

BAB II

LANDASAN TEORI

2.1 Pengertian Sistem

Sistem adalah kumpulan atau sekumpulan elemen atau variabel yang saling berhubungan, berinteraksi, dan bergantung untuk mencapai suatu tujuan. Selain itu, sistem dapat didefinisikan sebagai kumpulan objek yang saling berhubungan dan berinteraksi, dan hubungan antar objek dapat dianggap sebagai satu kesatuan yang dirancang untuk mencapai tujuan tertentu[3]

Sistem juga merupakan kumpulan elemen-elemen yang saling terkait dan bekerja sama untuk memproses (*input*) yang ditujukan kepada sistem tersebut dan mengelola masukan tersebut sampai menghasilkan keluaran (*output*) yang diinginkan. Secara sederhana sistem dapat diartikan sebagai kumpulan yang terorganisir, saling berinteraksi, saling bergantung, terintegrasi elemen, komponen, atau variabel. Tidak hanya penampilan, tetapi juga hal-hal abstrak atau konseptual seperti misi, pekerjaan, kegiatan, kelompok informasi, dan lain sebagainya. Secara umum, elemen-elemen yang membentuk suatu sistem senantiasa tidak terlepas dari lingkungan sekitarnya. Maka umpan balik selain berasal dari *output*, juga dapat berasal dari lingkungan sistem tersebut. Organisasi dipandang sebagai suatu sistem yang juga memiliki sama unsur ini[4].

2.1.1 Karakter Sistem

Sistem akan berjalan dengan baik apabila memiliki karakteristik dalam pelaksanaannya[5]. Karakteristik sistemnya adalah sebagai berikut :

1. **Komponen Sistem**

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang bekerjasama membentuk suatu kesatuan.

2. **Batasan sistem**

ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem lainnya atau sistem dengan lingkungan lainnya. Lingkungan luar sistem (*environment*).

Bentuk apapun yang ada diluar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut dengan lingkungan luar sistem.

3. Penghubung sistem (*Interface*)

Media yang menghubungkan sistem dengan subsistem yang lain disebut dengan penghubung sistem atau *interface*.

4. Masukan sistem (*Input*)

Energi yang dimasukkan ke dalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*).

5. Keluaran sistem (*Output*)

Hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna.

6. Pengolahan sistem (*Processing*)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran.

7. Sasaran Sistem (*Objective*)

Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat deterministik.

2.1.2 Sistem Informasi

Sistem Informasi adalah data yang telah dikelola dan diproses untuk memberikan arti memperbaiki proses pengambilan keputusan[6]. Informasi adalah suatu data ataupun fakta yang telah diolah terlebih dahulu agar menjadi suatu bentuk yang berguna bagi penerimanya dan juga memiliki nilai dalam proses pengambilan keputusan, baik untuk saat ini ataupun di masa yang akan datang. Berdasarkan pernyataan di diatas, dapat disimpulkan bahwa informasi adalah data atau fakta yang sudah diolah yang memiliki makna atau arti bagi penggunanya dalam pengambilan keputusan.

2.1.3 Konsep Dasar Sistem Informasi

Sistem merupakan suatu jaringan kerja yang saling berhubungan dan berkumpul bersama – sama untuk mencapai suatu jaringan tertentu. Sedangkan informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya. Oleh karena itu, sistem informasi dapat didefinisikan sebagai kumpulan elemen yang saling berhubungan satu sama lain yang membentuk satu kesatuan untuk mengintegrasikan data memproses dan menyimpan serta mendistribusikan informasi[7].

2.1.4 Kualitas Informasi

Kualitas informasi merupakan kualitas *output* yang berupa informasi yang dihasilkan oleh sistem informasi yang digunakan. Pengguna sistem informasi tentunya berharap bahwa dengan menggunakan sistem tersebut mereka akan memperoleh informasi yang mereka butuhkan[8]. Karakteristik informasi yang dihasilkan suatu sistem informasi tertentu, dapat saja berbeda dengan informasi dari sistem informasi yang lain. Sistem informasi yang mampu menghasilkan informasi yang tepat waktu, akurat, sesuai kebutuhan, dan relevan serta memenuhi kriteria dan ukuran lain tentang kualitas informasi, berpengaruh terhadap kepuasan pemakainya.

