30 April 2009 diberi nama Cupcake, versi 1.6 dirilis pada 15 September 2009 diberi nama Donut, versi 2.0 dirilis pada 26 Oktober 2009 diberi nama Eclair, versi 2.2 diberi nama Frozen Yoghurt atau Froyo, versi 2.3 yang diberi nama Gingerbread, versi 3.0 yang dikhusukan untuk tablet, dengan kode nama Honeycomb, versi 4.0 diberi nama Ice cream Sandwich atau ICS, versi 4.1-4.3 dengan nama Jellybean, versi 4.4+ diberi nama Kitkat dan yang terakhir untuk versi terbarunya adalah versi 5.0 diberi nama Lolipop.

#### 2.2.6.2 Anatomi Android

Dalam paket sistem operasi Android terdiri dari beberapa unsur. Secara sederhana arsitektur android merupakan sebuah  *kernel* Linux dan sekumpulan pustaka C / C++ dalam suatu  *framework* yang menyediakan dan mengatur alur proses aplikasi.

1. **Linux Kernel**

Android dibangun diatas *kernel* Linux 2.6. namun secara keseluruhan Android bukanlah Linux, karena dalam Android tidak terdapat paket standar yang dimiliki oleh linux lainnya. Linux merupakan sistem operasi terbuka yang handal dalam menejemen memori dan proses. Oleh karenanya pada Android hanya terdapat beberapa  *service* yang diperlukan seperti keamanan, menejemen memori, menejemen proses, jaringan dan  *driver*. *Kernel* Linux menyediakan *driver* layar, kamera, *keypad*, WiFi,  *Flash Memory*, audio, dan IPC (*Interprocess Comunication*) untuk mengatur aplikasi dan lubang keamanan.

1. **Libraries**

Android menggunakan beberapa paket pustaka yang terdapat pada C/C++ dengan Standar *Barkeley Software Distribution* (BSD) hanya setengah dari yang aslinya untuk tertanam pada *kernel*  Linux. Beberapa pustaka diantaranya:

1. *Media Library* untuk memutar berbagai macam format audio dan video.
2. *Surface Manager* untuk mengatur hak akses layer dan berbagai aplikasi.
3. *Graphic Library* termasuk didalamnya SGL dan OpenGL untuk tampilan 2D dan 3D.
4. SQLite untuk mengatur relasi  *database* yang digunakan pada aplikasi.
5. SSI dan WebKit untuk  *browser* dan keamanan internet.

Pustaka-pustaka tersebut bukanlah aplikasi yang berjalan sendiri, namun hanya dapat digunakan oleh program yang berada di level atasnya. Sejak versi Android 1.5, pengembang dapat membuat dan menggunakan pustaka sendiri menggunakan *Native Development Toolkit* (NDK).

1. **Android Runtime**

Pada Android tertanam paket pustaka inti yang menyediakan sebagian besar fungsi Android. Inilah yang membedakan Android dibandingkan dengan sistem operasi lain yang juga mengimplementasikan Linux. Android Runtime merupakan mesin virtual yang membuat aplikasi Android menjadi lebih tangguh dengan paket pustaka yang telah ada. Dalam Android Runtime terdapat dua bagian utama, diantaranya:

1. Pustaka inti, Android dikembangkan melalui bahasa pemrograman Java, tapi Android runtime bukanlah mesin *virtual* Java. Pustaka inti Android menyediakan hampir semua fungsi yang terdapat pada pustaka Java. Serta beberapa pustaka khusus Android.
2. Mesin Virtual Dalvik, Dalvik merupakan subuah mesin virtual yang dikembangkan oleh Dan Borstein yang terinspirasi dari nama sebuah perkampungan yang berada di Iceland. Dalvik hanyalah dengan interpreter mesin virtual yang mengeksekusi file dengan format Dalvik Executable (\*.dex). Dengan format ini Dalvik akan mengoptimalkan efisiensi penyimpanan dan pengalamatan memori pada file yang dieksekusi. Dalvik berjalan diatas *kernel* Linux 2.6, dengan fungsi dasar seperti *threading* dan manajemen memori yang terbatas.
3. **Aplication Framework**

Kerangka aplikasi menyediakan kelas-kelas yang dapat digunakan untuk mengembangkan aplikasi Android. Selain itu, juga menyediakan abstraksi generik untuk mengakses perangkat, serta mengatur tampilan *user interface* dan sumber daya aplikasi. Bagian yang terpenting dalam kerangka aplikasi Android adalah sebagai berikut:

