

BAB 2

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Adapun beberapa penelitian yang terkait dengan pemilihan karyawan terbaik, diantaranya adalah :

Penelitian yang sudah dilakukan oleh Iwan Rijayana dkk [2] tentang sistem pendukung keputusan pemilihan karyawan berprestasi menggunakan metode Analytic Hierarchy Process (AHP). Penelitian ini menghasilkan sebuah sistem untuk pendukung keputusan dalam pemilihan karyawan yang berprestasi berdasarkan kinerja dari karyawan dan mampu menampilkan sepuluh besar karyawan terbaik melalui grafik yang ditampilkan.

Penelitian yang di lakukan oleh Richie Cindy Anggria dkk [3] yaitu penerapan metode Fuzzy TOPSIS dalam sistem pendukung keputusan penilaian kinerja dan jabatan karyawan Balai Penelitian Sembawa yang menghasilkan sebuah sistem pendukung keputusan penilaian kinerja dan jenjang jabatan karyawan. Sistem yang dibuat secara otomatis memproses data penilaian ketika admin memasukkan nilai dan akan diproses oleh sistem secara cepat dan akurat.

Tabel 2.1 Penelitian Terkait

No	Nama Peneliti dan Tahun	Masalah	Metode	Hasil
1	Iwan rijayana, Lirien Okirindho, 2012	Pemilihan karyawan berprestasi didasarkan pada beberapa faktor yaitu penilaian kinerja, score TOEIC, dan	Metode Analytic Hierarchy Process (AHP)	Sistem pendukung keputusan pemilihan karyawan yang berprestasi berdasarkan kinerja dari karyawan dan

No	Nama Peneliti dan Tahun	Masalah	Metode	Hasil
		<p>kedisiplinan kerja (kehadiran karyawan).</p> <p>Pada saat ini proses penilaian kinerja karyawan masih dalam bentuk hardcopy dan keputusan dari satu pihak saja sehingga proses yang dilakukan masih belum akurat.</p>		<p>menampilkan sepuluh besar karyawan terbaik melalui grafik yang ditampilkan.</p>
2	<p>Richie Cindy Anggria, Afriyudi, dan Febriyanti panjaitan, 2015</p>	<p>penilaian kinerja karyawan pada balai penelitian sembawa ditentukan berdasarkan Daftar Penilaian Prestasi Kerja. Masalah yang dihadapi yaitu penilaian kurang cepat dan masih dalam bentuk hardcopy.</p>	<p>Metode TOPSIS</p>	<p>Sistem pendukung keputusan penilaian kinerja dan jenjang jabatan karyawan. sistem secara otomatis memproses data penilaian ketika admin memasukkan nilai dan sistem menilai secara cepat dan akurat.</p>

2.2 Konsep Sistem Pendukung Keputusan

2.2.1 Pengertian Pengambilan Keputusan

Menurut Simon dalam Turban [1], pengambilan keputusan merupakan sebuah proses yang memilih tindakan diantara berbagai alternatif yang ada untuk mencapai suatu tujuan ataupun beberapa tujuan.

Dengan kata lain, pengambilan keputusan yaitu hasil suatu proses komunikasi dan partisipasi sebagai wujud dari pencapaian tujuan yang diharapkan, sehingga pengambilan keputusan sangatlah penting sebagai dasar untuk membangun rencana kedepannya.

2.2.2 Pengertian Sistem Pendukung Keputusan

Dalam Turban [1] Konsep awal sistem pendukung keputusan dikenalkan pertama kali oleh Scott Morton pada awal tahun 1970-an. Ia mendefinisikan DSS sebagai sistem berbasis komputer interaktif yang membantu para pengambil keputusan untuk menggunakan data dan berbagai model untuk memecahkan masalah-masalah tidak terstruktur. Sistem pendukung keputusan atau *Decision Support System* merujuk pada sebuah sistem yang dimaksud untuk mendukung para pengambil keputusan manajerial dalam situasi keputusan semiterstruktur.

