

BAB II

TINJAUAN PUSTAKA

II.1. Sistem Informasi

II.1.1. Definisi Sistem

Istilah sistem sekarang ini banyak dipakai, banyak orang berbicara mengenai sistem perbankan, sistem akuntansi, sistem inventori, sistem persediaan, sistem pendidikan, sistem perangkat lunak, sistem tata surya, sistem teologi dan masih banyak lagi. Sebuah sistem terdiri atas bagian-bagian atau komponen yang terpadu untuk suatu tujuan. Model dasar dari bentuk sistem ini adalah adanya masukan, pengolahan dan keluaran. Akan tetapi, sistem ini dapat dikembangkan hingga menyertakan media penyimpanan. Sistem dapat terbuka dan tertutup, akan tetapi sistem informasi biasanya adalah sistem terbuka, artinya sistem tersebut dapat menerima beberapa masukan dari lingkungan luarnya. (Tata Sutabri; 2005: 3)

Secara sederhana, suatu sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur, komponen atau variabel yang terorganisir, saling berinteraksi, saling tergantung satu sama lain dan terpadu. Kecenderungan manusia yang mendapat tugas memimpin suatu organisasi adalah terlalu memusatkan perhatian pada salah satu komponen saja dari sistem organisasi. Teori sistem mengatakan bahwa setiap unsur pembentuk organisasi adalah penting dan harus mendapat perhatian yang utuh supaya manajer dapat bertindak lebih efektif. (Tata; 2005: 2)

II.1.2. Definisi Informasi

Belum ada metode untuk mengukur informasi dalam sebuah sistem dan kerumitan informasi tidak memungkinkan adanya suatu rumus atau algoritma untuk menghitung isinya. Informasi adalah sebuah istilah yang tidak tepat dalam pemakaiannya secara umum. Informasi dapat mengenai data mentah, data tersusun, kapasitas sebuah saluran komunikasi, dan lain sebagainya. Informasi ibarat darah yang mengalir di dalam tubuh suatu organisasi sehingga informasi ini sangat penting di dalam suatu organisasi. Suatu sistem yang kurang mendapatkan informasi manajemen berhubungan dengan informasi, tetapi apakah informasi itu ?. Informasi adalah data yang telah diklarifikasi atau diolah atau diinterpretasi untuk digunakan dalam proses pengambilan keputusan.(Tata Sutabri; 2005: 23)

Sistem pengolahan informasi mengolah data dari bentuk tidak berguna menjadi informasi atau tepatnya mengolah data dari bentuk tidak berguna menjadi berguna bagi penerimanya. Nilai informasi berhubungan dengan keputusan. Bila tidak ada pilihan atau keputusan, maka informasi menjadi tidak diperlukan. Keputusan dapat berkisar dari keputusan berulang sederhana sampai keputusan strategis jangka panjang. Nilai informasi dilukiskan paling berarti dalam konteks sebuah keputusan.(Tata Sutabri; 2005: 24)

II.1.3. Definisi Sistem Informasi

Komputer telah menambah satu atau dua dimensi, seperti kecepatan, ketelitian dan volume data yang meningkat yang memungkinkan pertimbangan alternatif-alternatif yang lebih banyak dalam suatu keputusan, yang di dalam suatu organisasi terdiri atas sejumlah unsur, orang yang mempunyai bermacam-macam

peran dalam organisasi, kegiatan atau tugas yang harus diselesaikan, tempat bekerja, wewenang pekerjaan, serta hubungan komunikasi yang mengikat bersama organisasi tersebut. Sistem informasi manajemen (SIM) merupakan penerapan sistem informasi di dalam organisasi untuk mendukung informasi-informasi yang dibutuhkan oleh semua tingkatan manajemen. Telah diketahui bahwa informasi merupakan hal yang sangat penting bagi manajemen di dalam pengambilan keputusan.(Tata Sutabri; 2005: 41)

