

2. Bagi Universitas Dian Nuswantoro
 - a. Menjalin kerjasama di bidang pengembangan teknologi antara pihak perusahaan dengan Universitas Nuswantoro, sehingga menguntungkan kedua belah pihak.
 - b. Mengetahui sejauh mana perkembangan mahasiswa dalam menyerap ilmu selama proses study.

3. Bagi Perusahaan

Sebagai bahan masukan perusahaan dalam mengolah data khususnya data berkaitan dengan persediaan barang agar dapat menambah efektifitas dan efisiensi perusahaan.

BAB 2

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Penelitian terkait adalah penelitian yang dilakukan sebelum penelitian yang dilakukan sekarang. Penelitian terkait memiliki hubungan dengan penelitian yang dilakukan sekarang. Ada beberapa penelitian terkait yang akan dijadikan sebagai referensi dari penelitian yang akan dilakukan sekarang, diantaranya adalah :

1. Penelitian terkait dilakukan oleh Romdoni Triat Widayantoro pada tahun 2013. Penelitian ini membahas tentang pengambilan keputusan yang dilakukan pada panitia pengadaan jasa pemborong dalam menentukan jasa pemborong proyek dengan memakai tiga kategori, yaitu administrasi, keuangan, dan teknis. Panitia pengadaan jasa pemborong merasa kesulitan dalam penilaian karena untuk menilai setiap unsur yang ada pada kriteria persyaratan. Dalam penilaian ini terjadi ketidakpuasan penyedia jasa pemborong terhadap penilaian yang dilakukan panitia pengadaan jasa pemborong. Metode pengembangan sistem yang digunakan menggunakan metode prototyping dan program yang digunakan adalah program visual basic 6.0. hasil dari penelitian ini adalah sistem pendukung keputusan yang dapat menentukan perusahaan yang menjadi pemenang jasa pemborong proyek beserta grafik penentuan yang lulus dan gugur.
2. Penelitian terkait selanjutnya adalah penelitian yang dilakukan oleh Giana pada tahun 2011. Judul dari penelitian ini adalah sistem pendukung keputusan pemilihan kontraktor proyek pada lelang pada Dinas Cipta Karya Kabupaten Indtamayu Dengan Menggunakan Metode Analytical Hierarhy Process. Permasalahan dari penelitian ini adalah karena proses pemilihan kontraktor yang membutuhkan waktu lama. Selain itu penyajian informasi mengenai pengadaan barang / jasa juga masih dilakukan secara konvensional. Sehingga informasi yang didapat mengenai pengadaan barang dan jasa kurang. Hal tersebut

membuat peserta yang mengikuti lelang menjadi kurang. Maka dari itu dibutuhkan sistem pendukung keputusan untuk pemilihan kontraktor lelang pada Dinas Cipta Karya. Metode dalam mengembangkan sistem menggunakan metode waterfall dengan penilaian Analytical Hierarchy Process. Hasil dari penelitian ini adalah informasi peserta lelang yang lolos pada tahap prakualifikasi dan pascakualifikasi.

Tabel 2. 1 Penelitian Terkait

No	Nama Peneliti dan Tahun	Masalah	Metode	Hasil
1.	Romdoni Triat Widayantoro, 2013	pengambilan keputusan yang dilakukan pada penitia pengadaan jasa pemborong dalam menentukan jasa pemborong proyek dengan memakai tiga kategori, yaitu administrasi, keuangan, dan teknis. Penitia pengadaan jasa pemborong merasa kesulitan dalam penilaian karena untuk menilai setiap unsur yang ada pada kriteria persyaratan. Dalam penilaian ini terjadi ketidakpuasan penyedia jasa pemborong	Prototyping, AHP	sistem pendukung keputusan vb 6.0 yang dapat menentukan perusahaan yang menjadi pemenang jasa pemborong proyek beserta grafik penentuan yang lulus dan gugur.