Kursini [5] mendefinisikan sistem pendukung keputusan merupakan sistem informasi interaktif yang menyediakan informasi, pemodelan dan manipulasi data. Sistem itu digunakan untuk membantu pengambilan keputusan dalam situasi yang semiterstruktur dan situasi tidak terstruktur, dimana tidak seseorang pun tau secara pasti bagaimana keputusan seharusnya dibuat.

Dari dua definisi diatas maka dapat ditarik kesimpulan bahwa sistem pendukung keputusan merupakan sebuah sistem yang mampu memberikan penilaian terhadap alternatif-alternatif yang berguna untuk membantu para manajer atau pihak yang bersangkutan dalam pengambilan keputusan.

2.2.3 Karakteristik dan Kapabilitas Sistem Pendukung Keputusan

Karakteristik dan kapabilitas sistem pendukung keputusan menurut Turban [1] adalah sebagai berikut :

1. Dukungan untuk pengambilan keputusan terutama pada situasi semiterstruktur dan tak terstruktur, dengan menyertakan penilaian manusia dan informasi terkomputerisasi.
2. Dukungan untuk semua level manajerial, mulai dari eksekutif puncak sampai manajer lini.
3. Dukungan untuk individu dan kelompok, masalah yang kurang terstruktur sering memerlukan keterlibatan individu dari departemen dan tingkat organisasi yang berbeda beda atau bahkan dari organisasi lain.
4. Dukungan untuk keputusan independen atau sekuensial. Keputusan dapat dibuat satu kali, beberapa kali bahkan berulang kali (dalam interval yang sama).
5. Dukungan di semua fase proses pengambilan keputusan seperti intelegensi, desain, pilihan, dan implementasi.
6. Dukungan berbagai proses dan gaya pengambilan keputusan.
7. Adaptif setiap waktu. Pengambilan keputusan seharusnya reaktif, dapat menghadapi perubahan kondisi secara cepat dan bisa mengadaptasikan DSS untuk memenuhi perubahan tersebut.
8. Pengguna merasa seperti dirumah. Ramah pengguna, kapabilitas grafis yang sangat kuat dan antarmuka manusia-mesin interaktif dengan satu bahasa alami dapat sangat meningkatkan keefektifan DSS.
9. Peningkatan terhadap keefektifan pengambilan keputusan (akurasi, *timeliness*, kualitas) ketimbang pada efisiensinya (biaya pengambilan keputusan).
10. Kontrol penuh oleh pengambil keputusan terhadap semua langkah proses pengambilan keputusan dalam memecahkan masalah.
11. Pengguna akhir dapat mengembangkan dan memodifikasi sendiri sistem sederhana. Sistem yang lebih besar dapat dibangun dengan bantuan ahli sistem informasi. Perangkat lunak OLAP dalam kaitannya dengan data

warehouse membolehkan pengguna untuk membangun DSS yang cukup besar dan kompleks.

12. Biasanya model-model digunakan untuk menganalisa situasi pengambilan keputusan.
13. Akses disediakan untuk berbagai sumber data, format dan tipe mulai dari sistem informasi geografis (GIS) sampai sistem berorientasi objek.
14. Dapat dilakukan sebagai alat *standalone* yang digunakan oleh seorang pengambil keputusan pada satu lokasi atau didistribusikan disatu organisasi keseluruhan dan dibeberapa organisasi sepanjang rantai persediaan.

2.2.4 Tujuan Sistem Pendukung Keputusan

Kursini [5] menjelaskan tujuan sistem pendukung keputusan, ialah sebagai berikut :

1. Membantu manajer dalam pengambilan keputusan atas masalah semi terstruktur.
2. Memberikan dukungan atas pertimbangan manajer, tidak bermaksud untuk menggantikan fungsi manajer.
3. Meningkatkan efektivitas keputusan yang diambil oleh manajer lebih dari pada perbaikan efisiensinya.
4. Kecepatan komputasi. Komputer memungkinkan para pengambil keputusan untuk melakukan banyak komputasi secara cepat dengan biaya yang relatif rendah.
5. Peningkatan produktifitas. Membangun satu kelompok pengambilan keputusan, terutama para pakar bisa sangat mahal.
6. Dukungan kualitas. Komputer bisa meningkatkan kualitas keputusan yang dibuat.
7. Berdaya saing. Manajemen dan pemberdayaan sumber daya perusahaan, tekanan persaingan menyebabkan tugas pengambilan keputusan menjadi sulit.
8. Mengatasi keterbatasan kognitif dalam pemrosesan dan penyimpanan.