Dari mana informasi tersebut bisa didapat ? Informasi dapat diperoleh dari sistem informasi. Sistem informasi merupakan suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan.(Tata Sutabri; 2005: 42)

II.2. Interaksi Manusia dan Komputer

Interaksi manusia dan komputer atau dalam bahasa Inggris lebih lazim disebut sebagai *human computer interaction* (HCI) merupakan sebuah disiplin ilmu yang mempelajari desain, evaluasi, implementasi dari sistem komputer interaktif untuk dipakai oleh manusia, beserta studi tentang faktor-faktor utama dalam lingkungan interaksinya atau disiplin ilmu yang melibatkan interaksi antara pengguna dan komputer dengan melibatkan berbagai disiplin ilmu yang lain.(Soetam Rizky; 2007: 3)

Deskripsi yang lain dari IMK adalah suatu ilmu yang mempelajari perencanaan dan desain tentang cara manusia dan komputer saling bekerja sama, sehingga manusia dapat merasa puas dengan cara yang paling efektif. Dikatakan juga bahwa sebuah desain antar muka yang ideal adalah yang mampu memberikan kepuasan terhadap manusia sebagai pengguna dengan faktor kapabilitas serta keterbatasan yang terdapat dalam sistem.(Soetam Rizky; 2007: 4)

II.3. Client-Server

II.3.1. Konsep Client-Server

Client-server adalah salah satu model komunikasi 2 komputer atau lebih yang berfungsi melakukan pembagian tugas. *Client* bertugas untuk melakukan *input, update*, penghapusan dan menampilkan data sebuah *database*. Sementara *server* bertugas menyediakan pelayanan untuk melakukan manajemen, yaitu menyimpan dan mengolah *database*.(Wahana Komputer; 2010: 5)

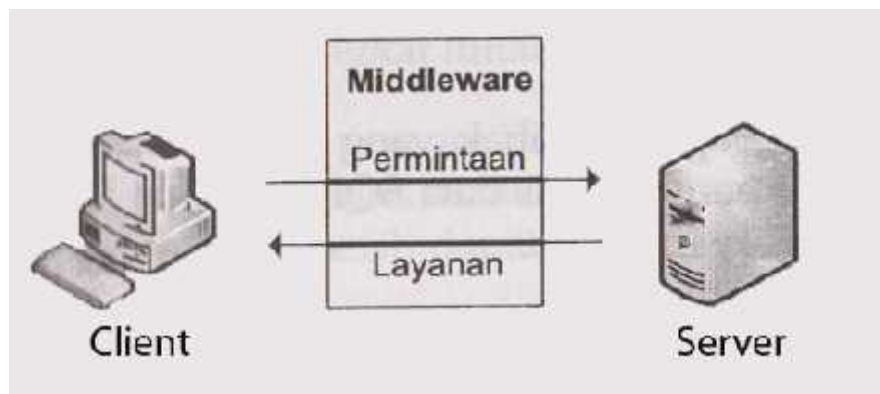
Arsitektur jaringan *client-server* merupakan model konektivitas pada jaringan yang membedakan fungsi komputer apakah sebagai *client* atau sebagai *server*. Arsitektur ini menempatkan sebuah komputer sebagai *server* yang bertugas memberikan layanan kepada terminal-terminal lain (*client*) yang terhubung dalam sistem jaringan itu. *Server* dapat bertugas untuk memberikan layanan berbagi pakai berkas (*file server*), *printer* (*printer server*), jalur komunikasi (*Server komunikasi*). (Budi Sutedjo; 2006: 76)

Pada model arsitektur ini , *client* tidak dapat berfungsi sebagai *server* tetapi *server* dapat berfungsi sebagai *client* (*server non-dedicated*). Namun hal ini

sebaiknya dihindari agar arsitektur *client-server* tidak berubah menjadi arsitektur P2P (*peer to peer*). 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 *client-server* ini tidak hanya diperuntukkan bagi pembangunan sistem jaringan komputer lokal, melainkan juga pembangunan jaringan komputer skala luas. Sistem ini menggunakan *protocol* utama *Transmission Control Protocol / Internet Protocol (TCP/IP)*.(Budi Sutedjo; 2006: 76)