No	Nama Peneliti dan Tahun	Masalah	Metode	Hasil
		terhadap penilaian yang dilakukan panitia pengadaan jasa pemborong		
2.	Giana, 2011	proses pemilihan kontraktor yang membutuhkan waktu lama. Selain itu penyajian informasi mengenai pengadaan barang / jasa juga masih dilakukan secara konvensional. Sehingga informasi yang didapat mengenai pengadaan barang dan jasa kurang. Hal tersebut membuat peserta yang mengikuti lelang menjadi kurang. Maka dari itu dibutuhkan sistem pendukung keputusan untuk pemilihan kontraktor lelang pada Dinas Cipta Karya.	Waterfall, AHP	Hasil dari penelitian ini adalah informasi peserta lelang yang lolos pada tahap prakualifikasi dan pascakualifikasi

2.2 Definisi Sistem

Beberapa ahli telah menjelaskan beberapa pendapat mengenai sistem, diantaranya adalah :

Menurut Abdul Kadir(2009) menjelaskan bahwa sistem adalah sekumpulan dari beberapa elemen yang saling terkait untuk mencapai tujuan tertentu. [1]

Sedangkan menurut Sutarbi Tata (2012),“ sistem merupakan suatu kumpulan dari unsur, komponen atau variable terorganisir, saling berinteraksi, saling bergantung satu sama lain dan terpadu.”. [2]

Berdasarkan pendapat dari beberapa ahli tentang sistem, maka dapat disimpulkan bahwa sistem adalah sekumpulan dari unsur, komponen atau variable yang saling terkait, berinteraksi, dan saling bergantung untuk mencapai tujuan tertentu.

2.3 Sistem Pendukung Keputusan

Sistem pendukung keputusan merupakan sebuah sistem informasi berbasis komputer yang menyediakan informasi, pemodelan, dan manipulasi data, digunakan untuk membantu keputusan dalam situasi semi terstruktur dan tidak terstruktur dimana tak seorangpun tahu secara pasti bagaimana keputusan itu seharusnya dibuat. Sistem pendukung keputusan biasanya dibuat untuk membantu mengevaluasi setiap peluang yang bisa didapatkan. Yang dikembangkan untuk mendukung solusi atas permasalahan manajemen spesifik yang tidak terstruktur.

Aplikasi pada sistem pendukung keputusan digunakan dengan menggunakan data, memberikan antarmuka pengguna, dan menggabungkan pemikiran pengambil keputusan. Pada dasarnya sistem ini lebih ditujukan untuk mendukung manajemen dalam melakukan pekerjaan yang bersifat analitis dalam situasi yang kurang terstruktur dan dengan kriteria yang kurang jelas. Sistem pendukung keputusan tidak dimaksudkan untuk mengotomatisasi setiap pengambilan keputusan. Namun memberikan perangkat interaktif yang memungkinkan pengambil keputusan untuk melakukan berbagai analisis menggunakan model-model yang telah tersedia [3].

2.4 *Analytical Hierarchy Process (AHP)*

Analytical Hierarchy Process (AHP) adalah sebuah metode yang merupakan hierarki fungsional dengan input utamanya adalah persepsi manusia. Dengan hierarki suatu masalah kompleks dan tidak terstruktur dipecahkan salah satu model pengambilan keputusan yang sering digunakan AHP digunakan dengan tujuan untuk menyusun prioritas dari berbagai alternatif atau pilke dalam kelompok-kelompok tersebut diatur menjadi suatu bentuk hierarki. [5]