2.2.5 Komponen Sistem Pendukung Keputusan

Tiga komponen penting yang terdapat dalam sistem penunjang keputusan menurut Hermawan [6], diantaranya adalah :

1. Manajemen Data

Manajemen data yaitu dengan mengambil data yang dibutuhkan, data yang berasal dari penyimpanan data yang berisikan data internal ataupun eksternal. Oleh karena itu fungsi dari manajemen data adalah sebagai pengatur data yang dibutuhkan oleh sebuah sistem penunjang keputusan.

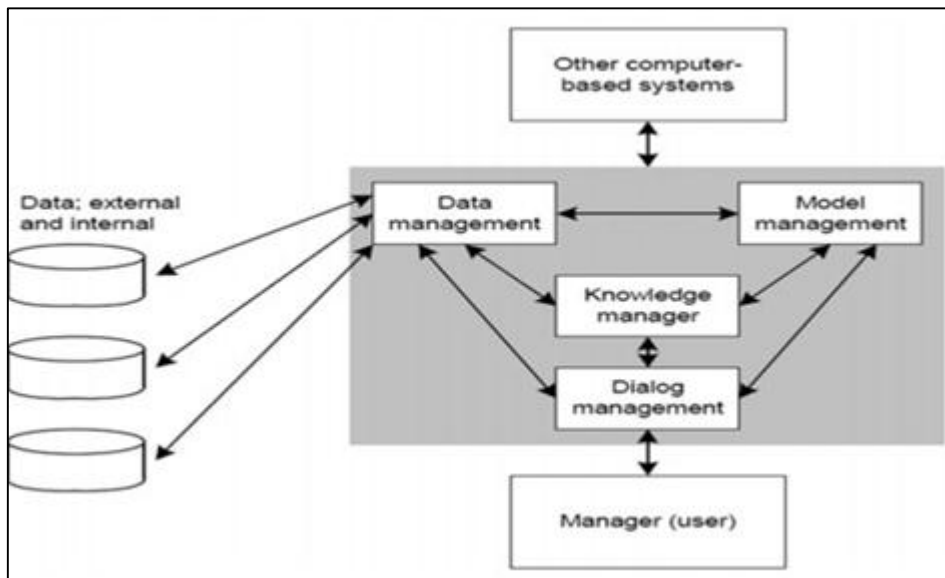
2. Manajemen Model

Manajemen model melakukan hubungan timbal balik dengan user interface untuk memperoleh komando ataupun data management untuk sebuah data yang hendak proses. Dengan tujuan untuk merubah data pada sebuah database menjadikan suatu informasi yang dapat digunakan untuk pengambilan sebuah keputusan.

3. *User Interface*

Berguna untuk berhubungan oleh user dengan sistem pendukung keputusan, melakukan input data/informasi kedalam sebuah sistem ataupun untuk memunculkan data/informasi kepada pengguna. Antarmuka pengguna sangatlah penting sehingga dalam perancangan ataupun penmbuatannya harus dibuat dengan semudah mungkin untuk dapat dipahami oleh penggunanya.