1. *Activity Manager*, barfungsi untuk mengontrol siklus hidup aplikasi dan menjaga keadaan “*Backstack*” untuk navigasi penggunaan.
2. *Content Providers*, berfungsi untuk merangkum data yang memungkinkan digunakan oleh aplikasi lainnya, seperti daftar nama.
3. *Resource Manager*, untuk mengatur sumber daya yang ada dalam program. Serta menyediakan akses sumber daya diluar kode program, seperti karakter, grafik, dan *file Layout*.
4. *Location Manager*, berfungsi untuk memberikan informasi detail mengenai lokasi perangkat Android berada.
5. *Notification Manager,* mencakup berbagai macam peringatan seperti, pesan masuk, janji, dan lain sebagainya yang akan ditampilkan pada *status bar.*
6. **Application Layer**

Puncak dari diagram arsitektur Android adalah lapisan aplikasi dan *widget*. Lapisan aplikasi merupakan lapisan yang paling tampak pada pengguna ketika menjalankan program. Pengguna hanya akan melihat program ketika digunakan tanpa mengetahui proses yang terjadi dibalik lapisan aplikasi. Lapisan ini berjalan dalam *Android Runtime* dengan menggunakan kelas dan *service* yang tersedia pada  *framework* aplikasi.

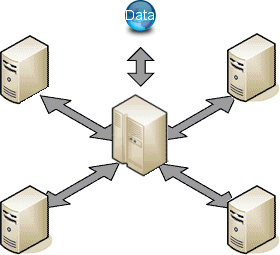
Lapisan aplikasi Android sangat berbeda dibandingkan dengan sistem operasi lainnya. Pada Android semua aplikasi, baik aplikasi inti (*native*) maupun aplikasi pihak ketiga berjalan diatas lapisan aplikasi dengan menggunakan pustaka API (*Aplication Progamming Interface*) yang sama.

### *2.2.7 Client-Server*

Pada jaringan tipe ini terdapat sebuah PC perangkat keras, dan server yang berfungsi untuk mengatur dan membagikan informasi ke setiap PC client yang terhubung dengannya. Server (pelayan) menyediakan sarana pada client (pemakai jasa) untuk mengambil data, sharing perangkat keras, dan mengonfigurasi security pada suatu jaringan. (Ahmad Yani. 2008:4)

#### 2.2.7.1 arsitektur Client Server

Model *client server* memisahkan secara jelas antara *server* dan *client.Server* memberikan layanan jaringan dan *client* menerima layanan. Beberapa komputer di *setup* sebagai *server* yang memberikan segala sumber daya yang tersedia dalam jaringan, misalnya printer, modem kepada komputer lain yang terkoneksi ke jaringan yang berfungsi sebagai *client*. *Server* dan *client* dapat berkomunikasi menggunakan aplikasi jaringan yang disebut *server* program pada *server* dan *client* program pada *client*.



**Gambar 2.1** Arsitektur *Client Server*

Prinsip kerja jaringan dengan arsitektur ini sangat sederhana, dimana *server* akan menunggu permintaan dari *client*, memproses dan memberikan hasilnya kepada *Client*, sedangkan *Client* akan mengirimkan permintaan ke *Server*, menunggu proses dan melihat visualisasi hasil prosesnya. Sistem *ClientServer* ini menggunakan protokol utama TCP/IP (*Transmission Control Protocol* / *Internet Protocol*), dan sistem operasi yang digunakan adalah Windows NT.

#### 2.2.7.2 komponen Dasar Client Server

*Client server* dibentuk oleh tiga komponen dasar, yaitu *Client*, *Middleware,* dan *Server.*

1. *Client*

*Client* merupakan terminal yang digunakan oleh pengguna untuk meminta layanan tertentu yang dibutuhkan. Melalui terminal *Client* ini, pengguna dapat mengirimkan pesan berupa permintaan layanan ke *Server* dan menerima hasil pemrosesan yang diminta dari *Server.*

1. *Middleware*

*Middleware* merupakan komponen perantara yang memungkinkan *Client* dan *Server* untuk saling terhubung dan berkomunikasi satu sama lain. *Middleware* ini dapat berupa *Remote Procedur Calling* (RPC), *Object Request Broker* (ORB) atau *Transaction Monitor* (TP).

*Middleware* memiliki peran yang strategis karena dengan adanya *Middleware* maka *Client* dapat mengirimkan pesan atau permintaan kepada *Server*, menerjemahkan pesan dari *Client* agar dapat dimengerti oleh *Server*, menerjemahkan hasil proses dari *Server* agar dapat dipahami oleh *Client* dan mengirimkan hasil proses yang telah diterjemahkan kembali ke *Client*.

1. *Server*

*Server* merupakan pihak yang menyediakan layanan. *Server* ini dapat berupa basis data SQL, atau *web*. Secara umum *Server* berperan menerima pesan permintaan dari *Client*, memproses permintaan tersebut dan mengirimkan hasilnya kepada *Client*.