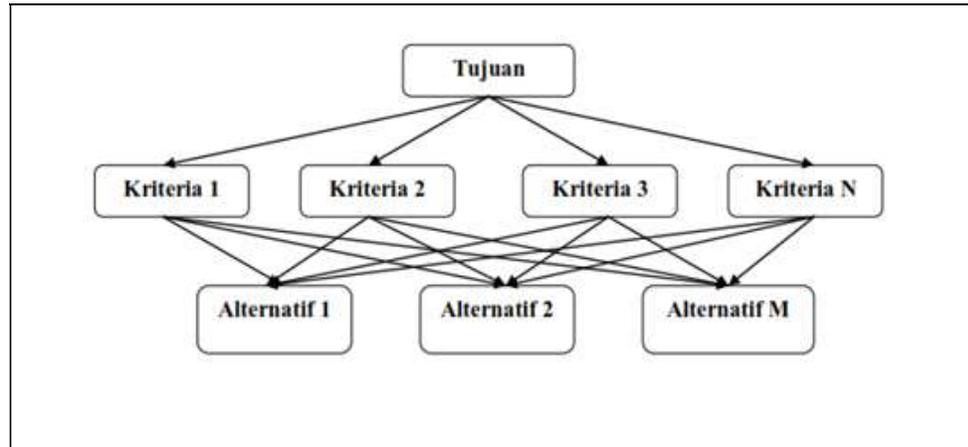
Analytical Hierarchy Process (AHP) merupakan salah satu metode untuk menyusun suatu prioritas dari berbagai pilihan dengan menggunakan beberapa kriteria (*multi kriteria*). Dikarenakan sifatnya yang multi kriteria, AHP cukup banyak digunakan dalam penyusunan prioritas. Selain memiliki sifat multi kriteria, AHP juga didasarkan pada suatu proses yang terstruktur dan logis.

AHP mamiliki beberapa keunggulan dalam menjelaskan tentang proses pengambilan keputusan. Salah satunya adalah dapat digambarkan secara grafis sehingga mudah dipahami oleh pihak-pihak yang terlibat dalam pengambilan keputusan. Kelebihan-kelebihan dari AHP adalah : [5]

1. Struktur yang memiliki hierarki sebagai konsistensi dari kriteria yang dipilih sampai ke sub kriteria yang paling dalam.
2. Memperhitungkan validasi sampai pada batas toleransi inkonsistensi berbagai kriteria dan alternatif yang dipilih oleh para pengambil keputusan.
3. Memperhitungkan daya tahan output analisis senvisitas pengambilan keputusan.

Tahapan – tahapan *Analytical Hierarchy Process (AHP)* adalah : [5]

1. Menyusun hirarki dari permasalahan yang dihadapi.
Penyusunan hirarki dilakukan dengan menentukan tujuan yang merupakan sasaran sistem secara keseluruhan pada level atas. Pada level berikutnya terdiri dari kriteria-kriteria untuk menilai atau mempertimbangkan alternatif-alternatif tersebut. Setiap kriteria dapat memiliki sub kriteria dibawahnya dan setiap kriteria memiliki nilai intensitas masing-masing.



Gambar 2. 1 Struktur Hierarki AHP

Sumber : Kusri (2007) [5]

2. Menentukan prioritas elemen.
 - a. Langkah pertama dalam menentukan prioritas elemen adalah membuat perbandingan berpasangan, yaitu membandingkan elemen secara berpasangan sesuai kriteria yang diberikan dengan menggunakan bentuk matriks. Matriks bersifat sederhana, berkedudukan kuat yang menawarkan kerangka untuk memeriksa konsistensi, memperoleh informasi tambahan dengan membuat semua perbandingan yang mungkin dan menganalisis kepekaan prioritas secara keseluruhan untuk merubah pertimbangan. Proses perbandingan berpasangan dimulai dari level paling atas hirarki untuk memilih kriteria, misalnya C , kemudian dari level dibawahnya diambil elemen-elemen yang akan dibandingkan, misal A_1, A_2, \dots, A_n , maka susunan elemen-elemen pada sebuah matriks sebagai berikut.

Tabel 2. 2 Matrik Perbandingan Berpasangan

C	A_1	A_2	...	A_n
A_1	a_{11}	a_{12}	...	a_{1n}
A_2	a_{21}	a_{22}	...	a_{2n}
:	:	:	...	:

A_m	a_{m1}	a_{m2}	...	a_{mn}
-------	----------	----------	-----	----------

- b. Mengisi matriks perbandingan berpasangan yaitu dengan menggunakan bilangan untuk merepresentasikan kepentingan relatif dari satu elemen terhadap elemen lainnya yang dimaksud dalam bentuk skala 1 sampai dengan 9. Skala ini mendefinisikan dan menjelaskan nilai 1 sampai 9 untuk pertimbangan dalam perbandingan berpasangan elemen pada setiap level hirarki terhadap suatu kriteria di level yang lebih tinggi. Berikut tabel skala kuantitatif nilai 1 – 9 untuk menilai tingkat suatu elemen dengan elemen lainnya.