2.2 Perancangan

Perancangan adalah sebuah proses untuk mendefinisikan sesuatu yang akan dikerjakan dengan menggunakan teknik yang bervariasi serta didalamnya melibatkan deskripsi mengenai arsitektur serta detail komponen dan juga keterbatasan yang akan dialami dalam proses pengerjaannya atau suatu pola yang dibuat untuk mengatasi masalah yang dihadapi perusahaan atau organisasi setelah melakukan analisis terlebih dahulu pada sistem yang berjalan atau merancang sistem yang baru untuk menemukan permasalahan yang dialami oleh perusahaan atau organisasi tersebut[9].

2.3 Pengertian Aplikasi *Desktop*

Aplikasi *desktop* adalah aplikasi yang berjalan lokal dalam lingkungan *desktop* dan hanya dapat diakses oleh pengguna *desktop*. Aplikasi *desktop* berbeda dengan aplikasi web yang dapat diakses dari manapun melalui jaringan[10]. Aplikasi adalah sebuah program komputer yang digunakan untuk tugas yang spesifik atau khusus, seperti akuntansi, analisis dalam bidang sains atau medis, atau pemrosesan kata. Bagian “aplikasi” mengacu kepada keseluruhan set dari program yang secara kolektif mengimplementasikan bagian dari proses bisnis ini dikenal sebagai “program aplikasi” atau “perangkat lunak aplikasi atau sebuah penerapan, menyimpan suatu data, permasalahan pekerjaan ke dalam suatu sarana atau media yang digunakan untuk menerapkan atau mengimplementasikan data atau permasalahan tersebut sehingga berubah menjadi suatu bentuk yang baru tanpa menghilangkan nilai-nilai dasar dari hal, data, permasalahan atau pekerjaan.

2.3.1 Persediaan

Persediaan adalah aktiva yang ditunjukkan untuk dijual atau proses lebih lanjut untuk menjadi barang jadi dan kemudian dijual sebagai kegiatan perusahaan. Persediaan merupakan barang dagangan yang disimpan kemudian dijual kembali dalam operasional normal perusahaan dan bahan yang terdapat dalam proses produksi atau yang telah disimpan untuk tujuan[11].

2.3.2 Pengertian Barang

Barang adalah Produk yang berbentuk fisik, dapat disentuh, dipindahkan, dirasa, dipegang, dan diperlukan fisik lainnya. Produk dalam wujud barang lebih cenderung bersifat memanjakan mata dan menarik perhatian para pembeli karena dinilai dari segi keindahan bentuk tersebut[12].

2.4 *Database*

Database sebagai kumpulan terorganisasi dari data-data yang berhubungan sedemikian rupa sehingga mudah disimpan, dimanipulasi serta dipanggil oleh pengguna. Terminologi hubungan berarti data mendeskripsikan domain (ranah)

tertentu sehingga pengguna mudah untuk mendapatkan jawaban atas pertanyaan yang diajukan ke basis data tersebut[13]. Sedangkan pengertian sistem basis data adalah sebagai koleksi dari data-data yang terorganisasi sedemikian rupa sehingga data mudah disimpan dan dimanipulasi (diperbarui, dicari, diolah dengan perhitungan-perhitungan tertentu, serta dihapus). *Database* atau basis data adalah kumpulan informasi yang disimpan didalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut. Kegunaan utama sistem basis data adalah agar pemakai mampu menyusun suatu pandangan (*view*) abstraksi data. Hal ini bertujuan untuk menyederhanakan interaksi antara pengguna dengan sistemnya dan basis data dapat mempresentasikan pandangan yang berbeda kepada para pengguna, programmer, dan administratornya.