Dari komponen yang ada maka dapat dibentuk sebuah sistem penunjang keputusan yang dapat dihubungkan pada jaringan suatu perusahaan. desain sebuah sistem penunjang keputusan dapat dilihat pada gambar dibawah ini:



Gambar 2.1 Arsitektur Sistem Pendukung Keputusan

2.3 Karyawan

Menurut Subri [7], karyawan merupakan setiap penduduk yang masuk ke dalam usia kerja (berusia di rentang 15 hingga 64 tahun), atau jumlah total seluruh penduduk yang ada pada sebuah negara yang melakukan produksi barang atau jasa bila terdapat permintaan akan tenaga yang mereka produksi, dan apabila mereka ingin berkecimpung atau beradaptasi dalam kegiatan tersebut.

Sedangkan menurut Hasibuan [8], karyawan adalah setiap orang yang menyediakan jasa (baik dalam bentuk pikiran maupun dalam bentuk tenaga) sehingga memperoleh balas jasa atau kompensasi yang besarnya telah ditetapkan.

Dari pengertian diatas maka dapat disimpulkan bahwa karyawan adalah setiap orang yang memberikan jasa kepada perusahaan ataupun organisasi yang membutuhkan jasa tenaga kerja yang mana dari jasa tersebut akan mendapatkan balas jasa berupa gaji dan kompensasi-kompensasi lainnya.

2.3.1 Jenis-jenis Karyawan di Perusahaan

Jika di kelompokkan berdasarkan statusnya, karyawan dalam perusahaan dapat dibagi menjadi dua jenis kelompok karyawan, yaitu :

1. Karyawan Tetap

Karyawan tetap merupakan karyawan yang telah memiliki kontrak ataupun perjanjian kerja dengan perusahaan dalam jangka waktu yang tidak ditetapkan (*permanent*). Karyawan tetap biasanya cenderung memiliki hak yang jauh lebih besar dibandingkan dengan karyawan tidak tetap. Selain itu, karyawan tetap juga cenderung jauh lebih aman (dalam hal kepastian lapangan pekerjaan) dibandingkan dengan karyawan yang tidak tetap.

2. Karyawan Tidak Tetap

Karyawan tidak tetap merupakan karyawan yang hanya dipekerjakan ketika perusahaan membutuhkan tenaga kerja tambahan saja. Karyawan tidak tetap biasanya dapat diberhentikan sewaktu-waktu oleh perusahaan ketika perusahaan sudah tidak membutuhkan tenaga tambahan lagi. Jika dibandingkan dengan karyawan tetap, karyawan tidak tetap cenderung memiliki hak yang jauh lebih sedikit dan juga cenderung sedikit tidak aman (dalam hal kepastian lapangan pekerjaan).

2.3.2 Kinerja Karyawan

Definisi kinerja karyawan menurut Mangkunegara [9] kinerja merupakan hasil dari kerja secara kuantitas dan kualitas yang berhasil diperoleh karyawan dalam pelaksanaan tugas yang sesuai dengan tanggung jawabnya.

Kinerja kerja yaitu hasil yang diperoleh seseorang dalam penyelesaian tugas yang diberikan kepadanya. Hasibuan [10].

Berdasarkan definisi diatas maka dapat diambil kesimpulan yaitu kinerja karyawan adalah suatu hasil kerja dari seseorang karyawan dalam pelaksanaan tugas sesuai dengan tanggung jawab dan berpengaruh terhadap pencapaian tujuan organisasi.

2.4 *Analytical Hierarchy Process (AHP)*

Metode analytic hierarchy process (AHP) yang dikembangkan oleh Thomas L. Saaty merupakan metode yang bisa digunakan untuk mengatasi masalah yang kompleks, pengambilan kriteria yang lumayan banyak, susunan masalah yang tidak begitu jelas, ketersediaan data statistik yang belum tentu benar.

Kursini [5], menjelaskan dalam penyelesaian masalah menggunakan metode analytic hierarchy process terdapat prinsip yang patut dimengerti, antara lain :

1. Pembuatan Hirarki

Untuk memahami sistem yang sangat kompleks dapat menggunakan cara memecah dan dijadikan komponen komponen pendukung, kemudian menyusun kembali komponen tersebut secara hirarki.