Tabel 2. 3 Skala Penilaian Perbandingan Berpasangan

Tingkat Kepentingan	Definisi	Keterangan
1	Sama pentingnya	Kedua elemen mempunyai pengaruh yang sama
3	Agak lebih penting yang satu atas lainnya	Pengalaman dan penilaian sangat memihak satu elemen dibandingkan dengan pasangannya
5	Cukup penting	Pengalaman dan keputusan menunjukkan kesukaan atas satu aktifitas lebih dari yang lain
7	Sangat Penting	Pengalaman dan keputusan menunjukkan kesukaan yang kuat atas satu aktifitas lebih dari yang lain

9	Mutlak	Satu elemen mutlak lebih disukai dibandingkan dengan pasangannya, pada tingkat keyakinan tinggi
2,4,6,8	Nilai tengah diantara dua nilai keputusan yang berdekatan	Bila kompromi dibutuhkan

c. Sintesis

Pertimbangan-pertimbangan terhadap perbandingan di sintesis untuk memperoleh keseluruhan prioritas.

- 1) Menjumlahkan nilai-nilai dari setiap kolom pada matriks.
- 2) Membagi setiap nilai dari kolom dengan total kolom yang bersangkutan untuk memperoleh normalisasi matriks.
- 3) Menjumlahkan nilai dari setiap matriks dan membaginya dengan jumlah elemen untuk mendapatkan nilai rata-rata.
- 4) Mengukur konsistensi

Konsistensi penting untuk mendapatkan hasil yang valid dalam dunai nyata. AHP mengukur konsistensi pertimbangan dengan rasio konsistensi (*consistency ratio*). Nilai konsistensi rasio harus kurang dari 5% untuk matrik 3x3, 9% untuk matrik 4x4 dan 10% untuk matrik yang lebih besar. Jika lebih dari rasio dari batas tersebut, maka nilai perbandingan matrik dilakukan kembali.

Langkah-langkah menghitung nilai rasio konsistensi yaitu :

- a) Mengkalikan nilai pada kolom pertama dengan prioritas relatif elemen pertama, nilai pada kolom kedua prioritas relatif elemen kedua, dan seterusnya

- b) Menjumlahkan setiap baris
- c) Hasil dari penjumlahan baris dibagikan dengan elemen prioritas relatif yang bersangkutan
- d) Membagi hasil diatas dengan banyak elemen yang ada, hasilnya disebut eigen value (λ_{max})
- e) Menghitung indeks konsistensi (*consistency index*) dengan rumus :

$$CI = (\lambda_{max} - n) / n$$

Dimana CI : Consistensi Index

λ_{max} : Eigen Value

n : Banyak Elemen

- f) Menghitung konsistensi ratio (CR) dengan rumus :

$$CR = CI / RC$$

Dimana CR : *Consistency Ratio*

CI : *Consistency Index*

RC : *Random Consistency*

Matrik random dengan skala penilaian 1 samapai 9 beserta kebalikannya sebagai *random consistency*(RC). Berdasarkan perhitungan *saatym* menggunakan 500 sampel, jika pertimbangan memilih secara acak dari skala 1/9, 1/8, ..., 1, 2, ..., 9 akan diperoleh rata-rata konsistensi untuk matriks yang berbeda seperti tabel dibawah ini.

