BAB II
TINJAUAN PUSTAKA

2.1 Tinjauan Instansi

2.1.1 Sejarah Instansi

Rahastra Credit Union (Rahastra CU) merupakan salah satu koperasi kredit di Indonesia yang mengedepankan pendidikan ekonomi anggota secara terpadu. Keterpaduan tersebut adalah dalam pemahaman ekonomi rumah tangga anggota, pelayanan koperasi, kewirausahaan, Pendidikan berjenjang,akutansi keuangan usaha dan visi jauh ke depan.

Rahastra CU merupakan koperasi yang keanggotaannya terdiri dari berbagai kalangan dan profesi, diantaranya: pedagang, pengrajin, karyawan, guru, dosen, bidan, dokter, dan para pengusaha kecil serta menengah dalam berbagai jenis bidang usaha. Bermula dari prakarsa pimpinan dan karyawan PT.C59 pada tanggal 5 Mei 1995 yang kemudian mendapatkan pengesahan Badan Hukum dari Pemerintah Nomor: 518/BH.30-DISKOP/2001, akhirnya RAHASTRA CU dapat mengembangkan potensi, pelayanan dan permodalannya. Bahkan, sejak tahun 2005 keanggotaannya sangat terbuka untuk umum karena mengedepankan prinsip *Open membership and voluntary* 'keanggotaan yang terbuka dan atas dasar sukarela'.

Saat ini Rahastra Credit Union telah memiliki 3 cabang yang tersebar di beberapa lokasi di daerah Bandung. Jumlah anggotanya saat in lebih dari 3000 orang. Berikut adalah daftar cabang Rahastra Credit Union:

1. Jl. Cikutra Ruko Delima 87 Bandung.
2. Perum Surya Rahayu Blok A/7/17 Sri Rahayu Kec. Cikancung Kab. Bandung.
3. Jl. Ma Eja No. 20 A Cibaduyut Bandung.

2.1.2 Logo Instansi

Logo Koperasi Kredit adalah sebuah keluarga yang sejahtera, susuai dengan tujuan koperasi yang meningkatkan standar kesejahteraan anggotanya. Logo Sikopdit ditunjukan pada gambar 2.1.



Gambar 2. 1 Logo Koperasi Kredit

2.1.3 Maskot Instansi

Maskot Koperasi Rahastar CU seperti yang ditunjukan pada gambar 2.2, adalah kupu-kupu. Ini berarti Rahastra CU telah melakukan metamorfosa dalam kurun waktu 10 tahun untuk menjadi koperasi kredit yang professional.

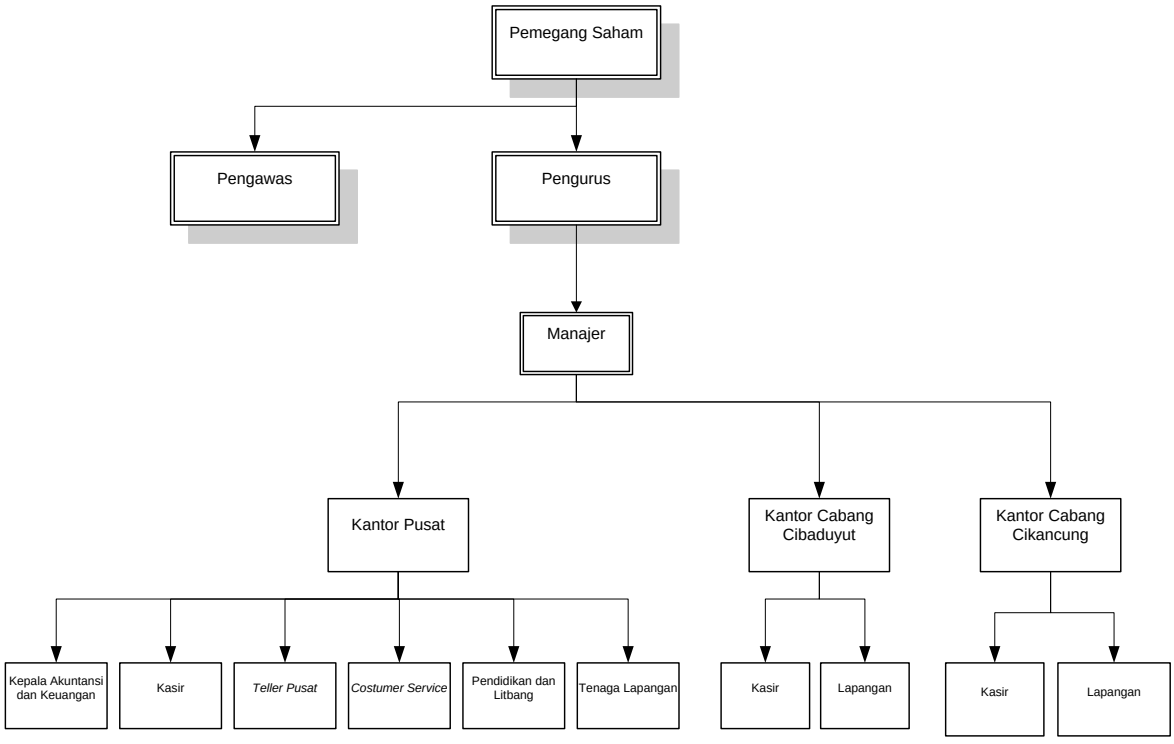


Gambar 2. 2 Maskot Rahastra CU

2.1.4 Badan Hukum Instansi

Rahastar Credit Union merupakan koperasi yang berbadan hukum dengan Nomor: 518/BH.30-DISKOP/2001.

2.1.5 Struktur Organisasi dan *Job Description*



Gambar 2. 3 Struktut Organisasi

Job Description:

1. Pemegang saham merupakan anggota dari Rahastra CU. Pemagang saham akan mengadakan RAT. Rapat Anggota Tahunan (RAT) merupakan kegiatan rutin yang diadakan setahun sekali oleh anggota koperasi untuk menentukan arah dan kebijakan selanjutnya. Dalam RAT akan ditentukan susunan pengurus dan pengawas baru serta pembagian Sisa Hasil Usaha (SHU) kepada anggota.
2. Pengurus akan membuat kebijakan dan peraturan yang akan diterapkan di koperasi untuk selanjutkan akan digunakan sebagai *SOP (Standard Operating Procedure)*. Karena pengurus bukanlah karyawan koperasi dan mempunyai tugas lain, maka diberikan penugasan kepada managemen koperasi untuk mengelola dan melaksanakan kebijakan yang dibuat oleh pengurus. Karena itu pengurus akan mengontrol pelaksanaan kegiatan operasional oleh managemen.
3. Pengawas bertugas mengontrol kebijakan yang dibuat oleh pengurus sesuai dengan UU Koperasi dan AD/ART. Walaupun ada kesalahan yang dibuat oleh managemen dalam pengelolaan, akan tetapi pengawas hanya akan meminta pertanggungjawaban dari pengurus.
4. Managemen adalah staf yang diberi tugas oleh pengurus untuk melaksanakan kegiatan rutin dan pengelolaan di koperasi. Kegiatan itu antara lain adalah penambahan anggota baru, penambahan saldo tabungan, peminjaman kredit dan lain sebagainya.

5. Manajer bertanggung jawab kepada pengurus koperasi atas pengelolaan koperasi yang dilakukannya. Selain itu manajer adalah pemimpin dari staf managemen, jadi ia akan mengontrol semua tugas yang dilakukan oleh staf managemen.
6. Staf akan membantu manager sesuai dengan kemapuan dan keahlian mereka yang telah ditentukan tugas dan fungsinya, baik itu di kantor pusat maupun cabang.
7. Kepala Akuntansi dan Keuangan bertugas me-*review* semua proses transaksi yang ada di koperasi.
8. Kasir bertugas melayani nasabah di kasir, seperti penyetoran uang dan pengambilan uang.
9. *Customer Service* bertugas melayani masalah dan keluhan dari pelanggan. Misalnya, tatarcara pengajuan kredit atau pendaftaran menjadi anggota baru.
10. Pendidikan dan Litbang merupakan divisi khusus yang bertujuan untuk mengembangkan koperasi, seperti menambah jumlah nasabah/anggota, mengadakan promosi-promosi maupun yang lainnya.
11. Tenaga Lapangan bertugas sebagai pembantu umum.
12. Sedangkan staf di kantor cabang akan bertugas sesuai dengan pos nya masing-masing dengan bertanggung jawab langsung kepada manajer koperasi di kantor pusat.

2.1.6 Produk Rahastra CU

2.1.6.1 Takara

Takara atau Tabungan Berjangka Rahastra ialah suatu produk simpanan dengan jangka wangktu tertentu. Disetor setiap bulan sesuai dengan tabel2.1 yang ada dengan jangka waktu 6 bulan sampai dengan 36 bulan, dimana nilai nominalnya dari Rp 500.000,- sampai dengan Rp 15.000.000,- serta jasa yang diberikan 10% per tahun.