II.3.2. Komponen Dasar Client-Server

Pada dasarnya, *client-server* dibentuk oleh tiga komponen dasar, yaitu *client*, *middleware* dan *server*. Secara sederhana, hubungan antar ketiga komponen tersebut dapat digambarkan sebagai berikut :



Gambar II.1. Komponen Dasar Client-Server
(Budi Sutedjo; 2006: 77)

1. Client

Client merupakan terminal yang digunakan oleh pengguna untuk meminta layanan tertentu yang dibutuhkan. Terminal *client* dapat berupa PC,

ponsel, komunikator, robot, televisi dan peralatan lain yang membutuhkan informasi. Di sini peran *client* adalah mengirimkan pesan berupa permintaan layanan ke *server*, misalnya *Automatic Teller Machine* (ATM) yang meminta informasi saldo dari nasabah tertentu. Melalui terminal *client* ini, pengguna dapat memasukkan data dan mengakses forum untuk transaksi. Terminal *client* ini juga digunakan untuk menerima hasil pemrosesan yang diminta dari *server*.

2. Middleware

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

3. Server

Server merupakan pihak yang menyediakan layanan. *Server* ini dapat berupa basis data SQL, *Monitor TP*, *server groupware*, *server* objek atau web. Secara umum, *server* berperan menerima pesan permintaan layanan

dari *client*, memproses permintaan tersebut dan mengirimkan hasil permintaan kepada *client*.(Budi Sutedjo; 2006: 76)

II.3.3. Ciri-Ciri Client-Server

Client-server memiliki ciri-ciri mendasar yang membedakan dirinya dengan perangkat lunak terdistribusi lain. Ciri-ciri tersebut adalah :

1. Berbasis layanan

Server memberikan sejumlah layanan yang dibutuhkan dan diminta oleh *client*, antara lain : berbagi pakai berkas, data, peralatan pendukung dan lain-lain.

2. Sumber daya yang digunakan bersama

Server mengelola sejumlah sumber daya yang dimilikinya agar dapat diakses dan digunakan secara bersama-sama oleh terminal-terminal *client* yang terhubung pada *server*.

3. Hubungan dan interaksi client server

Hubungan yang terjadi antara *server* dan *client* adalah *one-to-many*, yang berarti bahwa satu *server* melayani banyak *client*. *Client* selalu memulai interaksi dengan meminta layanan, sedangkan *server* menanti permintaan layanan secara pasif.

4. Client tidak perlu mengetahui lokasi fisik server

Server dapat terletak di berbagai tempat yang belum tentu diketahui oleh *client*. Walaupun demikian, *client* tetap dapat mengakses *server* untuk mendapatkan layanan sesuai kebutuhannya.

5. Interoperabilitas perangkat lunak dan perangkat keras

Perangkat keras dan perangkat lunak yang digunakan oleh masing-masing *client* tidak harus sama dengan yang digunakan pada *server*, namun masih dapat saling terkoneksi antara satu dan yang lain.

6. Pertukaran berbasis pesan

Mekanisme dari *client-server* berdasar pada pertukaran pesan. Pesan yang dipertukarkan ialah permintaan layanan dan umpan balik dari permintaan layanan tersebut.

7. Enkapsulasi layanan

Client tidak perlu mengetahui proses pengolahan permintaan yang terjadi dalam *server* sehingga *client* tidak dapat mengontrol proses pengolahan permintaan.

8. Skalabilitas

Skalabilitas ialah kemampuan untuk diperbesar atau diperkecil. Ukuran sistem *client-server* dapat diubah secara horizontal maupun vertikal. Pengubahan vertikal berarti berpindah ke *server* yang lebih besar atau lebih cepat atau mendistribusikan tugas melayani *client* ke beberapa *server*. Pengubahan horizontal berarti menambah atau mengurangi jumlah *client*.