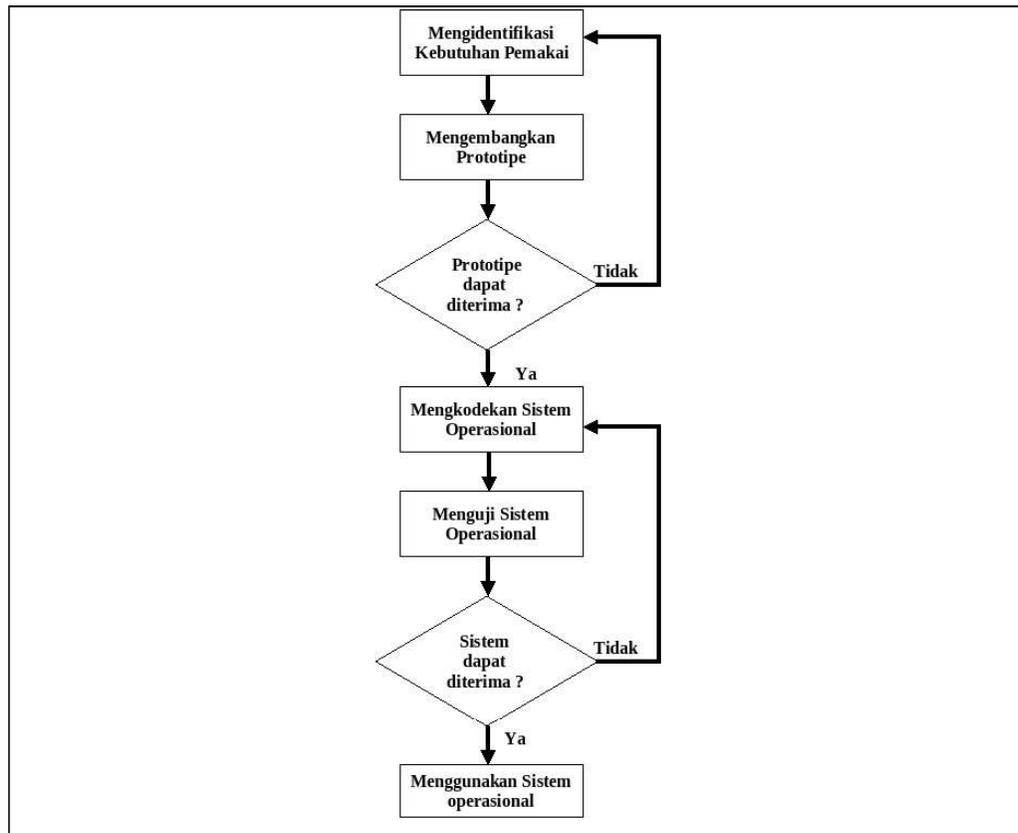
Tabel 2. 4 Nilai Rata – Rata Konsistensi

Ukuran Matriks	Konsistensi Acak
1	0.00

Ukuran Matriks	Konsistensi Acak
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

2.5 Metode Prototyping

Metode Prototyping adalah merupakan salah satu metode pengembangan sistem yang memungkinkan adanya interaksi antara pengembang dengan pelanggan selama proses pembuatan sistem. Berikut gambar metode prototyping [6] :



Gambar 2. 2 Metode Prototyping [6]

Tahapan – tahapan dalam pengembangan sistem prototyping adalah sebagai berikut [7] :

1. Mengidentifikasi Kebutuhan Pemakai

Tahap ini adalah tahapan pertama dalam metode prototyping. Pada tahap ini dilakukan studi kelayakan dan studi terhadap kebutuhan pemakai sistem, baik itu meliputi interface, teknik prosedural maupun teknologi yang akan digunakan.

2. Mengembangkan Prototype (Quick Design)

Pada tahap ini, dilakukan pengembangan prototype, yaitu pembuatan desain sistem yang akan digunakan untuk perancangan software.

3. Menentukan Penerima Prototype

Pada Tahap ketiga, yaitu mendeteksi serta mengidentifikasi sejauh mana pemodelan sistem yang dibuat dapat diterima, perbaikan dan perubahan sesuai dengan yang diinginkan pemesan. Bisa juga sampai merombak secara keseluruhan.

4. Mengadakan Sistem Operasional Melalui Program Sistem

Pada tahapan ini, pembuatan program dilakukan sesuai dengan pemodelan sistem yang telah disepakati.

5. Menguji Sistem Operasional

Setelah sistem aplikasi telah dibuat, dilakukan pengujian pada sistem tersebut baik menggunakan data primer maupun data sekunder. Ini dilakukan agar sistem dapat berjalan dengan baik sesuai yang diinginkan.

6. Menentukan Sistem Operasional

Tahap penentuan, apakah sistem aplikasi yang telah dibangun dapat diterima atau harus dilakukan perbaikan, atau bahkan harus dirombak dari awal lagi.