Tabel 2.1 Takara

Nominal	Jangka Waktu				
	6 Bulan	12 Bulan	16 Bulan	24 Bulan	36 Bulan
500.000	81.650	39.800	25.900	18.950	12.100
1.000.000	163.300	79.600	51.750	37.750	24.150
1.500.000	247.500	119.450	77.650	56.750	36.250
2.000.000	326.500	159.200	103.500	75.700	48.300
2.500.000	408.150	199.050	129.350	94.600	60.350
3.000.000	489.750	238.800	155.250	113.500	72.450
3.500.000	571.350	278.600	181.500	132.400	84.500
4.000.000	653.000	318.400	207.000	151.350.	96.600
4.500.000	734.600	358.200	232.850	1750.250	108.650
5.000.000	816.250	398.000	258.700	189.750	120.750
5.500.000	897.850	437.850	284.600	208.050	132.800
6.000.000	979.500	477.600	310.550	227.000	144.850
6.500.000	1.061.100	517.400	336.350	245.900	156.950
7.000.000	1.142.750	57.200	362.200	264.800	169.000
7.500.000	1.224.350	597.000	388.050	283.700	181.100
8.000.000	1.306.000	636.800	413.950	302.650	193.150

8.500.000	1.387.600	676.600	439.800	321.550	205.150
9.000.000	1.469.200	716.400	465.650	340.450	217.300
9.500.000	1.550.800	756.200	491.550	359.400	229.350
10.000.000	1.632.450	796.000	517.400	378.300	241.450
10.500.000	1.714.050	835.800	543.300	397.200	253.500
11.000.000	1.795.700	875.600	569.150	416.100	265.550
11.500.000	1.877.300	915.400	595.000	435.050	277.650
12.000.000	1.959.000	955.000	620.900	453.950	289.700
12.500.000	2.040.000	995.000	646.750	472.850	301.800
13.000.000	2.122.150	1.034.800	672.650	491.750	313.850
13.500.000	2.203.800	1.074.600	698.500	510.700	325.900
14.000.000	2.285.500	1.114.400	724.350	529.600	338.850
14.500.000	3.367.050	1.154.200	750.250	548.500	350.050
15.000.000	2.448.650	1.194.000	776.100	567.4000	362.150

2.1.6.2 Tahakhu

Tahakhu atau Tabungan Harian Khusus merupakan produk simpanan, dimana anggota menyetorkan dengan nilai awal setoran sebesar Rp 50.000,- dengan jasa yang diberikan sebesar 8,5% per tahun dan berhak mengikuti undian berhdiah selama enam bulan sekali.

2.1.6.3 Simpanan Saham

Adapun prosedur simpanan saham sebagai berikut:

1. Simpanan Pokok, disetor 1 kali saat masuk menjadi anggota sebesar Rp 100.000,- dan dapat dicicil sampai 4 kali.

2. Simpanan Wajib. Disetor anggota setiap bulan sebesar Rp 20.000,- s.d. Rp 150.000.
3. Simpanan Sukarela. Disetor anggota bebas kapan saja minimal setoran Rp 10.000,-.
4. *Deviden* akan diberikan kepada anggota setiap tahun.

2.1.6.4 Simapan

Simapan atau Simpanan Masa Depan merupakan produk simpanan berjangka waktu 5.s.d. 25 tahun. Disetor setiap bulan dengan setoran Rp 25.000,- s.d. Rp 100.000,-. Jasa simpanan yang diberikan sebesar 10% per tahun sesuai tabel 2.2.

Tabel 2. 2 Tahaku

Setoran Rp 25.000/Bulan			
Waktu	Seroran	Jasa Simpanan	Perolehan
5 Tahun	1.500.000	450.000	1.950.000
10 Tahun	3.000.000	2.160.000	5.160.000
15 Tahun	4.500.000	5.940.000	10.440.000
20 Tahun	6.000.000	13.140.000	19.140.000
25 Tahun	7.500.000	25.940.000	33.440.000
Setoran Rp 50.000/Bulan			
Waktu	Seroran	Jasa Simpanan	Perolehan
5 Tahun	3.000.000	900.000	3.900.000
10 Tahun	6.000.000	4.300.000	10.300.000
15 Tahun	9.000.000	11.800.000	20.800.000
20 Tahun	12.000.000	26.200.000	38.200.000

25 Tahun	15.000.000	51.800.000	66.800.000
Setoran Rp 75.000/Bulan			
Waktu	Seroran	Jasa Simpanan	Perolehan
5 Tahun	4.500.000	1.300.000	5.800.000
10 Tahun	9.000.000	6.400.000	15.400.000
15 Tahun	13.500.000	17.800.000	31.300.000
20 Tahun	18.000.000	39.400.000	57.400.000
25 Tahun	22.500.000	77.800.000	100.300.000
Setoran Rp 100.000/Bulan			
Waktu	Seroran	Jasa Simpanan	Perolehan
5 Tahun	6.000.000	1.800.000	7.800.000
10 Tahun	12.000.000	8.600.000	20.600.000
15 Tahun	18.000.000	23.800.000	41.800.000
20 Tahun	24.000.000	52.500.000	76.500.000
25 Tahun	30.000.000	103.700.000	133.700.000

;

2.1.6.5 Simpandik

Simpandik atau Simpanan Pendidikan Anak merupakan produk simpanan bagi anak-anak sekolah untuk persiapan biaya pendidikan. Simapann ini setorannya tidak ditentukan, kapan saja. Setoran minimal Rp 5.000. Setiap peserta Simpandik akan diikutkan dalam kegiatan bersama yang bersifat pendidikan.

2.1.6.6 Tabungan Rumahku

Tabungan Rumahku merupakan produk KPR. Disetor setiap bulan sesuai tabel. Jangka waktu yang diberikan 11 tahun dengan jasa KPR 8,9% per tahun dan 15 tahun dengan jasa KPR 9,4% per tahun. Jasa simpanan yang diberikan untuk 1-1,5 tahun pertama adalah 10% per tahun dan pada saat KPR sebesar 7,4% per tahun.

Tabel 2. 3 Tabungan Rumahku

Setoran Per Bulan	Nilai Akhir Tabungan	Nilai KPR	Santunan Pinjaman
250000	8.700.000	16.000.000	Saldo
300000	10.400.000	20.000.000	Saldo
350000	12.100.000	23.300.000	Saldo
400000	13.900.000	25.700.000	Saldo
450000	15.600.000	30.000.000	Saldo
500000	17.400.000	33.300.000	Saldo
550000	19.100.000	36.700.000	Saldo
600000	20.800.000	40.000.000	Saldo
650000	22.600.000	43.300.000	Saldo
700000	24.300.000	46.700.000	Saldo
750000	26.100.000	50.000.000	Saldo
800000	27.800.000	53.300.000	Maks 50 Juta
850000	29.600.000	56.700.000	Maks 50 Juta
900000	31.300.000	60.000.000	Maks 50 Juta
950000	33.000.000	63.300.000	Maks 50 Juta
1000000	34.000.800	66.700.000	Maks 50 Juta
1.500.000	52.200.000	100.000.000	Maks 50 Juta
2.000.000	69.600.000	133.300.000	Maks 50 Juta
Setoran Per Bulan	Nilai Akhir Tabungan	Nilai KPR	Santunan Pinjaman
201.500	8.600.000	15.000.000	Saldo
268.500	11.500.000	20.000.000	Saldo
335.500	14.400.000	25.000.000	Saldo
402.500	17.300.000	30.000.000	Saldo
469.500	20.200.000	35.000.000	Saldo
536.500	23.100.000	40.000.000	Saldo
603.500	26.000.000	45.000.000	Saldo
670.500	28.900.000	50.000.000	Saldo
737.000	31.800.000	55.000.000	Maks 50 Juta
804.500	34.700.000	60.000.000	Maks 50 Juta
871.500	37.600.000	65.000.000	Maks 50 Juta
938.500	40.400.000	70.000.000	Maks 50 Juta
1.005.500	43.300.000	75.000.000	Maks 50 Juta

1.072.500	46.200.000	80.000.000	Maks 50 Juta
1.139.500	49.100.000	85.000.000	Maks 50 Juta
1.206.500	52.000.000	90.000.000	Maks 50 Juta
1.273.500	54.900.000	95.000.000	Maks 50 Juta
1.341.000	57.800.000	100.000.000	Maks 50 Juta

2.1.6.7 Pinjaman

1. Pinjaman Produktif

   Berupa modal kerja atau investasi untuk pengembangan usaha dengan jasa sebesar 2,5% menurun atau 1,5% *flat* dimana jangka waktu maksimalnya adalah 36 bulan.

2. Pinjaman Konsumtif atau Kesejahteraan

   Pinjaman Konsumtif dengan jasa 2,5% menurun atau 1,5% *flat* dimana jangka waktu maksimalnya adalah 36 bulan.

3. Pinjaman Kesehatan

   Merupakan pinjaman untuk biaya kesehatan dengan maksimal pinjaman Rp 3.500.000. Jasa yang diberikan 0% untuk anggota yang bersangkutan, 0,5% untuk anak, 0,5% untuk suami/istri dan 1% untuk orang tua kandung.