9. Konsistensi data

Data hanya dikelola pada *server* pusat sehingga konsistensi dan data lebih terjamin dan biaya pemeliharaan pun menjadi lebih murah. (Budi Sutedjo; 2006: 78)

II.3.4. Alasan Menggunakan Client-Server

Dari gambaran umum tentang *client-server* yang dipaparkan di atas, dapat dirasakan keuntungan-keuntungan yang diperoleh dengan mengaplikasikan jaringan *client-server*. Terdapat 3 alasan untuk menggunakan *client-server* yaitu :

1. Kapasitas

Dengan semakin berkembangnya teknologi, makin besar pula data yang perlu ditampung. Kapasitas satu komputer tidak akan sanggup untuk menangani seluruh data yang digunakan pengguna. Bila sebagian besar data dialihkan ke *server* untuk digunakan bersama oleh banyak *client* maka hal ini akan jauh meringankan beban *client*.

2. Keamanan

Dengan menampung seluruh data di komputer *client*, maka pengguna akan memiliki akses langsung ke data tersebut. Hal ini menjadi suatu masalah bila data yang disimpan tersebut merupakan data rahasia yang hanya boleh diakses segelintir pengguna. Dengan menyimpan data di *server* dan membatasi pengguna hanya untuk mengakses antarmuka *server*, keamanan data dapat lebih terjamin.

3. Penghematan

Dalam sistem *client-server*, sumber daya yang dimiliki *server* dapat digunakan oleh terminal-terminal *client* sehingga biaya investasi sumber daya dapat dihemat. (Budi Sutedjo; 2006: 90)

II.4. Jaringan Wi-Fi

Komunikasi tanpa kabel/nirkabel (*wireless*) telah menjadi kebutuhan dasar atau gaya hidup baru masyarakat informasi. LAN nirkabel yang lebih dikenal dengan jaringan Wi-Fi menjadi teknologi alternatif dan relatif lebih mudah untuk diimplementasikan di lingkungan kerja seperti di perkantoran, laboratorium komputer, pusat perbelanjaan, restoran dan sebagainya. Instalasi perangkat jaringan Wi-Fi lebih fleksibel karena tidak membutuhkan penghubung kabel antar komputer. Tidak seperti halnya LAN (*Local Area Network*)/jaringan yang menggunakan jenis kabel koaksial dan kabel UTP (*Unshielded Twisted Pair*) sebagai media transfer. Komputer dengan Wi-Fi *device* dapat saling terhubung yang hanya membutuhkan ruang atau *space* dengan syarat jarak jangkauan dibatasi kekuatan pancaran sinyal radio dari masing-masing komputer. (Tri Kuntoro; 2005: 1)

II.4.1. Definisi Wi-Fi

Wi-Fi atau *Wireless Fidelity* adalah satu standar *wireless networking* tanpa kabel, hanya dengan komponen yang sesuai dapat terkoneksi ke jaringan. Teknologi Wi-Fi memiliki standar yang ditetapkan oleh sebuah institusi internasional yang bernama *Institute of Electrical and Electronic Engineers* (IEEE), yang secara umum sebagai berikut :

1. Standar IEEE 802.11a yaitu Wi-Fi dengan frekuensi 5 Ghz yang memiliki kecepatan 54 Mbps dan jangkauan jaringan 300 m.
2. Standar IEEE 802.11b yaitu Wi-Fi dengan frekuensi 2,4 Ghz yang memiliki kecepatan 11 Mbps dan jangkauan jaringan 100 m.

3. Standar IEEE 802.11g yaitu Wi-Fi dengan frekuensi 2,4 Ghz yang memiliki kecepatan 54 Mbps dan jangkauan jaringan 300 m.