7. Implementasi Sistem

Tahapan terakhir ini dilakukan jika sistem aplikasi disetujui.

2.6 Analisa Sistem

2.6.1 Pengertian Analisa Sistem

Analisa sistem adalah sebuah penguraian dari suatu sistem informasi yang utuh kedalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan, kesempatan, hambatan yang terjadi dan kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan [8].

2.6.2 Langkah Analisa Sistem

Dalam analisis sistem terdapa empat langkah, yaitu [8]:

1. Mengidentifikasi Masalah (*Identify*)

Langkah pertama yang harus dilakukan oleh analisa sistem adalah dengan mengidentifikasi masalah yang ada. Adapun tugas-tugas yang harus dilakukan adalah sebagai berikut :

- a. Mengidentifikasi penyebab masalah
 - b. Mengidentifikasi titik keputusan
 - c. Mengidentifikasi personil-personil kunci
2. Memahami Kerja Sistem (*Understand*)

Tugas yang harus diperhatikan dalam memahami dari kerja sistem adalah :

- a. Menentukan jenis penelitian
 - b. Menentukan jadwal penelitian
 - c. Membuat penugasan penelitian
 - d. Membuat agenda wawancara
 - e. Mengumpulkan agenda wawancara
3. Menganalisa Hasil Penelitian

Untuk menganalisa hasil dari penelitian juga dapat menggunakan daftar pertanyaan.

4. Membuat Laporan Hasil Analisis

Laporan juga perlu dibuat karena untuk dokumentasi dari penelitian yang dilakukan. Tujuan utamanya adalah sebagai bukti secara tertulis tentang hasil analisa yang telah dilakukan.

2.7 *Unified Modelling Language (UML)*

2.7.1 Definisi UML

Beberapa pendapat para ahli tentang UML, diantaranya adalah :

UML adalah bahasa pemodelan standart yang memiliki sintak dan semantik. [8]

UML adalah bahasa pemodelan perangkat lunak yang berorientasi objek. [7]

Maka, UML dapat diartikan sebagai suatu bahasa pemodelan standart yang digunakan untuk membangun perangkat lunak berorientasi objek.

2.7.2 Langkah – Langkah Penggunaan UML

langkah – langkah penggunaan UML adalah sebagai berikut [7]:

1. Membuat daftar business proses dari level yang tertinggi untuk mendefinisikan aktifitas dan proses yang nanti akan muncul.
2. Petakan Use Case Diagram pada setiap business process untuk mendefinisikan dengan tepat fungsi yang harus disediakan oleh sistem, lengkapi use case diagram dengan *requirement*, *constrains*, dan *catatan – catatan* lain.
3. Definisikan requirement lain non fungsional security yang harus disediakan sistem.
4. Dari *use case diagram*, mulailah dengan *activity diagram*.
5. Definisikan obyek – obyek package atau domain level atas , jika kemungkinan error, maka buat lagi satu diagram untuk masing – masing alur.
6. Buat rancangan *user interface model* yang menyediakan antarmuka bagi pengguna untuk menjalankan scenario diagram.
7. Berdasarkan model – model yang telah ada, buatlah class diagram. Setiap package dipecah menjadi hierarki class lengkap dengan attribut dan metodenya.
8. Setelah *class diagram* dibuat, kita dapat melihat kemungkinan pengelompokkan class menjadi komponen – komponen. Kemudian buatlah *component diagram*.
9. Kemudian perhalus dengan *depelopment diagram*. Detailkan dengan dengan perangkat lunak, sistem operasi, dan jaringan. Kemudian petakan ke dalam node.
10. Setelah itu, mulailah membangun sistem. Ada dua pendekatan yang tepat untuk digunakan, yaitu :
 - a. Pendekatan use case dengan mengassign setiap use case kepada tim pengembang tertentu untuk mengembangkan unit kode yang lengkap dengan test.

- b. Pendekatan komponen, mengaassign setiap komponen kepada tim pengembang tertentu.