2.1.6.8 Daperma

Daperma atau Dana Perlindungan Bersama merupakan produk *insurance* ‘perlindungan’ terhadap simpanan dan pinjaman anggota yang akan diterima oleh anggota Rahastra CU tanpa harus membayar lagi iuran

atau kewajiban premi sebab sudah ditanggung oleh Rahastra CU. Daperma tersebut berupa:

1. Santunan duka. Yaitu santunan tabungan 100% dari saldo akhir Simpanan Saham anggota maksimal sebesar Rp 25.000.000 batas umur 54 tahun.
2. Pembebasan saldo pinjaman anggota. Yaitu berupa santunan pinjaman bagi anggota yang meninggal dunia atau cacat. Maksimal sebesar Rp 50.000.000 untuk batas umur 54 tahun. Sehingga ahli waris atau keluarga anggota terbebas dari tanggungan hutang dan juga menerima tabungan serta santunan.

2.2 Landasan Teori

2.2.1 Sistem Informasi

2.2.1.1 Konsep Sistem

Sistem adalah sekumpulan unsur atau elemen yang saling berkaitan dan saling mempengaruhi dalam melakukan kegiatan bersama untuk mencapai tujuan.

Menurut L. Akcof “Sistem adalah setiap kesatuan konseptual atau fisik yang terdiri dari bagian-bagian dalam keadaan saling tergantung sata sama lainya”.

2.2.1.2 Karakteristik Sistem

1. Memilki komponen

   Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, bekerja sama membentuk satu kesatuan. Komponen-komponen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Setiap sistem tidak peduli betapapun kecilnya, selalu mengandung komponen-komponen atau subsistem-subsistem. Setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan. Suatu sistem dapat mempunyai suatu sistem yang lebih besar yang disebut supra sistem, misalnya suatu perusahaan dapat disebut dengan suatu sistem dan indutri yang merupakan sistem yang lebih besar dapat disebut dengan supra sistem. Kalau dipandang industri sebagai suatu sistem, maka perusahaan dapat dipandang sabagai suatu sistem, maka sistem akuntansi adalah subsistemnya.

2. Batas Sistem

   Boundary system ‘batas sistem’ merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainya atau merupakan dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan. Batas suatu sistem menunjukan *scope* ‘ruang lingkup’ dari sistem tersebut.

3. Lingkungan luar sistem

Adalah apapun diluar batas dari sistem yang mempengruhi operasi sistem.

4. *Interface*

   Interface ‘penghubung sistem’ merupakan media penghungung antara satu subsistem dengan subsistem yang lainnya.

5. *Input*

   Input ‘masukan sistem’ merupakan energi yang dimasukan kedalam sistem. Masukan dapat berupamaintenance input ‘masukan perawatan’ dan signal input ‘masukan sinyal’.

6. Keluaran sistem

   Merupakan *output* dari *input* yang diproses oleh sistem.

7. Pemrosesan sistem

   Merupakan bagian yang memproses masukan untuk menjadi keluaran yang diinginkan.

8. Sasaran sistem

   Kalau sistem tidak mempunyai sasaran, maka operasi sistem tidak ada gunanya.

Secara garis besari sistem dapat dibedakan menjadi 2, yaitu:

1. Sistem Fisik

   Sistem fisik atau *physical system* ialah kumpulan elemen-elemen atau unsur-unsur yang saling berinteraksi satu sama lain secara fisik serta dapat diidentifikasi secara nyata tujuan-tujuannya.

2. Sistem Abstrak

   Sistem abstrak atau *abstract system* ialah sistem yang dibentuk akibat terselenggaranya ketergantungan ide dan dapat diidentifikasikan secara nyata, tetapi dapat diuraikan elemen-elemennya.

2.2.1.1 Konsep Informasi

Informasi ialah data yang telah diproses menjadi bentuk yang memiliki arti bagi penerima dan dapat berupa fakta, suatu nilai yang bermanfaat. Jadi ada suatu proses transformasi data menjadi suatu informasi.

Data merupakan *raw material* untuk suatu informasi. Perbedaan informasi dan data sangat relatif tergantung pada nilai gunanya bagi manajemen yang memerlukan. Suatu informasi bagi level manajemen tertentu bisa menjadi data bagi manajemen level diatasnya, atau sebaliknya.

Kualitas informasi tergantung pada 3 hal, yaitu informasi harus:

1. Akurat. Berarti informasi harus bebeas dari kesalahan-kesalahan dan tidak bias atau menyesatkan. Akurat berarti juga informasi harus jelas mencerminkan maksudnya.
2. Tepat pada waktunya. Berarti informasi yang data pada penerima tidak boleh terlmbat.
3. Relevan. Berarti informasi tersebut mempunyai manfaat untuk pemakainya. Relevansi informasi untuk tiap-tiap orang satu dengan lainya mungkin berbeda.

2.2.1.2 Definisi Sistem Informasi

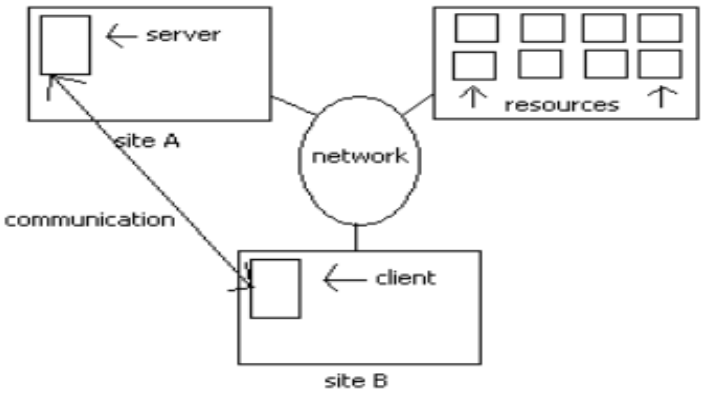
Sistem informasi ialah sebuah sistem yang terintregrasi atau sistem manusia-mesin, untuk menyediakan informasi untuk mendukung operasi, manajemen dalam suatu organisasi.

Sistem ini memanfaatkan perangkat keras dan perangkat lunak komputer, prosedur manual, model manajemen dan basis data.

Menurut Robert A. Leicth “Sistem infornasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manejerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan.

2.2.2 Sistem Terdistribusi

Sistem terdistribusi adalah sekumpulan prosesor yang tidak saling berbagi memori atau *clock* dan terhubung melalui jaringan komunikasi yang bervariasi, yaitu melalui *Local Area Network* ataupun melalui *Wide Area Network*.Prosesor dalam sistem terdistribusi bervariasi, dapat berupa *small microprocessor, workstation, minicomputer*, dan lain sebagainya. Berikut adalah ilustrasi struktur sistem terdistribusi :



Gambar 2. 4 Struktur Sistem Terdistribusi

Karakteristik sistem terdistribusi adalah sebagai berikut:

1. *Concurrency of Components*

Pengaksesan suatu komponen/sumber daya (segala hal yang dapat digunakan bersama dalam jaringan komputer, meliputi H/W dan S/W) secara bersamaan. Contoh: Beberapa pemakai *browser* mengakses halaman *web* secara bersamaan

2. *No Global Clock*

Hal ini menyebabkan kesulitan dalam mensinkronkan waktu seluruh komputer atau perangkat yang terlibat. Dapat berpengaruh pada pengiriman pesan/data, seperti saat beberapa proses berebut ingin masuk ke *critical session*.

3. *Independent Failures of Components*

Setiap komponen atau perangkat dapat mengalami kegagalan namun komponen atau perangkat lain tetap berjalan dengan baik.

Ada empat alasan utama untuk membangun sistem terdistribusi, yaitu:

1. *Resource Sharing*

   Dalam sistem terdistribusi, situs-situs yang berbeda saling terhubung satu sama lain melalui jaringan sehingga situs yang satu dapat mengakses dan menggunakan sumber daya yang terdapat dalam situs lain. Misalnya, *user* di situs A dapat menggunakan *laser printer* yang dimiliki situs B dan sebaliknya *user* di situs B dapat mengakses *file* yang terdapat di situs A.

2. *Computation Speedup*

   Apabila sebuah komputasi dapat dipartisi menjadi beberapa subkomputasi yang berjalan bersamaan, maka sistem terdistribusi akan mendistribusikan subkomputasi tersebut ke situs-situs dalam sistem. Dengan demikian, hal ini meningkatkan kecepatan komputasi (*computation speedup*).

3. *Reliability*

   Dalam sistem terdistribusi, apabila sebuah situs mengalami kegagalan, maka situs yang tersisa dapat melanjutkan operasi yang sedang berjalan.Hal ini menyebabkan reliabilitas sistem menjadi lebih baik.

4. *Communication*

   Ketika banyak situs saling terhubung melalui jaringan komunikasi, *user* dari situs-situs yang berbeda mempunyai kesempatan untuk dapat bertukar informasi.

Tantangan-tantangan yang harus dipenuhi oleh sebuah sistem terdistribusi:

1. Keheterogenan perangkat atau multiplisitas perangkat

   Suatu sistem terdistribusi dapat dibangun dari berbagai macam perangkat yang berbeda, baik sistem operasi, *hardware* maupun *software*.