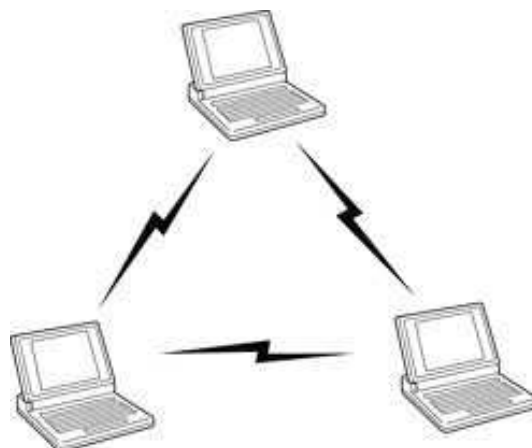
Teknologi Wi-Fi yang akan diimplementasikan adalah standar IEEE 802.11g karena standar tersebut lebih cepat untuk proses transfer data dengan jangkauan jaringan yang lebih jauh. Perangkat tersebut bekerja di frekuensi 2,4 Ghz atau disebut sebagai pita frekuensi ISM (Industrial, Scientific and Medical) yang juga digunakan oleh peralatan lain, seperti Microwave oven, cordless phone dan bluetooth. (Tri Kuntoro; 2005: 1)

II.4.2. Tipe Jaringan Wi-Fi

Seperti halnya Ethernet-LAN (jaringan dengan kabel), jaringan Wi-Fi juga dikonfigurasi ke dalam dua jenis jaringan :

1. Jaringan peer to peer/Ad Hoc wireless LAN

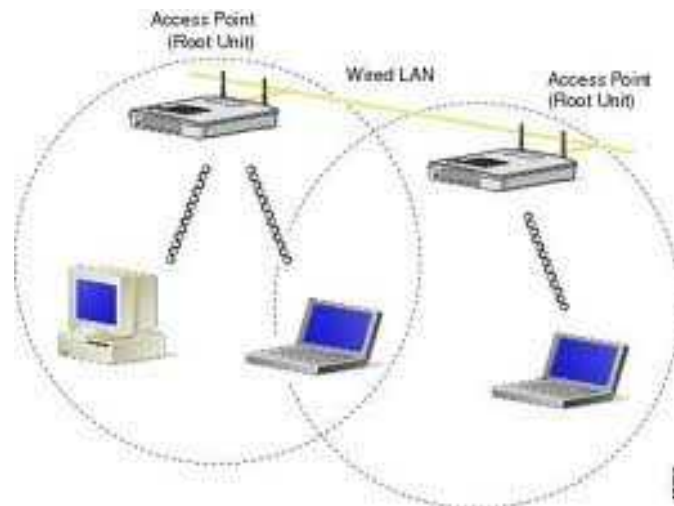
Komputer dapat saling berhubungan berdasarkan nama SSID (Service Set Identifier). SSID adalah nama identitas komputer yang memiliki komponen nirkabel.



Gambar II.2. Ad Hoc Wireless LAN
(Tri Kuntoro; 2005: 2)

2. Jaringan Server Based/Wireless Infrastructure

Sistem infrastruktur membutuhkan sebuah komponen khusus yang berfungsi sebagai Access Point. (Tri Kuntoro; 2005: 2)



Gambar II.3 Wireless Infrastructure
(Tri Kuntoro; 2005: 3)

II.5. Java

Java lahir tahun 1991 ketika *Sun MicroSystem* melakukan penelitian terhadap berbagai produk elektronika. James Gosling (kepala proyek penelitian) beserta Patrick Naughton ditugaskan untuk merancang perangkat lunak aplikasi yang independen yang dapat berjalan pada perangkat yang memiliki memori ukuran kecil. Selain itu mereka juga menginginkan program yang dapat berjalan di *platform* manapun, ini dikarenakan setiap perangkat mempunyai manufaktur yang berbeda. (Eko Priyo Utomo, 2009: 3)

Penamaan Java terinspirasi ketika James Gosling bingung mencari nama untuk bahasa baru tersebut karena setiap nama yang dipikirkannya untuk nama bahasa pemrogramannya ternyata sudah ada yang memakai, sampai suatu ketika

dia mampir di sebuah kafe untuk minum kopi, dan ia meminta “Java”. Timbullah idenya untuk memakai nama JAVA.