#### 2.2.6.3 Karakteristik Client Server

*Client Server* memiliki karakteristik sebagai berikut :

1. *Server* menyediakan layanan yang dibutuhkan dan diminta oleh *Client*.
2. *Server* mengelola sejumlah sumber daya yang dimilikinya agar dapat diakses dan digunakan secara bersama-sama oleh *Client* yang terhubung ke *Server*.
3. Hubungan yang terjadi antara *Client* dan Server adalah *one to many*, yang berarti satu Server melayani banyak *Client*. *Client* selalu memulai interaksi dengan meminta layanan sedang *Server* hanya menunggu permintaan *Client* secara pasif.
4. *Server* dapat terletak diberbagai tempat yang belum tentu diketahui oleh *Client*. Walaupu begitu *Client* tetap dapat mengakses *Server* untuk mendapat layanan sesuai kebutuhannya.
5. Perangkat keras dan perangkat lunak yang digunakan oleh masing-masing *Client* tidak harus sama dengan yang digunakan oleh *Server,* namun masih dapat saling terkoneksi antara satu dengan yang lainnya.
6. Mekanisme dari *Client/Server* berdasar pada pertukaran pesan permintaan layanan dan hasil dari permintaan layanan tersebut.
7. *Client* tidak perlu mengetahui proses pengolahan permintaan yang terjadi dalam *Server* sehingga *Client* tidak dapat mengontrol proses pengolahan permintaan.
8. Ukuran *Client Server* dapat diubah secara horizontal yaitu menambah atau mengurangi jumlah *Client* dan secara vertikal yaitu berpindah ke *Server* yang lebih cepat atau lebih besar dan mendistribusikan tugas melayani *Client* ke beberapa *Server.*
9. Data hanya dikelola pada *Server* pusat sehingga konsistensi data lebih terjamin dan biaya pemeliharaan pun menjadi lebih murah.

Beberapa keuntungan dalam menggunakan *Client Server* adalah :

1. *Client Server* mampu menciptakan aturan dan kewajiban komputansi secara terdistribusi
2. Mudah dalam *maintenance,* memungkinkan untuk mengganti, memperbaiki *Server* tanpa mengganggu *Client.*
3. Semua data disimpan di *Server,* hanya yang memiliki autorisasi yang dapat mengakses sumber daya.
4. Mendukung banyak *Client* berbeda dengan kemampuan yang berbeda pula.

Beberapa kerugian dalam menggunakan *Client Server* adalah :

1. Jika banyak *Client* yang mengakses ke *Server* secara simultan, maka *Server* akan *overload*.
2. Adanya kemungkinan *Server fail*.

#### 2.2.6.4 Hambatan Implementasi Sistem Client Server

1. Hambatan terhadap organisasi, yaitu :
   * + 1. *Skill* personel yang kurang memadai untuk implementasi sistem *Client Server*.
       2. Anti perubahan terhadap teknologi baru.
       3. Biaya konversi.
       4. Membutuhkan koordinasi dan kontrol yang lebih pada end user.
2. Hambatan terhadap teknologi
3. Membutuhkan infrastruktur LAN dan WAN.
4. Skill dan peralatan yang belum memadai.
5. Belum adanya pemahaman dan pengalaman dalam merencanakan sistem *Client Server*.
6. Tidak tersedianya produk dan *tools* pengembangan sistem *Client Server*.
7. Sedikitnya aplikasi *Client Server.*
8. Sedikitnya standar nasional dan internasional untuk sistem *Client Server.*

### 2.2.8 Pengertian PHP (*Hypertext Processor*)

PHP *Hypertext Preprocessor* adalah suatu bahasa scripting khususnya digunakan untuk membuat web development. Karena sifatnya yang server side scripting, maka untuk menjalankannya harus menggunakan web server. PHP juga dapat diintergrasikan dengan HTML, Javascript, Jquery, Ajax. Namun pada umunnya PHP lebih banyak digunakan dengan file bertipe HTML. ( Priyanto Hidayatullah, Jauhari Khairul K., 2014: 232)

#### 2.2.8.1 Sejarah PHP

PHP pertama kali ditemukan oleh Rasmus Lerdorf pada tahun 1995, dia adalah seorang *programmer* unix dan perl pada waktu itu. PHP pada waktu Form Interpreted ( FI ), yang wujudnya berupa sekumpulan script yang digunakan untuk mengelola data formulir dari web. Selanjutnya ramus merilis kode sumber tersebut untuk umum dan menamakannya PHP/FI. Dengan perilisan kode sumber ini menjadi sumber terbuka, maka banyak pemrogramer yang tertarik untuk ikut mengembangkan PHP. Pada November 1997 dirilis PHP/FI 2.0 kemudian bulan juni 1998 rilis 3.0, sehingga perusahaan bernama Zend menulis ulang interpreter PHP menjadi lebih bersih, lebih baik, dan lebih cepat,. pada pertengahan tahun 1999 Zend merilis PHP 4.0, dan pada juni 2004 Zend merilis PHP 5.0. ( Priyanto Hidayatullah, Jauhari Khairul K., 2014: 233).