2. Keterbukaan

   Setiap perangkat memiliki antarmuka (*interface*) yang di-publish ke komponen lain. Perlu integrasi berbagai komponen yang dibuat oleh *programmer* atau *vendor* yang berbeda

3. Keamanan

   Shared resources dan transmisi informasi atau data perlu dilengkapi dengan enkripsi.

4. Penangan kegagalan

   Setiap perangkat dapat mengalami kegagalan secara independen. Namun, perangkat lain harus tetap berjalan dengan baik.

5. *Concurrency of Components*

   Pengaksesan suatu komponen atau sumber daya secara bersamaan oleh banyak pengguna.

6. Transparansi

   Bagi pemakai, keberadaan berbagai perangkat dalam sistem terdistribusi tampak sebagai satu sistem saja.

2.2.3 *Database*

Basis data *(Database)* terdiri dari dua kata, yaitu basis dan data. Basis dapat diartikan sebagai markas atau gudang, tempat bersarang atau berkumpul. Sedangkan data adalah reprsentasi fakta dunia nyata yang mewakili suatu objek

seperti manusia, barang, hewan, peristiwa, konsep, keadaan dan sebagainya yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi atau kombinasinya (Fathansyah, 1999).

Database merupakan kupulan dari data-data yang saling berhubungan yang terorganisasi sedemikian rupa sehingga mudah digunakan kembali. *Database*merupakan salah satu bagian yang terpenting dalam sistem informasi, karena merupakan dasar dalam menyediakan informasi bagi para pemakai. Disamping berisi data, setiap *Database*juga mengandung definisi struktur (baik untuk *Database*maupun objek-objeknya) seperti *file*/tabel, indeks dan lain-lain secara detail. *Database* memberikan keuntungan pada data yang diolah, antara lain:

a. Kecepatan dan Kemudahan *(Speed)*

Pemanfaatan *databae* memungkinkan kita untuk dapat menyimpan data, melakukan perubahan atau manipulasi terhadap data, menampilkan kembali data tersebut dengan lebih cepat dan lebih mudah daripada menyimpan data secara manual.

b. Efisiensi Ruang Penyimpanan (*Space)*

Dengan *Database,* efisiensi atau optimalisasi penggunaan ruang penyimpanan dapat dilakukan, hal ini dikarenakan *Database*mampu melakukan penekanan jumlah redudansi data, baik dengan menerapkan sejumlah pengkodean atau dengan membuat relasi antar kelompok data yang saling berhubungan.

c. Keakuratan *(Accuracy)*

Pemanfaatan pengkodean atau pembentukan relasi bersama dengan penerapan autran atau batasan *(constraint)* tipe data, domain data, keunikan data dan sebagainya yang secara ketat dapat diterapkan dalam sebuah *Database*. Cara ini sangat berguna untuk menekan ketidakakuratan pemasukan atau penyimpanan data.

d. Ketersediaan *(Availibility)*

Pertumbuhan data baik dari segi jumlah maupun jenisnya sejalan dengan waktu akan semakin membutuhkan ruang penyimpanan yang besar. Akan tetapi tidak semua data itu akan selalu digunakan atau dibutuhkan. Dengan penggunaan *databse*, data yang sudah jarang atau tidak pernah digunakan, dapat diatur dengan dilepaskan dari *Database*yang sedang aktif. Disisi lain, karena kepentingan pemakaian data, sebuah *Database* dapat memilki data yang tersebar di banyak lokasi.

e. Kelengkapan *(Completeness)*

Dalam sebuah *databse,* disamping data kia juga harus menyimpan struktur (bauk yang mendefenisikan objek-objek dalam basisi data maupun definisi detail dari setiap objek, seperti struktur *file*, tabel atau indeks) dan juga dapat mengakomodasikan kebutuhan yang terus berkembang terhadap data, selain dapat menambah *record* namun juga dimungkinkan menambhakan *field-field* baru pada suatu tabel data.

f. Keamanan *(security)*

Dengan memanfaatkan keamnanan pada *Database* kita dapat menentukan siapa yang boleh menggunkan basis data beserta objek-objek didalamnya dan menetukan operasi apa saja yang boleh dilakukan.

g. Kebersamaan pemakaian *(sharability)*

Pemanfaatan *Database* dengan menggunakan aplikasi atau sistenm yang *multi user* akan memenuhi kebutuhan banyak pemakai, namun tetap dengan menjaga atau menghindari munculnya inkonsistensi dan kondisi *deadlock.*

2.2.3.1 Replikasi *Database*

Replikasi ialah proses peng-copy an *Database* sebuah server ke server lain. Bilamana *Database* original berubah, maka perubahan tersebut akan otomatis diaplikasikan pada copy an *Database* tersebut. Replikasi adalah suatu teknik untuk melakukan copy dan pendistribusian data dan objek-objek *Database* dari satu *Database* ke *Database* lain dan melaksanakan sinkronisasi antara *Database* sehingga konsistensi data dapat terjamin.

Dengan menggunakan teknik replikasi ini, data dapat didistribusikan ke lokasi yang berbeda melalui koneksi jaringan lokal maupun internet. Replikasi juga memungkinkan untuk mendukung kinerja aplikasi, penyebaran data fisik sesuai dengan penggunaannya, seperti pemrosesan transaksi online dan DSS (*Desiscion Support System*) atau pemrosessan *Database* terdistribusi melalui beberapa server.

Keuntungan replikasi tergantung dari jenis replikasi tetapi pada umumnya replikasi mendukung ketersediaan data setiap waktu dan dimanapun diperlukan.

Adapun keuntungan lainnya adalah :

1. Memungkinkan beberapa lokasi menyimpan data yang sama. Hal ini sangat berguna pada saat lokasi-lokasi tersebut membutuhkan data yang sama atau memerlukan server yang terpisah dalam pembuatan aplikasi laporan.
2. Aplikasi transaksi online terpisah dari aplikasi pembacaan seperti proses analisis *Database* secara online, data smarts atau data warehouse.
3. Memungkinkan otonomi yang besar. Pengguna dapat bekerja dengan meng-copy data pada saat tidak terkoneksi kemudian melakukan perubahan untuk dibuat *Database* baru pada saat terkoneksi
4. Data dapat ditampilkan seperti layaknya melihat data tersebut dengan menggunakan aplikasi berbasis Web
5. Meningkatkan kinerja pembacaan
6. Membawa data mendekati lokasi individu atau kelompok pengguna. Hal ini akan membantu mengurangi masalah karena modifikasi data dan pemrosesan query yang dilakukan oleh banyak pengguna karena data dapat didistribusikan melalui jaringan dan data dapat dibagi berdasarkan kebutuhan masing-masing unit atau pengguna.
7. Penggunaan replikasi sebagai bagian dari strategi *standby server*.

Replikasi dapat digunakan apabila sebuah organisasi atau perusahaan didukung oleh hardware dan aplikasi sofware dalam sebuah sistem yang terdistribusi. Aplikasi yang berbeda mempunyai kebutuhan yang berbeda untuk otonomi dan konsistensi data. Replikasi diperlukan dalam sistem terdistibusi apabila berikut ini:

1. Mengcopy dan mendistribusikan data dari satu atau lebih lokasi
2. Mendistribusikan hasil copy data berdasarkan jadwal
3. Mendistribusikan perubahan data ke server lain
4. Memungkinkan beberapa pengguna di beberapa lokasi untuk melakukan perubahan dan kemudian menggabungkan data yang telah dimodifikasi
5. Membangun aplikasi data yang menggunakan perlengkapan online maupun offline
6. Membangun aplikasi Web sehingga pengguna dapat melihat volume data yang besar.

2.2.3.2 Mode Replikasi *Synchronous*

Mode replikasi sinkron memungkinkan pertukaran data secara real-time sehingga kesinkronan suatu data akan terjaga, dimana saat ada transaksi operasional yang sedang menulis sesuatu ke disk sumber, maka saat yang bersamaan penulisan juga dilakukan terhadap disk target yang ada di lokasi remote. Keseluruhan proses penulisan pada disk sumber dan disk target harus selesai terlebih dahulu sebelum beranjak ke transaksi operasional selanjutnya dan

diberi acknowledge untuk keduanya jika telah selesai. Pada mode replikasi ini, kebutuhan akan performansi sistem yang tinggi harus dipertimbangkan. Selain itu jarak antara disk sumber dan disk target juga menjadi prasyarat utama, bahwa pihak yang terlibat dalam mode replikasi ini harus berjarak < 100km antara keduanya. Keuntungan dari mode replikasi ini adalah menyediakan recovery yang konsisten dan lengkap untuk semua jangka waktu.