II.5.1. Teknologi Java

Secara garis besar teknologi java terbagi menjadi beberapa bagian, berdasarkan buku Panduan Mudah Mengenal Bahasa Java (Eko Priyo Utomo, 2009: 19), yaitu sebagai berikut:

1. J2SE (Standart Edition)

Teknologi java ini dirancang untuk membuat aplikasi yang berjalan ada PC dan *workstation* yang berbeda *platform* *Windows*, *Linux*, *Macintosh*. J2SE terbagi menjadi dua bagian besar yaitu J2SE *Core* dan J2SE *Desktop*. J2SE *Core* digunakan untuk teknologi meliputi beberapa teknologi yaitu JRE (*Java Runtime Environment*), JFC (*Java Foundation Classes*), *Java Sound API* dan sebagainya.

2. J2EE (*Enterprise Edition*)

Teknologi java ini digunakan untuk pengembangan aplikasi-aplikasi *enterprise*. Meliputi beberapa teknologi yaitu JSP (*Java Server Page*), *Java Servlet*, *COBRA* (untuk aplikasi terdistribusi) dan sebagainya.

3. J2ME (*Micro Edition*)

Teknologi java ini digunakan untuk pengembangan sistem *micro* dan sistem *embedded* seperti *handphone*, PDA dan lain sebagainya. Meliputi dua bagian besar yaitu *CLDC* (*MIDP*, *Bluetooth* dan lain sebagainya) dan *CDC Tecnology* (*JDBC/teknologi database* dan *RMI*).

4. *Java Web Service*

Merupakan Aplikasi *web* berbasis *enterprise* dengan standar *XML* dan protokol tertentu untuk bertukar data dengan klien. Teknologi ini meliputi beberapa *API* yang dirancang untuk bekerjasama dengan *XML* seperti *Java API for XML Based RPC (JAX-RPC)*, *Java API for XML Based Messaging (JAXM)*, *Java API for XML Processing (JAXP)* dan *Java API for XML Binding (JAXB)*.

II.5.2. Keunggulan Java

Java memiliki keunggulan yang tidak dimiliki oleh bahasa lain, diantaranya:

1. Sederhana

Sintaks pada *java* telah banyak diperbaiki terutama menghilangkan penggunaan pointer yang rumit dan *multiple inheritance*.

2. Berorientasi objek

Java menggunakan pemrograman berorientasi objek yang membuat program dapat dibuat secara modular dan dapat dipergunakan kembali.

3. Dapat didistribusikan dengan mudah

Java dibuat untuk membuat aplikasi terdistribusi secara mudah dengan adanya *libraries networking* yang terintegrasi.

4. Interpreter

Program *java* dijalankan menggunakan interpreter yaitu *Java Virtual Machine (JVM)*.

5. Robust

Java mempunyai reliabilitas yang tinggi. *Java* mempunyai *runtime-exception handling* untuk membantu mengatasi eror pada pemrograman.

6. Aman

Java memiliki beberapa mekanisme keamanan untuk menjaga aplikasi agar tidak digunakan untuk merusak sistem computer yang menjalankan aplikasi tersebut.

7. Arsitektur Netral

Program *Java* bersifat platform independent. Program cukup mempunyai satu buah versi yang dapat dijalankan pada platform yang berbeda dengan JVM

8. *Portable*

Kode sumber maupun program *Java* dapat dengan mudah dibawa ke platform yang berbeda-beda tanpa harus dikompilasi ulang.

9. Kinerja

Kinerja pada *java* sering dikatakan kurang tinggi. Namun kinerja *Java* dapat ditingkatkan dengan menggunakan kompilasi *Java* lain.

10. *Multithreaded*

Java mempunyai kemampuan untuk membuat suatu program yang dapat melakukan beberapa pekerjaan sekaligus dan simultan.