Konsep dasar dari basis data adalah kumpulan dari catatan-catatan, atau potongan dari pengetahuan. Sebuah basis data memiliki penjelasan terstruktur dari jenis fakta yang tersimpan di dalamnya, penjelasan ini disebut skema. Skema menggambarkan objek yang diwakili suatu basis data, dan hubungan di antara objek tersebut. Ada banyak cara untuk mengorganisasikan skema, atau memodelkan struktur basis data, ini dikenal sebagai model basis data atau model data. Model yang umum digunakan sekarang adalah model relasional, yang menurut istilah lain mewakili semua informasi dalam bentuk tabel-tabel yang saling berhubungan di mana setiap tabel terdiri dari baris dan kolom (definisi yang sebenarnya menggunakan terminologi matematika). Dalam model ini, hubungan antar tabel diwakili dengan menggunakan nilai yang sama antar tabel. Model yang lain seperti model hierarkis dan model jaringan menggunakan cara yang lebih eksplisit untuk mewakili hubungan antar tabel.

2.5 Data

Data merupakan suatu sumber yang sangat berguna bagi hampir disemua organisasi[14]. Dengan tersedianya data yang melimpah, maka masalah pengaturan data secara efektif menjadi suatu hal yang sangat penting dalam

pengembangan sistem informasi manajemen. Oleh karena itu, tujuan dari diadakannya pengaturan data adalah sebagai berikut:

- a. Menyediakan penyimpanan data untuk dapat digunakan oleh organisasi saat sekarang dan kedepan
- b. Sebagai cara pemasukan data sehingga sehingga memudahkan tugas operator dan menyangkut pada waktu yang diperlukan oleh pemakai untuk mendapatkan data serta hak-hak yang dimiliki terhadap data yang ditangani
- c. Pengendalian data untuk setiap siklus agar data selalu *up to date* dan dapat mencerminkan perubahan spesifik yang terjadi di setiap sistem.
- d. Pengamanan data terhadap kemungkinan penambahan, modifikasi, pencurian, dan gangguan-gangguan lain.

2.6 Perangkat Lunak Pendukung Aplikasi

Dalam perancangan pada program aplikasi, untuk laporan skripsi ini, penulis menggunakan beberapa perangkat lunak yang menjadi penunjang pembuatan program aplikasi berbasis *desktop*, yaitu *windows* sebagai sistem operasi yang akan digunakan, *visual basic* sebagai bahasa pemrograman, *mysql* sebagai *database*[15].

2.6.1 Visual Basic

Visual Basic adalah program untuk membuat aplikasi berbasis *microsoft windows* secara cepat dan mudah. *Visual Basic* menyediakan *tool* untuk membuat aplikasi yang sederhana sampai aplikasi kompleks atau rumit baik untuk perusahaan atau instansi dengan sistem yang lebih besar[16].

Sejarah *Visual basic* diawali dari pengembangan bahasa *BASIC (Beginners All Purpose Symbolic Instruction Code)* di *Dartmount College*, Amerika Serikat pada awal tahun 1960. Sejak semula *basic* dirancang untuk mudah dipelajari. Pada awal tahun 1975, muncul komputer pribadi (*Personal Computer/PC*) pertama, bermerek MITS Altair. Namun, karena hanya memiliki RAM 4KB, satu-satunya bahasa yang digunakan untuk pemrograman adalah *Assembly*, bahasa ini sangat

sulit untuk dipahami karena perintahnya mendekati bahasa mesin. Tentu hal ini sangat menghambat perkembangan komputer pribadi. Bill Gate dan Paul Allen melihat potensi yang sangat besar pada komputer pribadi dimasa depan. Keduanya menciptakan *basic* untuk altair. Ternyata, sambutan masyarakat sangat baik dan akhirnya keduanya mendirikan perusahaan sendiri yang bergerak dibidang *software* yang bernama *Microsoft*. Pada tahun 1982 IBM/PC diperkenalkan kepada masyarakat, *microsoft* pun membuat sistem operasi MS-DOS untuk komputer ini. Dimana di dalamnya disertakan pula bahasa *basic* yang dikenal sebagai Quick Basic (QBASIC). Zaman pun berganti, pada tahun 1990 era DOS berlalu dan digantikan era *windows*.

Visual studio 2010 pada dasarnya adalah sebuah bahasa pemrograman komputer. Dimana pengertian dari bahasa pemrograman itu adalah perintah-perintah atau instruksi yang dimengerti oleh komputer untuk melakukan tugas-tugas tertentu[17]. *Visual studio* 2010 yang sering juga disebut sebagai sarana *tool* untuk menghasilkan program-program aplikasi berbasis *Windows*. Beberapa kemampuan atau manfaat dari *Visual Studio* 2010 diantaranya seperti:

1. Untuk membuat program aplikasi berbasis *Windows*
2. Untuk membuat objek-objek pembantu program seperti: kontrol *ActiveX*, *File*, *Help*, aplikasi internet dan sebagainya.