2.2.3.3 Mode Replikasi Asynchronous

Mode replikasi asinkron memungkinkan pertukaran data secara buffering dalam artian bahwa data akan diletakkan dalam sebuah 'penampung sementara terlebih dahulu, kemudian pada jangka waktu tertentu akan direplikasi ke disk target. Data yang direplikasi ke disk target tidak membutuhkan acknowledgement agar penulisan transaksi operasional pada disk sumber dapat berlangsung kembali. Sehingga mode replikasi ini tidak menjamin kesinkronan suatu data pada dua pihak yang terlibat karena jika suatu saat terjadi crash pada salah satu pihak dan data belum sempat direplikasi maka data yang terdapat pada kedua pihak tidak bisa dikatakan sebagai sebuah data yang sinkron. Walaupun hal ini dapat meningkatkan performansi sistem, namun lebih memiliki banyak risiko. Jika hal ini terjadi maka recovery yang cukup rumit dilakukan (namun tidak menjamin data hasil recovery adalah data yang benar dan konsisten karena ada kemungkinan hilangnya beberapa data). Keuntungan dari mode replikasi ini adalah efektivitas biaya.

2.2.3.4 Basis Data Terdistribusi

Dalam sebuah *Database* terdistribusi, *Database* disimpan pada beberapakomputer. Komputer-komputer dalam sebuah sistem terdistribusi berhubungansatu sama lain melalui bermacam-macam media komunikasi seperti high-speedbuses atau telephone line.

Sebuah sistem *Database* terdistribusi berisikan sekumpulan site, di mana tiap-tiapsite dapat berpartisipasi dalam pengeksekusian transaksi-transaksi yangmengakses data pada satu site atau beberapa site. Tiap-tiap site dapatmemproses transaksi lokal yaitu sebuah transaksi yang mengakses data padasatu site di mana transaksi telah ditentukan.Sebuah site juga dapat mengambil bagian dalam mengeksekusi transaksi globalyaitu transaksi yang mengakses data pada site yang berbeda di mana transaksitelah ditentukan, atau transaksi yang mengakses data pada beberapa site yangberbeda.

Karakteristik basis data terdistribusi yaitu :

1. Kumpulan data yang digunakan bersama secara *logic* tersebar pada sejumlah komputer yang berbeda.
2. Komputer yang dihubungkan menggunakan jaringan komunikasi.
3. Data pada masing-masing situs dapat menangani aplikasi-aplikasi lokal secara otonom.
4. Data pada masing situs di bawah kendali satu*DBMS.*

Dalam sebuah *Database* terdistribusi, *Database* disimpan pada beberapa komputer. Komputer-komputer dalam sebuah sistem terdistribusi berhubungan

satu sama lain melalui bermacam-macam media komunikasi seperti *high-speed buses* atau *telephone line*.

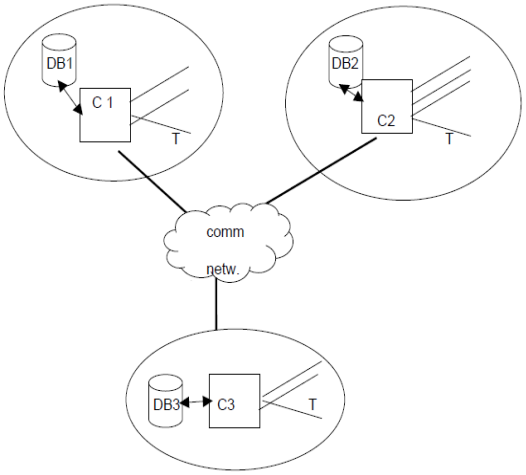
Sebuah sistem *Database* terdistribusi berisikan sekumpulan *site*, di mana tiap-tiap *site* dapat berpartisipasi dalam pengeksekusian transaksi-transaksi yang mengakses data pada satu *site* atau beberapa *site*.Tiap-tiap *site* dapat memproses transaksi lokal yaitu sebuah transaksi yang mengakses data pada satu *site* di mana transaksi telah ditentukan.

Sebuah *site* juga dapat mengambil bagian dalam mengeksekusi transaksi global yaitu transaksi yang mengakses data pada *site* yang berbeda di mana transaksi telah ditentukan, atau transaksi yang mengakses data pada beberapa *site* yang berbeda.

Ada 2 aspek penting dari *distributed Database (DDB)* :

1. Distribusi : data tidak disimpan pada tempat (prosesor) yang sama, sehingga *DDB* dapat dibedakan dari *Database* tunggal, sentralisasi

2. Korelasi logika : data memiliki properti yang berhubungan sehingga *DDB* dapat dibedakan dari sekumpulan *Database* lokal atau *file* yang disimpan pada tempat yang berbeda pada jaringan komputer.

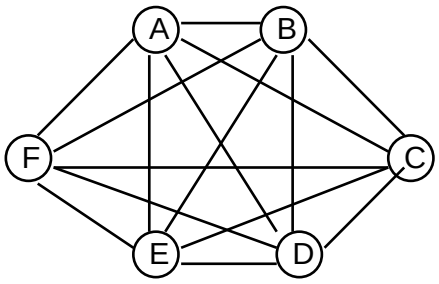


Gambar 2. 5*Database* Terdistribusi Secara Geografis

Site-site dalam *Database* terdistribusi dihubungkan secara fisik dengan berbagaicara. Beberapa topologi digambarkan sebagai sebuah graph yang simpulsimpulnyabersesuaian dengan site. Sebuah edge dari simpul A ke simpul Bbersesuaian dengan sebuah hubungan langsung antara dua site. Beberapakonfigurasi (bentuk) digambarkan sebagai berikut:

1. *Fully Connected Network*

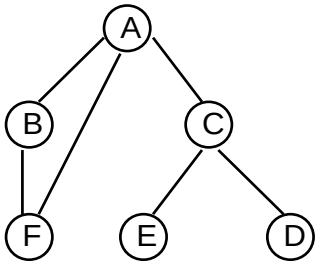
   Kalau salah satu node rusak, yang lainnya masih dapat berjalan (biaya mahal), kontrol manajemen tidak terjamin.



Gambar 2. 6 Fully *Connected Network*

2. *Partially Connected Network*

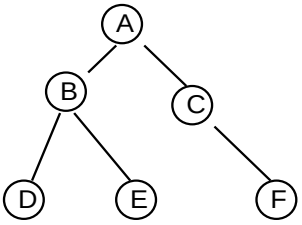
Reliability rendah, biaya dapat ditekan, kontrol manajemen tidak terjamin.



Gambar 2. 7*Partially Connected Network*

3. *Tree Structured Network*

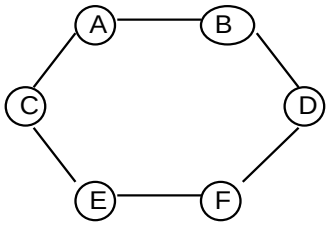
Bersifat sentral, control manajemen lebih terjamin. Kalau node pusat rusak, semua akan rusak. (setiap proses dimulai dari bawah).



Gambar 2. 8*Tree Structured Network*

4. *Ring Network*

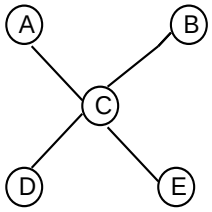
Rusak satu, yang lain masih berjalan. Kontrol manajemen kurang terjamin karena bersifat desentralisasi.



Gambar 2. 9*Ring Network*

5. *Star Network*

Kontrol manajemen lebih terjamin, karena bersifat sentral. Kalau pusat rusak yang lain rusak.



Gambar 2. 10*Star Network*

Keuntungan *Database*terdistribusi:

1. Otonomi lokal. Karena data didistribusikan, user dapat mengakses dan bekerjadengan data tersebut sehingga memiliki kontrol lokal.
2. Meningkatkan kinerja. Karena setiap site menangani hanya bagian dari *Database*,CPU dan I/ O tidak seberat seperti *Database* terpusat. Data yang dipakai untuktransaksi disimpan dalam beberapa site, sehingga eksekusi transaksi dapatsecara parallel.
3. Meningkatkan reliability/ availability. Jika satu site mengalami *crash*, dapatmembuat beberapa site tidak dapat diakses. Jika data

direplikasi ke banyaksite, kerusakan hubungan komunikasi tidak menjadikan sistem total tidak dapatdioperasikan.

4. Ekonomis. Biaya komunikasi, baik membagi aplikasi dan memprosessecara lokal di setiap site. Dari biaya komunikasi data, akan lebih murah untukmemelihara sistem komputer dalam satu site dan menyimpan data secaralokal.
5. Expandibility. Lebih mudah mengakomodasikan ukuran *Database* yang semakinbesar. Ekspansi dapat dilakukan dengan menambah proses dan kekuatanpenyimpanan ke jaringan.
6. Shareability. Jika sistem informasi tidak terdistribusi, akan sulit untuk berbagidata dan sumber daya. Sistem *Database* terdistribusi memungkinkan hal ini.

Kerugian *Database*terdistribusi :

1. Harga *software* mahal. Hal ini disebabkan sangat sulit untuk membuat sistem *Database* terdistribusi.
2. Kemungkinan kesalahan lebih besar. Site-site beroperasi secara paralel sehingga lebih sulit untuk menjamin kebenaran dan algoritma. Adanya kesalahan mungkin tak dapat diketahui.
3. Biaya pemrosesan tinggi. Perubahan pesan dan penambahan perhitungan dibutuhkan untuk mencapai koordinasi antar *site.*

2.2.4 Jaringan Komputer

Jaringan komputer adalah sebuah sistem yang terdiri atas komputer dan perangkat jaringan lainya yang terhubung melalui media yang bekerja bersama-sama untuk mencapai suatu tujuan yang sama.