11. Dinamis

Java didesain untuk dapat dijalankan pada lingkungan yang dinamis.

II.6. MySQL

MySQL (*My Structure Query Language*) atau yang biasa dibaca “mai-sekuel” adalah sebuah program pembuat *database* (sistem manajemen *database*) yang bersifat *open source*, artinya siapa saja boleh menggunakannya dan tidak dicekal. Kelebihan MySQL adalah ia menggunakan bahasa *query* standar yang dimiliki SQL. SQL adalah suatu bahasa permintaan yang terstruktur yang telah distandarkan untuk semua program pengakses *database* seperti *Oracle*, *posgres*, *SQL server* dan lain-lain. (Agus Saputra, 2011: 2)

Ada beberapa pertimbangan mengapa memilih MySQL, antara lain:

1. Kecepatan. Berdasarkan hasil pengujian, MySQL memiliki kecepatan paling baik diantara *database server* lainnya.
2. Mudah digunakan. Perintah-perintah dan aturan-aturan pada MySQL maupun proses instalasinya relatif mudah digunakan.
3. *Open source*. Dengan konsep ini siapapun dapat berpartisipasi mengembangkan MySQL dan hasil pengembangan itu diserahkan kepada umum atau kepada komunitas *open source*.
4. Kapabilitas. MySQL telah digunakan untuk mengelola *database* dengan jumlah 50 juta record.
5. Replikasi data. Dengan adanya replikasi data ini, anda dapat mempunyai beberapa *database* bayangan pada beberapa server anak lainnya yang berasal dari satu *database* induk sehingga akan meningkatkan kinerja dan kecepatan MySQL.
6. Biaya rendah (relatif gratis).

7. Konektifitas dan keamanan.
8. Fleksibilitas/Portabilitas. MySQL mendukung perintah-perintah ANSI SQL 99 dan beberapa perintah database alternatif lainnya sehingga memudahkan untuk beralih dari dan ke MySQL.
9. Lintas platform sistem operasi. MySQL dapat dijalankan pada beberapa sistem operasi yang berbeda, seperti Linux, Microsoft Windows dan lain-lain. (Agus Saputra, 2011: 8)

II.7. UML (*Unified Modelling Language*)

Unified Modelling Language (UML) adalah suatu alat untuk memvisualisasikan dan mendokumentasikan hasil analisa dan desain yang berisi sintak dalam memodelkan sistem secara visual (Braun, *et. al.* 2001). Juga merupakan satu kumpulan konvensi pemodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem *software* yang terkait dengan objek (Whitten, *et. al.* 2004). (Haviluddin; 2011: 1)

Sejarah UML sendiri terbagi dalam dua fase; sebelum dan sesudah munculnya UML. Dalam fase sebelum, UML sebenarnya sudah mulai diperkenalkan sejak tahun 1990an namun notasi yang dikembangkan oleh para ahli analisis dan desain berbeda-beda, sehingga dapat dikatakan belum memiliki standarisasi. (Haviluddin; 2011: 1)

Fase kedua, dilandasi dengan pemikiran untuk mempersatukan metode tersebut dan dimotori oleh Object Management Group (OMG) maka pengembangan UML dimulai pada akhir tahun 1994 ketika Grady Booch dengan metode OOD (*Object-Oriented Design*), Jim Rumbaugh dengan metode OMT

(*Object Modelling Technique*) mereka ini bekerja pada Rational Software Corporation dan Ivar Jacobson dengan metode OOSE (*Object-Oriented Software Engineering*) yang bekerja pada perusahaan Objectory Rational.(Haviluddin; 2011: 1)