Menguji program *debugging* dan menghasilkan program berakhiran *exe* yang bersifat *executable* atau dapat langsung dijalankan

2.6.3 MySQL

MySQL merupakan sistem manajemen *database* yang bersifat *relational*. Artinya, data yang dikelola dalam *database* yang akan diletakkan pada beberapa tabel terpisah sehingga manipulasi data akan jauh lebih cepat[18]. *MSQL* dapat digunakan untuk mengelola *database* mulai dari kecil sampai dengan yang sangat besar. Pada kemajuannya, *MySQL* disebut juga *SQL* yang menjadi pendekatan dari *structured query language*. *SQL* merupakan bahasa tersistematis yang mengutamakan untuk mengolah *database*. *SQL* pertama kali dideskripsikan oleh

American National Standards Institute (ANSI) Pada tahun 1986. *MYSQL* adalah sebuah koordinasi manajemen *database* yang bersifat *open source*.

Jadi *SQL* adalah permintaan yang melekat pada suatu *database* atau *smbb* tertentu. Dengan pengertian lain, *SQL* adalah perintah atau bahasa yang melekat dalam *smbb*. Sebagai suatu bahasa permintaan, *SQL* didukung oleh *smbb*, seperti *MySQL Server*, *PostgresSQL*, *Interbase*, dan *Oracle*. Selain itu *SQL* juga didukung oleh *database* bukan server, seperti *ms access* maupun *paradox*.

2.6.3 *Crystal Report*

Crystal report adalah merupakan perangkat lunak yang dikhususkan untuk membangun sebuah laporan[19]. *Crystal report* dapat digunakan dengan bahasa pemrograman berbasis *windows* seperti *borland delphi*, *visual basic 6.0*, *visual basic.net*, *visual c++*, dan *visual interdev*. beberapa kelebihan dari *crystal report* ini adalah:

1. Dari segi pembuatan laporan tidak terlalu rumit yang memungkinkan para programmer pemula sekalipun dapat membuat laporan yang sederhana tanpa melibatkan banyak kode program.
2. Integrasi dengan bahasa-bahasa pemrograman lain yang memungkinkan dapat digunakan oleh banyak programmer dengan masing-masing keahlian.
3. Fasilitas *Import* hasil laporan yang mendukung format-format populer seperti *microsoft word*, *excel*, *access*, *adobe acrobat reader*, *html*, dan sebagainya.

2.6.2 Konseptual

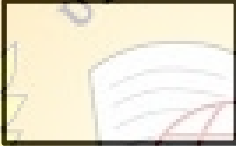


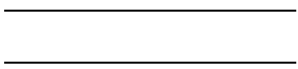
Dalam perancangan sistem ini penulis mencoba untuk menerapkan konsep perubahan data menjadi informasi melalui sistem yang akan dirancang[20]. Adapun yang menjadi inti dalam hal ini adalah prosedur persediaan barang, data barang, data barang masuk, data barang keluar dan stok barang yang merupakan bukti otentik untuk digunakan dalam melakukan *entry* data (pemasukan data). Sedangkan yang melakukan fungsi pemrosesan data adalah sistem yang dirancang

oleh penulis sehingga menjadi informasi yang diperlukan dalam proses persediaan barang.

2.7 Data Flow Diagram (DFD)

Data flow diagram (DFD) adalah suatu bagan yang menggambarkan arus data dalam suatu perusahaan, yang digambarkan dengan sejumlah simbol tertentu untuk menunjukkan perpindahan data yang terjadi dalam proses suatu sistem[21]. Simbol atau lambang digunakan dalam membuat *data flow diagram*, dapat dilihat pada tabel 2.1

Tabel 2. 1 *Data Flow Diagram*

	Simbol	Keterangan
		<i>External Entity</i> Simbol ini digunakan untuk menggambarkan asal atau tujuan data.
		Proses Simbol ini digunakan untuk proses pengolahan atau transformasi data.
		<i>Data Flow</i> Simbol ini digunakan untuk menggambarkan aliran data yang berjalan.
		<i>Data Store</i> Simbol ini digunakan untuk menggambarkan <i>data flow</i> yang sudah disimpan atau diarsipkan.