Jaringan komputer muncul dari adanya kebutuhan untuk berbabagi data diantara pengguna. Komputer memiliki kemampuan dalam memprouksi beberapa jenis informasi yang berupa data, *spreadsheet* atau grafik. Tanpa jaringan komputer data tersebut dapat diberikan ke *user* lain melalui perangkat penyimpanan, seperti *flashdisj, disket, hardisk* dan lain sebagainya.

Tujuan dari jaringan komputer adalah:

1. Membagi sumber daya: contohnya berbagi pemakaian printer, *CPU*, memori, harddisk
2. Komunikasi: contohnya surat elektronik, *instant messaging*, *chatting*
3. Akses informasi: contohnya *web browsing*

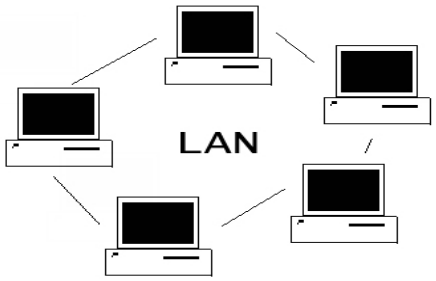
Agar dapat mencapai tujuan yang sama, setiap bagian dari jaringan komputer meminta dan memberikan layanan (*service*). Pihak yang meminta layanan disebut klien (*client*) dan yang memberikan layanan disebut pelayan (*server*).Ars*ite*ktur ini disebut dengan sistem *client-server*, dan digunakan pada hampir seluruh aplikasi jaringan komputer.

2.2.4.1 Jenis-jenis jaringan:

Berdasarkan lokasi geografis dan ukurannya, jaringan trebagi kedalam beberapa jenis, diantaranya:

1. *Local Area Network (LAN)*

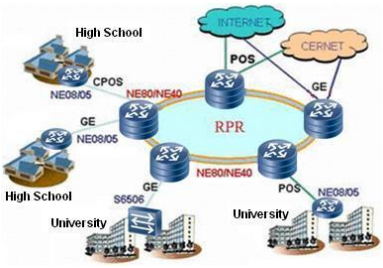
Suatu jaringan komputer yang menghubungkan suatu komputer dengan komputer lain dengan jarak yang terbatas.



Gambar 2. 11*Local Area Network*

2. *Metropolitant Area Network (MAN)*

Merupakan suatu jaringan yang cakupannya meliputi suatu kota. MAN mengnhubungkan LAN-LAN yang lokasiny berjauhan. Jangkauan MAN biasanya mencapai 10 km sampai beberapa ratus km. suatu MAN biasanya bekerja pada kecepatan 1,5 sampai 150 Mbps.

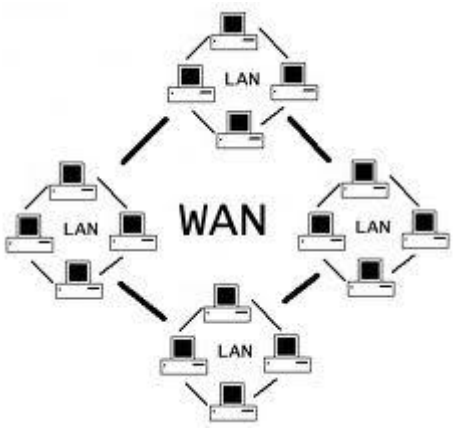


Gambar 2. 12*Metropolitan Area Network*

3. Wide Area Network (WAN)

WAN dirancang untuk menghubungkan komputer-komputer yang terletak pada suatu cakupan geografis yang luas seperti hubungan antara suatu Negara.

Cakupan WAN bias melliputi 100 km sampai 1000 km dan kecepatan antarkota bias bervariasi antara 1,5 Mbps sampai 2,4 Gbps. Dalam WAN biaya untuk transmisi sangat tinggi, dan biasanya dimiliki dan dioperasikan sebagai suatu jaringan public.



Gambar 2. 13*Wide Area Network*

4. Global Area Network (GAN)

GAN merupakan suatu jaringan yang menghubungkan Negara-negara di seluruh dunia. Kecepatan GAN bervariasi mulai dari 1,5 Mbps sampai dengan 100Gbps dan cakupannya mencakup ribuan kilometer. Contoh yang sangat baik dari GAN ini adalah *internet.*



Gambar 2. 14*Global Area Network*

Pada dasarnya setiap jaringan komputer ada yang berfungsi sebagai *client* dan juga *server*. Tetapi ada jaringan yang memiliki komputer yang khusus didedikasikan sebagai *server* sedangkan yang lain sebagai *client*. Ada juga yang tidak memiliki komputer yang khusus berfungsi sebagai *server* saja. Karena itu berdasarkan fungsinya maka ada dua jenis jaringan komputer:

1. *Client-server*

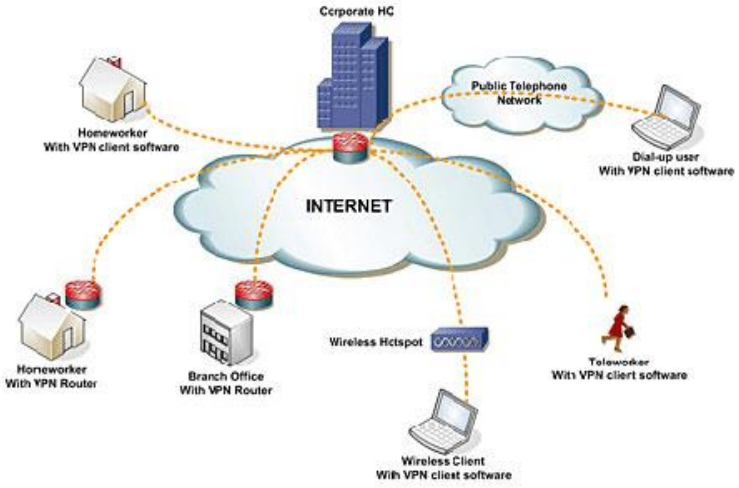
Yaitu jaringan komputer dengan komputer yang didedikasikan khusus sebagai *server*.Sebuah service/layanan bisa diberikan oleh sebuah komputer atau lebih.Contohnya adalah sebuah domain seperti www.detik.com yang dilayani oleh banyak komputer web *server*.Atau bisa juga banyak *service*/layanan yang diberikan oleh satu komputer. Contohnya adalah *server* jtk.polban.ac.id yang merupakan satu komputer dengan multi *service* yaitu *mail server, web server, fileserver, Databaseserver* dan lainnya.

2. *Peer-to-peer*

Yaitu jaringan komputer dimana setiap host dapat menjadi *server* dan juga menjadi *client* secara bersamaan. Contohnya *dalam file sharing* antar komputer di jaringan *Windows Network Neighbourhood* ada 5 komputer (kita beri nama A,B,C,D dan E) yang memberi hak akses terhadap *file* yang dimilikinya. Pada satu saat A mengakses *file* share dari B bernama data_nilai.xls dan juga memberi akses *file* soal_uas.doc kepada C. Saat A mengakses *file* dari B maka A berfungsi sebagai *client* dan saat A memberi akses *file* kepada C maka A berfungsi sebagai *server*. Kedua fungsi itu dilakukan oleh A secara bersamaan maka jaringan seperti ini dinamakan *peer to peer*.

3. Virtual Private Network (VPN)

VPN adalah singkatan dari *virtual private network*, yaitu jaringan pribadi (bukan untuk akses umum) yang menggunakan medium nonpribadi (misalnya internet) untuk menghubungkan antar *remote-site* secara aman. Perlu penerapan teknologi tertentu agar walaupun menggunakan medium yang umum, tetapi *traffic*(lalu lintas) antar *remote-site* tidak dapat disadap dengan mudah, juga tidak memungkinkan pihak lain untuk menyusupkan *traffic* yang tidak semestinya ke dalam *remote-site*.



Gambar 2. 15Virtual Private Network

VPN menyediakan koneksi *point-to-point* baik kepada kantor cabang maupun kepada seorang karyawan yang sedang bertugas ditempat lain. Menghubungkan antar kantor pusat/cabang dengan menggunakan VPN jauh lebih ekonomis dengan keamanan yang dapat diandalkan daripada menyewa jaringan khusus (*leased lines*) atau dengan panggilan jarak jauh melalui modem.

VPN dapat menjadi jaringan khusus yang besar dan tidak terbatas. Sebuah WAN khusus yang jauh lebih efisien, aman dan berbiaya ekonomis dari WAN atau LAN tradisional. Sehingga telah banyak perusahaan-perusahaan yang menggunakan VPN sebagai infrastruktur jaringanya yang menghubungkan antara kantor pusat dengan kantor cabang dan dengan agen serta client nya.

Tidak ada standar tertentu untuk VPN, namun secara umum dapat disebut bahwa VPN menggunakan jaringan internet umum untuk satu ataubeberapa keperluan dengan membentuk lorong khusus (jaringan khusus / tunnelling) secara virtual.