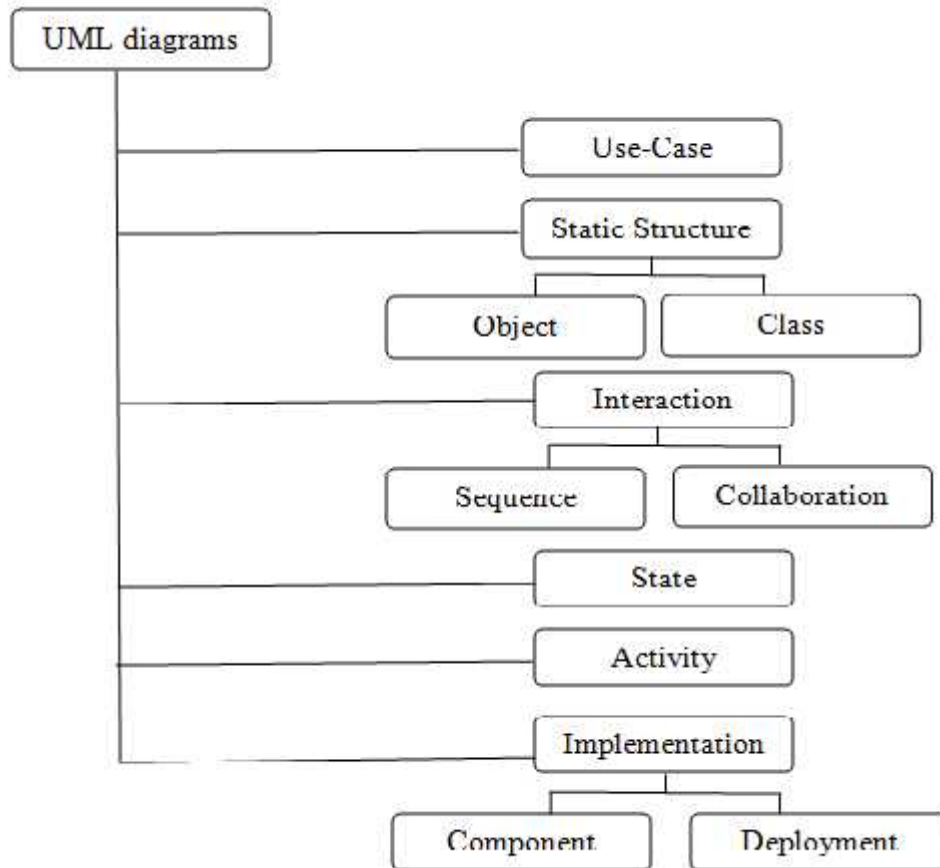
Sebagai pencetus metode-metode tersebut mereka bertiga berinisiatif untuk menciptakan bahasa pemodelan terpadu sehingga pada tahun 1996 mereka berhasil merilis UML versi 0.9 dan 0.91 melalui Request for Proposal (RFP) yang dikeluarkan oleh OMG.

Kemudian pada Januari 1997 IBM, ObjecTime, Platinum Technology, Ptech, Taskon, Reich Technologies dan Softeam juga menanggapi Request for Proposal (RFP) yang dikeluarkan oleh OMG tersebut dan menyatakan kesediaan untuk bergabung.(Haviluddin; 2011: 1)

Perusahaan-perusahaan ini menyumbangkan ide-ide mereka, dan bersama para mitra menghasilkan UML revisi 1.1. Fokus dari UML versi rilis 1.1 ini adalah untuk meningkatkan kejelasan UML Semantik versi rilis 1.0. Hingga saat ini UML versi terbaru adalah versi 2.0.

Saat ini sebagian besar para perancang sistem informasi dalam menggambarkan informasi dengan memanfaatkan UML diagram dengan tujuan utama untuk membantu tim proyek berkomunikasi, mengeksplorasi potensi desain, dan memvalidasi desain arsitektur perangkat lunak atau pembuat program. Secara filosofi UML diilhami oleh konsep yang telah ada yaitu konsep permodelan *Object Oriented* karena konsep ini menganalogikan sistem seperti kehidupan nyata yang didominasi oleh obyek dan digambarkan atau dinotasikan

dalam simbol-simbol yang cukup spesifik. Berikut gambar dari diagram *UML* yaitu :



Gambar II.4. Diagram UML
(Haviluddin; 2011: 2)