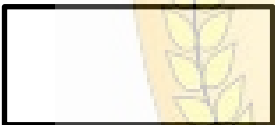
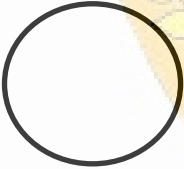
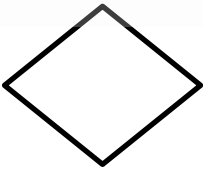
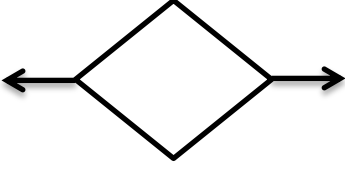
2.7.1 Entity Relational Diagram (ERD)

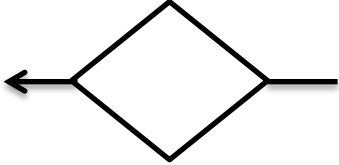
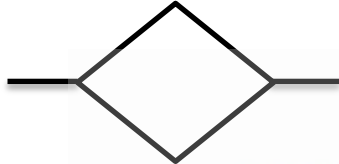
Entity relational diagram (erd) merupakan diagram yang menggambarkan *relationship* antar entitas yang relevan dari *system interest*. Entitas dapat di

gambarkan sebagai sesuatu yang menyebabkan seseorang mengumpulkan data untuk diproses menjadi informasi pendukung kegiatan bisnis. Pada aplikasi *traditional view driven*, konsep model data hanya berfokus pada pemilihan struktur data dari satu atau beberapa cara pandang[22]. Dalam model data “*event-driven*” digunakan notasi yang sama dengan model data “*viewdriven*”, tetapi fokus dan sasaran modeling data berbeda. Model data *event-driven* fokus pada kegiatan bisnis untuk seluruh proses bisnis, bukan hanya melihat fungsi-fungsi organisasi. Atau pengertian sempitnya adalah sebuah konsep yang mendeskripsikan hubungan antara penyimpanan *database* dan didasarkan pada persepsi dari sebuah dunia nyata yang terdiri dari sekumpulan objek yaitu disebut sebagai *entity* dan hubungan atau relasi antar objek-objek tersebut.

Komponen- komponen pembentuk ERD dapat dilihat pada tabel dibawah ini.

Tabel 2. 2 *Entity Relational Diagram*

Notasi	Komponen	Keterangan
	Entitas	Individu yang mewakili suatu objek dan dapat dibedakan dengan objek lain
	Atribut	Properti yang dimiliki oleh suatu entitas, dimana dapat mendeskripsikan karakteristik dari entitas tersebut
	Relasi	Menunjukkan hubungan diantara sejumlah entitas yang berbeda
	Relasi 1 : 1	Relasi yang ditunjukkan bahwa setiap entitas pada himpunan entitas pertama berhubungan dengan paling banyak entitas pada himpunan entitas kedua




	<p>Relasi 1 : N</p>	<p>Relasi yang menunjukkan bahwa relasi hubungan antara entitas pertama dengan kedua adalah satu banding banyak atau sebaliknya. Setiap entitas dapat berelasi dengan banyak entitas pada himpunan entitas yang lain</p>
	<p>Relasi N : N</p>	<p>Hubungsn ini menunjukkan bahwa setiap entitas pada himpunan entitas yang pertama dapat berhungan dengan banyak entitas yang kedua, demikian juga sebaliknya.</p>

2.7.2 Flowchart

Flowchart adalah cara penyusunan algoritma dengan memakai catatan grafis. *flowchart* merupakan gambar atau bagan yang memperlihatkan urutan atau langkah-langkah dari suatu program dan hubungan antar proses bersama. Gambaran ini dinyatakan dengan simbol[23]. Dengan begitu setiap simbol mendeskripsikan proses tertentu. Sedangkan antara proses digambarkan dengan garis penghubung. Dengan menggunakan *flowchart* akan memudahkan dalam melakukan pengecekan bagian-bagian tertentu. Adapun simbol dan kegunaan *flowchart* dapat dilihat pada Tabel 2.3 di bawah ini.