2.7.3 Bangun Dasar UML

Bangunan dasar metodologi UML menggunakan dua bangunan dasar untuk mendeskripsikan sistem/perangkat lunak yang akan dikembangkan, yaitu [8] :

1. Sesuatu (*things*)

Ada 4 (empat) things dalam UML, yaitu:

a. *Structural Things*

Merupakan bagian yang relatif statis dalam model Unified Modeling Language (UML). Bagian yang relatif statis dapat berupa elemen-elemen yang bersifat fisik maupun konseptual.

b. *Behavioral Things*

Merupakan bagian yang dinamis pada model Unified Modeling Language (UML), biasanya merupakan kata kerja dari model Unified Modeling Language (UML), yang mencerminkan perilaku sepanjang ruang dan waktu.

c. *Grouping Things*

Merupakan bagian pengorganisasi dalam Unified Modeling Language (UML). Dalam penggambaran model yang rumit kadang diperlukan penggambaran paket yang menyederhanakan model. Paket-paket ini kemudian dapat didekomposisi lebih lanjut. Paket berguna bagi pengelompokkan sesuatu, misalnya model-model dan subsistem-subsistem.

d. *Annotational Things*

Merupakan bagian yang memperjelas model Unified Modeling Language (UML) dan dapat berupa komentar-komentar yang menjelaskan fungsi

serta ciri-ciri setiap elemen dalam model Unified Modeling Language (UML).

2. Relasi (*Relationship*)

Ada 4 (empat) macam relationship dalam Unified Modeling Language (UML), yaitu:

a. Ketergantungan (*Dependention*).

Merupakan hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (*independent*) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri(*dependent*).

b. Asosiasi (*Association*)

Merupakan apa yang menghubungkan antara objek satu dengan objek lainnya, bagaimana hubungan suatu objek dengan objek lainnya. Suatu bentuk asosiasi adalah agregasi yang menampilkan hubungan suatu objek dengan bagian-bagiannya.

c. Generalisasi (*Generalization*)

Merupakan hubungan dimana objek anak (*descendent*) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (*ancestor*).Arah dari atas ke bawah dari objek induk ke objek anak dinamakan spesialisasi, sedangkan arah berlawanan sebaliknya dari arah bawah ke atas dinamakan generalisasi.

d. Realisasi (*Realization*)

Merupakan operasi yang benar-benar dilakukan oleh suatu objek.

2.7.4 Diagram – Diagram UML

1. Use Case Diagram

Merupakan alat komunikasi untuk mewakili persyaratan sistem. Diagram ini menunjukkan interaksi antara pengguna dengan entitas eksternal lainnya pada sistem yang sedang dikembangkan.

2. Activity Diagram

Menangkap alur sebuah sistem. Diagram ini digunakan untuk mendokumentasikan proses bisnis.

3. Class Diagram

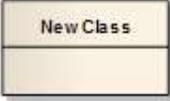
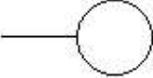
Diagram yang menggambarkan struktur kelas yang statis dalam sistem. Menggabungkan atribut, operasi dan hubungan antar class.

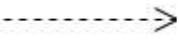
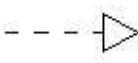
4. Sequence Diagram

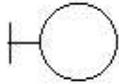
Diagram yang menunjukkan sejumlah objek dan pesan yang melalui objek.

2.7.5 Simbol – Simbol UML

Tabel 2. 5 Simbol – Simbol UML [9]

No.	Nama	Gambar	Fungsi
1.	<i>Actor</i>		Menggambarkan peran dari pemakai.
2.	<i>Use case</i>		Menunjukkan sekumpulan aktor dan use case dan hubungan diantara keduanya.
3.	<i>Class</i>		Mengilustrasikan sekumpulan kelas, paket dan hubungan yang merinci satu aspek tertentu dari sistem.
4.	<i>Interface</i>		Kumpulan operasi yang menspesifikasikan layanan dari kelas.

5.	<i>Interaction</i>		Menunjukkan aliran pesan atau informasi antar obyek ataupun hubungan antar obyek
6.	<i>Note</i>		Menyajikan komentar atau keterangan dari suatu elemen agar dapat langsung terlampir dalam model.
7.	<i>Dependency</i>		Relasi yang menunjukkan perubahan pada satu elemen dan memberikan pengaruh pada elemen yang lain.
8.	<i>Asociation</i>		Menyajikan gambaran navigasi antar data
9.	<i>Generalization</i>		Menyajikan hubungan antar elemen yang lebih umum ke elemen yang lebih spesifik.
10.	<i>Realization</i>		Hubungan bahwa elemen yang ada di bagian tanpa panah akan me-realisasikan apa yang dinyatakan oleh elemen. yang ada di bagian dengan panah.