2. Penilaian Kriteria dan Alternatif

Kriteria dan alternatif dilakukan dengan perbandingan berpasangan. Menurut Saaty [5], dalam penyelesaian beragam masalah, skala 1 - 9 merupakan skala yang paling baik dalam mengekspresikan sebuah gagasan. Berikut ini adalah tabel kriteria perbandingan berpasangan menurut Saaty :

Tabel 2.2 Kriteria Perbandingan Berpasangan

Konsep dan Aplikasi Sistem Pendukung Keputusan

Intensitas Kepentingan	Keterangan
1	Kedua elemen sama pentingnya
3	Elemen yang satu sedikit lebih penting daripada elemen yang lainnya
5	Elemen yang satu lebih penting daripada elemen yang lainnya
7	Satu elemen jelas lebih mutlak penting daripada elemen lainnya
9	Satu elemen mutlak penting daripada elemen lainnya
2,4,6,8	Nilai-nilai antara dua nilai pertimbangan yang berdekatan
Kebalikan	Jika aktivitas i mendapat satu angka dibanding dengan aktivitas j, maka j memiliki nilai kebalikannya dibandingkan dengan i

3. Menentukan Prioritas

Perbandingan berpasangan/berganda digunakan untuk menentukan prioritas setiap kriteria dan alternatif. Nilai perbandingan kriteria dan alternatif bisa disamakan pada ketentuan yang ada untuk dijadikan nilai prioritas. Untuk perhitungan nilai prioritas dapat dilakukan dengan cara manipulasi matriks ataupun dengan menyelesaikan persamaan matematik.

4. Konsistensi Logis

Dalam hal ini konsistensi mempunyai dua arti. Yang pertama yaitu objek yang satu rupa dapat dijadikan satu sepadan pada jenisnya dan relevansinya. Yang Kedua yaitu mengenai tingkatan sangkut paut antara objek berdasarkan kriteria tertentu.

Berikut ini merupakan langkah-langkah dari metode analytic hierarchy process antara lain :

1. Pengidentifikasian permasalahan dan menetapkan solusi yang akan digunakan, setelah itu penyusunan hirarki. Penyusunan hirarki dilakukan dengan cara mempatenkan tujuan dari sistem secara keseluruhan.
2. Menentukan prioritas elemen
 - a. Pertama-tama membuat perbandingan berganda, yaitu dengan membandingkan elemen secara berganda sesuai dengan kriteria yang diberikan.
 - b. Matrik perbandingan berganda diisikan dengan bilangan, guna merepresentasikan kepentingan relatif elemen satu dengan lainnya.
3. Sintesis

Untuk memperoleh keseluruhan prioritas maka harus memperrtimbangan perbandingan berganda yang telah disintesiskan. langkah lagngkah yang harus dilakukan dalam hal ini adalah sebagai berikut :

 - a. Nilai tiap kolom pada matriks harus dijumlahkan.
 - b. Selanjutnya nilai dari tiap kolom di bagi dengan total kolom yang bersangkutan guna mendapatkan normalisasi matriks.

- c. Untuk mendapatkan nilai rata-rata harus menjumlah nilai tiap kolom dan dibagi dengan jumlah elemennya.

4. Mengukur Konsistensi

Mengetahui seberapa baik konsisten yang ada sangat penting untuk membuat sebuah keputusan, karena keputusan didasarkan pada pertimbangan konsistensi yang kecil sangat tidak diinginkan. Langkah-langkah yang dilakukan pada hal ini adalah :

- a. Mengalikan nilai yang ada pada tiap kolom dengan prioritas relatifnya. nilai yang ada di kolom kesatu dengan prioritas relatif yang kesatu, kedua dengan yang kedua dan begitu selanjutnya.
- b. Jumlahkanlah tiap baris.
- c. Pembagian elemen prioritas relatif yang berhubungan dengan hasil penjumlahan baris.
- d. Setelah itu untuk mengetahui λ_{max} dengan cara menjumlahkan hasil dari pembagian diatas dengan banyak elemen yang ada.