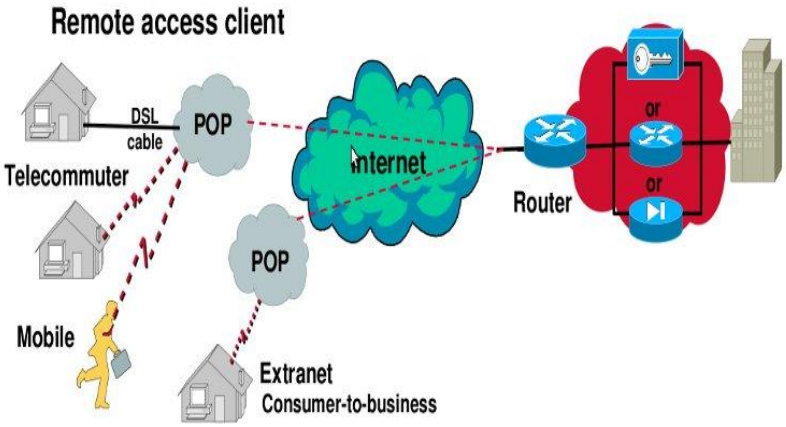
Dalam penggunaan sebagai jaringan khusus ini, VPN diset sedemikian rupa dengan sebuah *software* dan *hardware* dengan protokol tertentu yang akan digunakan untuk otentikasi antar *user* dan untuk penyandian jaringannya. Umumnya VPN dipasangi *firewall* di dekat *server*nya yang berfungsi untuk menyaring sehingga hanya *client* yang telah terdaftar saja yang dilayani.

VPN terbagi dalam 2 bagian yaitu bagian "dalam" yang diproteksi dengan sistem sandi tertentu dan bagian "luar" yang merupakan infrastruktur internet yang tidak diproteksi. Memproteksi data dengan penyandian selama perjalanan antar *user* dalam sebuah VPN telah sangat populer dan selalu digunakan.

Tipe VPN

1. *Remote Site VPN*

Remote-acces juga disebut sebagi *Virtual private dial-up network (VPDN)*, adalah sebuah koneksi (seperti LAN) yang digunakan oleh sebuah perusahaan yang mempunyai pegawai yang membutuhkan koneksi ke jaringan *private*(pribadi) dari beberapa lokasi yang jauh. Biasanya, sebuah perusahaan yang ingin mengembangkan remote-acces VPN akan menggunakan jasa *enterprise service provider (ESP)*. ESP akan menyiapkan sebuah *network access server (NAS)* dan akan menyiapkan sebuah *VPN client software* yang akan digunakan pada *dekstop user* (pengguna).Kemudian sebuah *dial number*digunakan untuk mengakses *NAS* dan kemudian menggunakan *VPN client software* untuk memasuki jaringan perusahaan.



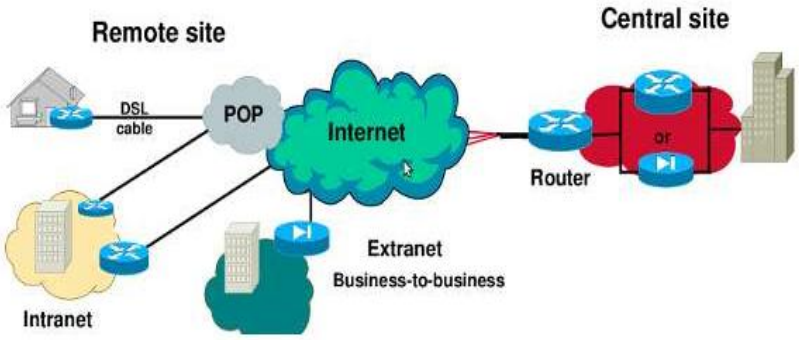
Gambar 2. 16*Remote to Site*

2. *Site to Site VPN*

Melalui sebuah penggunaan peralatan khusus dan metode-metode enkripsi, sebuah perusahaan dapat menghubungkan berbagai lokasi melalui sebuah publin network misalnya internet.

Site-to-site VPN terbagi menjadi 2 , yaitu:

1. *Intranet Based*, jika sebuah perusahaan memiliki 1 atau beberapa remote location (lokasi terpisah atau jauh) yang ingin bergabung dalam sebuah *single private network*, maka perusahaan tersebut dapat membuat sebuah VPN intranet untuk menghubungkan LAN ke LAN
2. Ekstranet based, Ketika sebuah perusahaan mempunyai sebuah kedekatan hubungan dengan sebuah perusahaan lain (misalnya *patner*, *supplier*, atau *customer*), mereka dapat membuat sebuah ektranet *VPN* yang dapat menghubungkan *LAN*-ke-*LAN*, dan mengizinkan beberapa perusahaan yang berbeda untuk bekerja dalam sebuah lingkungan yang sama dimana perusahaan-perusahaan tersebut dapat saling melakukan *sharing*.



Gambar 2. 17Site to Site

2.2.5 Pemograman

2.2.5.1 Pemograman Berorientasi Objek

Pemograman berorientasi objek *(Obhect oriented programming – OOP)* merupakan paradigma pemograman yang berorientsikan kepada objek. Semua data dan fungsi di dalam paradigma ini dibungkus dalam kelas-kelas atau objek-objek. Bandingkan dengan logika pemrograman terstruktur. Setiap objek dapat menerima pesan, memproses data, dan mengirim pesan ke objek lainnya,

Model data berorientasi objek dikatakan dapat memberi fleksibilitas yang lebih, kemudahan mengubah program, dan digunakan luas dalam teknik piranti lunak skala besar. Lebih jauh lagi, pendukung OOP mengklaim bahwa OOP lebih mudah dipelajari bagi pemula dibanding dengan pendekatan sebelumnya, dan pendekatan OOP lebih mudah dikembangkan dan dirawat.

A. Konsep pemograman berorientasi objek.

Pemrograman berorientasiobjek menekankan konsep berikut:

1. kelas, kumpulan atas definisi data dan fungsi-fungsi dalam suatu unit untuk suatu tujuan tertentu. Sebagai contoh *'class of dog'* adalah suatu unit yang terdiri atas definisi-definisi data dan fungsi-fungsi yang menunjuk pada berbagai macam perilaku/turunan dari anjing. Sebuah *class* adalah dasar dari modularitas dan struktur dalam pemrograman berorientasi oobjek. Sebuah *class* secara tipikal sebaiknya dapat dikenali oleh seorang *non-programmer* sekalipun terkait dengan domain permasalahan yang ada, dan kode yang terdapat dalam sebuah *class* sebaiknya (relatif) bersifat mandiri dan independen (sebagaimana kode tersebut digunakan jika tidak menggunakan OOP). Dengan modularitas, struktur dari sebuah program akan terkait dengan aspek-aspek dalam masalah yang akan diselesaikan melalui program tersebut. Cara seperti ini akan menyederhanakan pemetaan dari masalah ke sebuah program ataupun sebaliknya.
2. Objek, membungkus data dan fungsi bersama menjadi suatu unit dalam sebuah program komputer. Objek merupakan dasar dari modularitas dan struktur dalam sebuah program komputer berorientasi objek.
3. Abstraksi , kemampuan sebuah program untuk melewati aspek informasi yang diproses olehnya, yaitu kemampuan untuk memfokus pada inti. Setiap objek dalam sistem melayani sebagai model dari

"pelaku" abstrak yang dapat melakukan kerja, laporan dan perubahan keadaannya, dan berkomunikasi dengan objek lainnya dalam sistem, tanpa mengungkapkan bagaimana kelebihan ini diterapkan. Proses, fungsi atau metode dapat juga dibuat abstrak, dan beberapa teknik digunakan untuk mengembangkan sebuah pengabstrakan.

4. Enkapsulasi, memastikan pengguna sebuah objek tidak dapat mengganti keadaan dalam dari sebuah objek dengan cara yang tidak layak; hanya metode dalam objek tersebut yang diberi ijin untuk mengakses keadaannya. Setiap objek mengakses interface yang menyebutkan bagaimana objek lainnya dapat berinteraksi dengannya. Objek lainnya tidak akan mengetahui dan tergantung kepada representasi dalam objek tersebut.
5. Polimorfisme melalui pengiriman pesan. Tidak bergantung kepada pemanggilan subrutin, bahasa orientasi objek dapat mengirim pesan; metode tertentu yang berhubungan dengan sebuah pengiriman pesan tergantung kepada objek tertentu di mana pesa tersebut dikirim. Contohnya, bila sebuah burung menerima pesan "gerak cepat", dia akan menggerakan sayapnya dan terbang. Bila seekor singa menerima pesan yang sama, dia akan menggerakkan kakinya dan berlari. Keduanya menjawab sebuah pesan yang sama, namun yang sesuai dengan kemampuan hewan tersebut. Ini disebut polimorfisme karena sebuah variabel tungal dalam program dapat memegang berbagai jenis objek yang berbeda selagi program berjalan, dan teks program yang

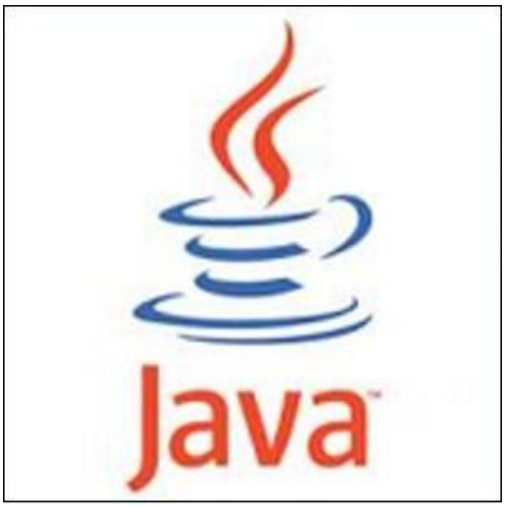
sama dapat memanggil beberapa metode yang berbeda di saat yang berbeda dalam pemanggilan yang sama. Hal ini berlawanan denganbahasa fungsional yang mencapai polimorfisme melalui penggunaan fungsi kelas-pertama.