11.	<i>Boundary Class</i>		Kelas yang memodelkan interaksi antara satu atau lebih <i>actor</i> dengan sistem.
12.	<i>Control Class</i>		Kelas yang mengkoordinasikan aktivitas dalam sistem.
13.	<i>Entity Class</i>		Kelas yang menyimpan dan mengolah data.
14	<i>Lifeline Class</i>		Kelas yang menjelaskan entitas tunggal.

2.8 Konsep Dasar Database

2.8.1 Definisi Database

Sebagai kesatuan istilah, Basis Data (*Database*) dapat didefinisikan dalam dalam sejumlah sudut pandang, seperti [10] :

1. Himpunan kelompok data (arsip) yang saling berhubungan, yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.
2. Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redundansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan.
3. Kumpulan file/tabel/arsip yang saling berhubungan yang disimpan dalam media elektronik.

Sedangkan menurut Kustiyarningsih (2011), “*Database* merupakan struktur penyimpanan data, digunakan untuk menambah, mengakses, dan memproses data

yang disimpan dalam sebuah database komputer. Dibutuhkan sebuah data base management system seperti sql server.”. [11]

Dari pengertian yang telah diuraikan oleh para ahli, dapat disimpulkan bahwa “Database adalah sekumpulan tabel – tabel berisi data, merupakan kumpulan dari field struktur file yang dapat digunakan untuk menambah data, mengakses data, dan memproses data yang disimpan dalam database komputer”.

2.8.2 MySql

Menurut Kadir (2009), “MySQL merupakan software open source yang dapat digunakan untuk membuat sebuah database”. [1]

Sedangkan menurut Raharjo (2011), “MySQL adalah RDBMS atau server data database yang dapat mengelola database dengan cepat dan dapat menampung data dalam jumlah yang besar”. [13]

Dari berbagai pendapat beberapa ahli dapat disimpulkan bahwa “ MySQL adalah software open source yang digunakan untuk membuat database “.

2.9 Definisi Microsoft Visual Basic 6.0

Microsoft Visual Basic 6.0 adalah salah satu *development tools* untuk membangun aplikasi dalam lingkungan Windows. Dalam pengembangan aplikasi, Visual Basic menggunakan pendekatan visual untuk merancang *user interface* dalam bentuk *form*, sedangkan untuk kodenya menggunakan dialek Bahasa Basic yang cenderung lebih mudah untuk dipelajari. Visual Basic telah menjadi tools yang terkenal untuk para pemula ataupun para *developer* [14].

Di dalam lingkungan Windows *user interface* juga sangat memegang peranan penting, karena dalam pemakaian aplikasi yang telah dibuat pemakai seperti berinteraksi dengan *user interface* tanpa menyadari bahwa di akhir akan ada intruksi-intruksi program yang mendukung tampilan dan proses yang sedang dilakukan [14].

Dalam pemrograman Visual Basic pengembangan aplikasi biasanya dimulai dengan membentuk *user interface*, kemudian mengatur properti dari objek-objek

yang telah digunakan dalam *user interface* dan baru dilakukan penulisan kode program untuk menangani kejadian-kejadian (*event*). Tahap pengembangan aplikasi dikenal dengan istilah pengembangan aplikasi dengan pendekatan *Button Up*.

2.10 Pengujian Blackbox

Black Box Testing memungkinkan kondisi masukan yang sepenuhnya akan melaksanakan semua persyaratan. Black Box Testing adalah pendekatan komplementer yang mungkin untuk mengungkap kelas yang berada dari kesalahan. Black Box menemukan kesalahan dalam kategori [15]:

1. Fungsi tidak benar atau hilang.
2. Kesalahan interface.
3. Kesalahan kinerja.
4. Kesalahan inisialisasi dan terminasi.