5. Persamaan untuk menghitung *consistency* indeks (CI) :

$$CI = \frac{\lambda_{max} - n}{n - 1} \quad (2.1)$$

n adalah banyaknya elemen

6. Persamaan untuk menghitung *consistency ratio* (CR) :

$$CR = \frac{CI}{RI} \quad (2.2)$$

CI adalah *Consistency* indeks

IR adalah Indeks *Random Consistency*

CR adalah *Consistency Ratio*

7. Mengecek kesesuaian hirarki.

Apabila nilai 10% lebih maka penilaiannya harus dibenahi, namun apabila CR samadengan ataupun kurang 0,1 itu berarti hitungannya dapat dikatakan benar.

Berikut ini adalah daftar IR :

Tabel 2.3 Daftar Indeks Random Konsistensi (IR)

Konsep dan Aplikasi Sistem Pendukung Keputusan

Ukuran Matriks	Nilai IR
1,2	0,00
3	0,58
4	0,90
5	1,12
6	1,24
7	1,32
8	1,41
9	1,45
10	1,49
11	1,51
12	1,48
13	1,56
14	1,57
15	1,59

Dan hasilnya prioritas global yang memiliki nilai paling tinggilah yang digunakan para pengambil keputusan.

2.5 Desain Sistem

Menurut Pratama B [11], desain sistem sebagai penentu sebuah sistem untuk menyelesaikan perintah yang harus diselesaikannya. Tahap ini bersangkutan dengan konfigurasi dari komponen sebuah perangkat lunak dan perangkat kerasnya. Sehingga sesudah instalasi, sistem akan sangat memuaskan dan sesuai dengan rancang bangun yang telah ditetapkan pada tahap akhir analisis sitem.

Dari uraian diatas dapat disimpulkan bahwa desain sistem adalah penggambaran yang berupa rancangan ataupun elemen elemen yang disatukan menjadi kesatuan utuh yang memiliki fungsi dan dapat bermanfaat.

2.5.1 Tahap Desain Sistem

Tahap desain sistem meliputi :

1. Desain Output

Yaitu sebuah produk/hasil sistem informasi yang mampu dilihat pada sebuah media seperti kertas, media elektronik ataupun dari hasil pemrosesan yang akan digunakan oleh proses lain dan disimpan pada sebuah media penyimpanan seperti hardisk. Desain output bertujuan untuk menjelaskan dokumen dan laporan dari sistem.

2. Desain Input

Desain input juga dapat diartikan sebagai rancangan input dari dokumen dasar yang digunakan untuk memasukan data, bahasa pemrograman yang digunakan serta bentuk tampilan input/masukan yang dirancang pada alat input.

3. Desain Database

Perancangan database bertujuan untuk mendeskripsikan isi ataupun struktur dari data ataupun file yang berada pada database guna memenuhi kebutuhan.

4. Implementasi Sistem

Tahap ini bertujuan untuk menjelaskan implementasi dari desain-desain yang telah dibuat menjadi sebuah sistem informasi. Tahap dari implementasi antara lain :

a. Programing dan testing

Untuk mengkonversi dari desain-desain sistem menjadi perintah dalam bahasa pemrograman dan menguji dari semua program guna memastikan bahwa proses program berjalan dengan baik.

b. Pelatihan dan persiapan lainnya

Tujuannya untuk melakukan pengenalan sistem, pelatihan sistem, pengenalan tempat, serta beberapa tugas lainnya.

c. Pergantian atau perubahan sistem

Perubahan dari sistem lama ke sistem yang baru. Penyerahan tanggung jawab sistem yang baru dari tim pengembang kepada pemakain.

2.6 Pengujian Sistem

Menurut Quadri dan Farooq [12], pengujian software adalah proses verifikasi dan validasi apakah sebuah aplikasi software atau program memenuhi persyaratan bisnis dan persyaratan teknis yang mengarahkan desain dan pengembangan dan cara kerjanya seperti yang diharapkan dan juga mengidentifikasi kesalahan yang penting yang digolongkan berdasarkan tingkat severity pada aplikasi yang harus diperbaiki.