#### 2.2.8.2 Kelebihan PHP

PHP berbasis server side scripting. PHP sendiri dapat melakukan tugas-tugas yang dilakukan dengan mekanisme CGI seperti mengambil, mengumpulkan data dari database, meng-generate halaman dinamis, atau bahkan menerima dan mengirim cookie. Dan yang menjadi keutamaan PHP itu sendiri adalah PHP digunakan diberbagai operating sistem, diantaranya Linux, Unix, Windows, Mac OsX, RISC OS, dan Operating System lainnya.( Priyanto Hidayatullah, Jauhari Khairul K., 2014: 234).

Beberapa kelebihan PHP adalah :

1. PHP berbasis server side scripting,
2. Command line scripting pada PHP,
3. PHP dapat membuat Aplikasi Desktop,
4. Digunakan untuk berbagai macam platform OS
5. Mendukung berbagai macam Web Server,
6. Mendukung banyak RDMS ( Database ),
7. Dan lain-lain.

### 2.2.9 MYSQL

Mysql merupakan salah satu relasional *database management* system yang bersifat *Open Source*. Struktur *database* disimpan dalam tabel-tabel yang saling berelasi. Karena bersifat *open source*, maka Mysql dapat dipergunakan dan didistribusikan baik untuk kepentingan individu maupun *corporate* secara gratis, tanpa memerlukan lisensi dari pembuatnya. Mysql dapat dijalan dalam berbagai *platform* sistem operasi antara lain Windows, Linux, Unix, Sun OS dan lain-lain.

#### 2.2.9.1 Sejarah MySQL

MySQL dikembangkan oleh pengembangan database dan konsultan database bernama MySQL AB sekitar tahun 1994 di swedia. Tujuan awal dikembangkan MySQL yaitu untuk mengembangkan aplikasi berbasis web pada client. Michael Monty Widenius yang akrab disapa Monty melakukan pengembangan MySQL AB yang memiliki sebuah aplikasi UNIREG dan ISAM buatannya yang pada waktu itu ingin membuat tampilan antar muka ( user interface ) SQL ( Structured Query Language )yang cocok untuk diimplementasikan.

Pada saat itu monty telah menggunakan user interface buatan David Hughes pada aplikasi SQL-nya dengan nama miniSQL atau disingkat mSQL, namun pada kenyataannya mSQL dirasa kurang atau sangat lambat dalam melakukan pemrosesan query, sehungga dianggap menghambat kinerja dari SQL buatannya.

Saat Monty mengalami permasalahan pada user interface untuk aplikasi SQL buatannya, kemudian Monty menghubungi David Hughes kembali untuk menyempurnakan mSQL buatannya agar dapat menjalankan perintah SQL lebih baik lagi. Setelah David Hughes yang dibantu Monty berhasil menyempurnakan mSQL maka lahirlah MySQL yang memiliki kemampuan yang lebih baik dari mSQL.

MySQL adalah sebuah perangkat lunak sistem management basis data SQL ( Database Management System ) ata DBMS yang multithread, Multi-user, dengan sekitar 6 juta installasi di seluruh dunia. MySQL merupakan perangkat lunak (*software)* gratis dibawah lisensi GPL ( *GNU General Publik License* ).

MySQL dapat disebut sebagi sebuah implementasi dari siste manajement basis data relasional (RDBMS – Relational Database Management System ) yang didistribusikan secara gratis. Setiap pengguna dapat secara bebsa menggunakan MySQL namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersial.

MySQL sebenarnya merupakan turunan salah satu konsep utama dalam basis data yang telah ada sebelumnya, yaitu SQL. SQL adalah sebuah konsep pengoperasian basis data, terutama untuk pemilihan atau seleksi dan input data, yang memungkinkan pengoperasian data dikerjakan dengan mudah. (Yosef Murya & Rizki Hesananda, 2014 :166).

DBMS biasanya menawarkan beberapa kemampuan yang terintegrasi seperti :

1. Membuat, Menghapus, Menambah, dan memodifikasi basis data,
2. Pada beberapa DBMS pengelolaannya berbasis windows ( Berbentuk jendela-jendela) sehingga memudahkan digunakan
3. Tidak semua orang bisa mengakses basis data yang ada sehingga memberikan keamanan bagi data
4. Kemampuan berkomunikasi dengan program aplikasi yang lain.
5. Kemampuan pengaksesan melalui komunikasi antar komputer ( client server ). ( Priyanto H. & Jauhari K.K, 2014:179 ).