Tabel 2. 3 *Flowchart*

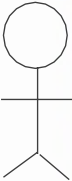



No	Simbol	Fungsi
1		Terminal, untuk memulai atau mengakhiri suatu program
2		Proses, suatu simbol yang menunjukkan setiap pengolahan yang dilakukan oleh komputer.
3		<i>Input-Output</i> , untuk memasukkan data ataupun menunjukkan hasil dari suatu proses.
4		<i>Decision</i> , suatu kondisi yang akan menghasilkan beberapa kemungkinan jawaban atau pilihan.
5		<i>Predefine Process</i> , suatu simbol untuk menyediakan tempat-tempat pengolahan dalam <i>storage</i> .
6		<i>Connector</i> , suatu prosedur akan masuk atau keluar melalui simbol ini dalam lembar yang sama.
7		<i>Off-Line Connector</i> , merupakan simbol masuk atau keluarnya suatu prosedur pada lembar kertas lainnya.
8		Arus / <i>Flow</i> dari prosedur yang dapat dilakukan dari atas ke bawah, dari bawah ke atas, dari kiri ke kanan, ataupun dari kanan ke kiri.
9		<i>Document</i> , merupakan simbol untuk data yang berbentuk kertas maupun untuk informasi.
10		Untuk menyatakan sekumpulan langkah proses yang ditulis sebagai prosedur.





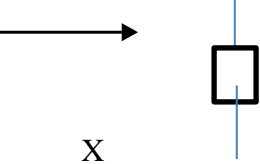
11		Simbol untuk <i>output</i> , yang ditunjukkan ke suatu <i>device</i> , seperti <i>printer</i> , <i>plotters</i> dan lain-lain sebagainya.
12		Menunjukkan <i>input</i> atau <i>output</i> menggunakan <i>harddisk</i>
13		Menunjukkan <i>input</i> atau <i>output</i> menggunakan disket

2.7.3 Sequency Diagram

Sequency diagram “menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek[24].

Tabel 2. 4 *Sequency Diagram*

Simbol	Nama	Deskripsi
	Aktor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda diawal frase nama aktor.
	<i>Lifeline</i>	Menyatakan kehidupan suatu objek
	<i>Obyek</i>	Menyatakan objek yang berinteraksi pesan
	Waktu aktif	Menyatakan objek dalam keadaan aktif dan berinteraksi,

		semuanya yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya.
<< create >> 	Pesan tipe <i>create</i>	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat
1: nama_metode() 	Pesan tipe <i>call</i>	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri.
1: masukan 	Pesan tipe <i>send</i>	Menyatakan bahwa suatu objek mengirimkan data / masukan / informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.
1: keluaran 	Pesan tipe <i>return</i>	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian
<< destroy >> 	Pesan tipe <i>destroy</i>	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaliknya jika ada <i>create</i> maka ada <i>destroy</i>

2.8 Pengujian *White Box*

White-box Testing adalah pengujian yang menggunakan *cyclomatic complexity*. salah satu cara untuk menguji dengan melihat kode program yang ada dengan melihat modul dan menganalisis kesalahannya[25]. Jika *output* yang dihasilkan oleh modul tidak sesuai dengan proses yang benar. Maka barisan program, variabel, dan parameter yang termasuk dalam unit tersebut akan diperiksa atau dikoreksi secara individual dan kemudian dikompilasi ulang.

2.9 Pengujian *Black Box*

Black-Box adalah sebuah metode yang dipakai untuk menguji sebuah *software* tanpa harus memerlukan detail *software*. Pengujian ini hanya memeriksa nilai keluaran berdasarkan nilai masing – masing. Tidak ada upaya untuk mengetahui kode program apa yang *output* pakai. Proses *Black-Box Testing* dengan cara mencoba program yang telah dibuat dengan mencoba memasukkan data pada setiap halamannya. Pengujian ini diperlukan untuk mengetahui program tersebut berjalan sesuai dengan yang dibutuhkan oleh perusahaan[26].