Menurut Nidhra dan Dondeti [13], pengujian software adalah teknik yang sering digunakan untuk verifikasi dan validasi kualitas suatu software. Pengujian software adalah prosedur untuk eksekusi sebuah program atau sistem dengan tujuan untuk menemukan kesalahan.

Kesimpulan yang dapat diambil dari pendapat-pendapat tersebut adalah pengujian software merupakan proses verifikasi dan validasi apakah software memenuhi requirement dan mengidentifikasi kesalahan-kesalahan yang ditemukan saat eksekusi program.

2.6.1 Black Box Testing

Menurut Nidhra dan Dondeti [13], black box testing juga disebut functional testing, sebuah teknik pengujian fungsional yang merancang test case berdasarkan informasi dari spesifikasi.

Menurut Pressman [14], Pengujian black-box berfokus pada persyaratan fungsional perangkat lunak. Dengan demikian, pengujian black-box memungkinkan perekayasa perangkat lunak mendapatkan serangkaian kondisi input yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program. Pengujian black-box bukan merupakan alternative dari teknik white-box, tetapi merupakan pendekatan komplementer yang kemungkinan besar mampu mengungkap kelas kesalahan daripada metode white-box.

Pengujian black-box berusaha menemukan kesalahan dalam kategori sebagai berikut :

1. Fungsi – fungsi yang tidak benar atau hilang
2. Kesalahan Interface,
3. Kesalahan dalam struktur data atau akses database eksternal,
4. Kesalahan Kinerja,
5. Inisialisasi dan kesalahan terminasi

2.7 UML

Menurut Booch [15] UML merupakan bahasa standar untuk pembuatan rancangan software. UML sering digunakan dalam penggambaran dan pembangunan, dokumen artifak dari *software intensive system*.

Sedangkan menurut Nugroho [16], *Unified Modeling Language* (UML) yaitu metodologi kolaborasi antara *Object Modeling Technique* (OMT), serta *Object Oriented Software Engineering* (OOSE) serta beberapa metode lain yang kerap digunakan guna analisa dan perancangan sistem dengan model orientasi objek.

Dari dua penjelasan diatas maka dapat disimpulkan bahwa UML adalah sebuah bahasa yang sering digunakan dalam penggambaran dalam perancangan sebuah sistem yang akan dibangun.

2.7.1 Langkah Langkah Penggunaan UML

Menurut Nugroho [16], langkah-langkah penggunaan UML sebagai berikut:

1. Pembuatan daftar proses bisnis mulai dari level paling tinggi guna pendefinisian aktivitas dan proses yang mungkin muncul.
2. Pemetaan use case setiap proses bisnis untuk mendefinisikan dengan tepat fungsional yang harus disediakan oleh sistem, kemudian perhalus use case diagram dan lengkapi dengan *requirement*, *constraints* dan catatan-catatan lain.
3. Buatlah deployment diagram guna pendefinisian arsitektur fisik sistem.

4. Definiskan requirement lain non fungsional, keamanan dan sebagainya yang harus disediakan oleh sistem.
5. Berdasarkan use case diagram, mulailah membuat activity diagram.
6. Definiskan obyek-obyek level atas *package* serta buatlah *sequence* dan kolaborasi untuk setiap alur pekerjaan, jika sebuah use case memiliki kemungkinan alur normal dan error, buat lagi satu diagram untuk masing-masing alur.
7. Buatlah rancangan antar muka pengguna untuk menjalankan skenario use case.
8. Berdasarkan model-model yang sudah ada, buatlah class diagram. Setiap domain dipecah menjadi hirarki class lengkap dengan atribut dan metodenya.
9. Pada tahap ini buatlah diagram komponen setelah class diagram dibuat, kita dapat melihat kemungkinan pengelompokkan class menjadi komponen-komponen.
10. Perhalus deployment diagram yang sudah dibuat. Detilkan kemampuan dan requirement piranti lunak, sistem operasi, jaringan dan sebagainya. Petakan komponen ke dalam node.

2.7.2 Model UML

Menurut Widodo [17], model-model dari UML antara lain :

1. *Class Diagram*
Bersifat statis, Diagram ini memperlihatkan himpunan kelas-kelas, antarmuka-antarmuka, kolaborasi-kolaborasi, serta relasi-relasi. Diagram ini umum dijumpai pada pemodelan sistem berorientasi objek. Meskipun bersifat statis, sering pula class diagram memuat kelas-kelas aktif.
2. *Package Diagram*
Diagram ini memperlihatkan kumpulan kelas-kelas, merupakan bagian dari diagram komponen.

3. *Usecase Diagram*

Bersifat statis. Diagram ini memperlihatkan himpunan use-case dan aktor-aktor. Diagram ini sangat penting untuk memodelkan perilaku suatu sistem yang dibutuhkan serta diharapkan pengguna.

4. *Sequence Diagram*

Bersifat dinamis. *Sequence diagram* adalah interaksi yang menekankan pada pengiriman pesan dalam suatu waktu tertentu.

5. *Communication Diagram*

Sebagai pengganti diagram kolaborasi UML yang menekankan organisasi struktural dari objek-objek yang menerima serta mengirim pesan.

6. *Statechart Diagram*

Memperlihatkan keadaan-keadaan pada sistem, memuat status, transisi, kejadian serta aktivitas.

7. *Activity Diagram*

Diagram aktivitas adalah tipe khusus dari diagram status yang memperlihatkan aliran dari suatu aktivitas ke aktivitas lainnya dalam sebuah sistem. Diagram ini sangat penting untuk pemodelan fungsi-fungsi suatu sistem dan memberi tekanan pada aliran kendali antar objek.

8. *Component Diagram*

Diagram komponen memperlihatkan organisasi serta ketergantungan sistem pada komponen-komponen yang telah ada sebelumnya.

9. *deployment diagram*

Diagram ini memperlihatkan konfigurasi saat aplikasi dijalankan. Memuat simpul-simpul beserta komponen-komponen yang di dalamnya. Kesembilan diagram ini tidak mutlak harus digunakan dalam pengembangan perangkat lunak, semuanya dibuat sesuai kebutuhan.

2.8 MySQL

MySQL merupakan sebuah jenis dari database server yang begitu terkenal. Karena SQL digunakan sebagai kaidah dasarnya untuk akses dari database sehingga menjadikan MySQL sangat populer. Selain itu mysql tidak berbayar atau bisa

disebut dengan gratis, tidak memungut biaya untuk penggunaannya. Mysql tergolong Relational Database Management System (RDBMS) maka dari itu, kata tabel, kolom bahkan baris dipakai di mysql. Menurut Kursini [18], di mysql suatu database berisi satu ataupun beberapa tabel. Serta tabel terdiri atas beberapa baris serta pada setiap barisnya berisi sejumlah kolom didalamnya.

2.9 PHP

PHP merupakan sebuah bahasa server side scripting yang bersatu dengan HTML guna membuat sebuah halaman web dinamis. dikarenakan php adalah server side scripting dengan demikian sintaksis dan komando php hendak dieksekusi pada server setelah itu hasilnya hendak dikirim ke browser dengan berformat html. Dengan begitu kode pemrograman yang dituliskan pada php tidak dapat dilihat pengguna lain dan dengan begitu security dari website lebih terpercaya. Dalam pembuatannya php itu didesain guna membuat sebuah web yang dinamis, website yang bisa dibentuk berdasarkan permintaan yang diinginkan sepertihalnya menampilkan isi database ke dalam sebuah website. Arief [19].

Keunggulan php yaitu kemampuan php melakukan hubungan ke beraneka ragam perangkat lunak DBMS, oleh sebab itu php bisa menciptakan sebuah web yang dinamis. Nyaris seluruh aplikasi yang berbasis website dapat diciptakan hanya dengan php, tetapi sebenarnya kekuatan dari php adalah hubungan database dengan website.