6. Inheritas- Mengatur polimorfisme dan enkapsulasi dengan mengijinkan objek didefinisikan dan diciptakan dengan jenis khusus dari objek yang sudah ada - objek-objek ini dapat membagi (dan memperluas) perilaku mereka tanpa haru mengimplementasi ulang perilaku tersebut (bahasa berbasis-objek tidak selalu memiliki inheritas.)
7. Dengan menggunakan OOP maka dalam melakukan pemecahan suatu masalah kita tidak melihat bagaimana cara menyelesaikan suatu masalah tersebut (terstruktur) tetapi objek-objek apa yang dapat melakukan pemecahan masalah tersebut. Sebagai contoh anggap kita memiliki sebuah departemen yang memiliki manager, sekretaris, petugas administrasi data dan lainnya. Misal manager tersebut ingin memperoleh data dari bag administrasi maka manager tersebut tidak harus mengambilnya langsung tetapi dapat menyuruh petugas bag administrasi untuk mengambilnya. Pada kasus tersebut seorang manager tidak harus mengetahui bagaimana cara mengambil data tersebut tetapi manager bisa mendapatkan data tersebut melalui objek petugas adminiistrasi. Jadi untuk menyelesaikan suatu masalah

dengan kolaborasi antar objek-objek yang ada karena setiap objek memiliki deskripsi tugasnya sendiri.

B. Java

Java adalah bahasa pemograman yang dapat dijalankan di berbagai berbagai komputer termasuk telepon genggam. Dikembangkan oleh Sun Microsystems dan dirilis tahun 1995. Java berbeda dengan JavaScript. JavaScript adalah bahasa scripting yang digunakan oleh web browser.



Gambar 2. 18 Logo Java

Bahasa pemrograman Java pertama lahir dari The Green Project, yang berjalan selama 18 bulan, dari awal tahun 1991 hingga musim panas 1992. Proyek tersebut belum menggunakan versi yang dinamakan Oak. Proyek ini dimotori oleh Patrick Naughton, Mike Sheridan,James Gosling dan Bill Joy, beserta sembilan pemrogram lainnya dari Sun Microsystems. Salah satu hasil proyek ini adalah mascot Duke yang dibuat oleh Joe Palrang.

Nama Oak, diambil dari pohon oak yang tumbuh di depan jendela ruangan kerja "Bapak Java", James Gosling. Nama Oak ini tidak dipakai untuk versi release Java karena sebuah perangkat lunak lain sudah terdaftar dengan merek dagang tersebut, sehingga diambil nama penggantinya menjadi "Java". Nama ini diambil dari kopi murni yang digiling langsung dari biji (kopi tubruk) kesukaan Gosling. Konon kopi ini berasal dari Pulau Jawa. Jadi nama bahasa pemrograman Java tidak lain berasal dari kata Jawa (bahasa Inggris untuk Jawa adalah Java).

Kelebihan:

1. Multiplatform. Kelebihan utama dari Java ialah dapat dijalankan di beberapa platform / sistem operasi komputer, sesuai dengan prinsip tulis sekali, jalankan di mana saja. Dengan kelebihan ini pemrogram cukup menulis sebuah program Java dan dikompilasi (diubah, dari bahasa yang dimengerti manusia menjadi bahasa mesin / *bytecode*) sekali lalu hasilnya dapat dijalankan di atas beberapa platform tanpa perubahan. Kelebihan ini memungkinkan sebuah program berbasis java dikerjakan diatas *operating system* Linux tetapi dijalankan dengan baik di atas Microsoft Windows. Platform yang didukung sampai saat ini adalah Microsoft Windows, Linux, Mac OS dan Sun Solaris. Penyebanya adalah setiap sistem operasi menggunakan programnya sendiri-sendiri (yang dapat diunduh dari situs Java) untuk meninterpretasikan *bytecode* tersebut.
2. *Objek Oriented Programming* (*OOP*) yang artinya semua aspek yang terdapat di Java adalah objek. Java merupakan salah satu bahasa pemrograman berbasis objek secara murni. Semua tipe data diturunkan dari kelas dasar yang

disebut objek. Hal ini sangat memudahkan pemrogram untuk mendesain, membuat, mengembangkan dan mengalokasi kesalahan sebuah program dengan basis Java secara cepat, tepat, mudah dan terorganisir. Kelebihan ini menjadikan Java sebagai salah satu bahasa pemograman termudah, bahkan untuk fungsi fungsi yang *advance* seperti komunikasi antara komputer sekalipun.

3. Perpustakaan kelas Yang Lengkap. Java terkenal dengan kelengkapan *library/*perpustakaan (kumpulan program-program yang disertakan dalam pemrograman java) yang sangat memudahkan dalam penggunaan oleh para pemrogram untuk membangun aplikasinya. Kelengkapan perpustakaan ini ditambah dengan keberadaan komunitas Java yang besar yang terus menerus membuat perpustakaan-perpustakaan baru untuk melingkupi seluruh kebutuhan pembangunan aplikasi.
4. Bergaya C++, memiliki sintaks seperti bahasa pemrograman C++ sehingga menarik banyak pemrogram C++ untuk pindah ke Java. Saat ini pengguna Java sangat banyak, sebagian besar adalah pemrogram C++ yang pindah ke Java. Universitas-universitas di Amerika Serikat juga mulai berpindah dengan mengajarkan Java kepada murid-murid yang baru karena lebih mudah dipahami oleh murid dan dapat berguna juga bagi mereka yang bukan mengambil jurusan komputer.
5. Pengumpulan sampah otomatis, memiliki fasilitas pengaturan penggunaan memori sehingga para pemrogram tidak perlu melakukan pengaturan memori secara langsung (seperti halnya dalam bahasa C++ yang dipakai secara luas).

Kekurangan :

1. Tulis sekali, perbaiki di mana saja. Masih ada beberapa hal yang tidak kompatibel antara platform satu dengan platform lain. Untuk J2SE, misalnya *SWT-AWT bridge* yang sampai sekarang tidak berfungsi pada Mac OS X.
2. Mudah didekompilasi. Dekompilasi adalah proses membalikkan dari kode jadi menjadi kode sumber. Ini dimungkinkan karena kode jadi Java merupakan *bytecode* yang menyimpan banyak atribut bahasa tingkat tinggi, seperti nama-nama kelas, metode, dan tipe data. Hal yang sama juga terjadi pada Microsoft .NET Platform. Dengan demikian, algoritma yang digunakan program akan lebih sulit disembunyikan dan mudah dibajak/*direverse-engineer.*
3. Penggunaan memori yang banyak. Penggunaan memori untuk program berbasis Java jauh lebih besar daripada bahasa tingkat tinggi generasi sebelumnya seperti C/C++ danPascal (lebih spesifik lagi, Delphi dan Objek Pascal). Biasanya ini bukan merupakan masalah bagi pihak yang menggunakan teknologi terbaru (karena trend memori terpasang makin murah), tetapi menjadi masalah bagi mereka yang masih harus berkutat dengan mesin komputer berumur lebih dari 4 tahun.

2.2.5.2 Unified Modelling Language (UML)

Unified Modeling Language (UML) adalah bahasa spesifikasi standar untuk mendokumentasikan, menspesifikasikan, dan membangun sistem perangkat lunak.

Unified Modeling Language (UML) adalah himpunan struktur danteknik untuk pemodelan desain program berorientasi objek (OOP) serta aplikasinya.UML adalah metodologi untuk mengembangkansistem OOP dan sekelompok perangkat tool untuk mendukung pengembangan sistem tersebut UML mulai diperkenalkan oleh *Objek Management Group*, sebuah organisasi yang telah mengembangkan model, teknologi, dan standar OOP sejak tahun 1980-an. Sekarang UML sudah mulai banyak digunakan oleh para praktisi OOP. UML merupakan dasar bagi perangkat *(tool)* desain berorientasi objek dari IBM

UML adalah suatu bahasa yang digunakan untuk menentukan, memvisualisasikan, membangun, dan mendokumentasikan suatu sistem informasi. UML dikembangkan sebagai suatu alat untuk analisis dan desain berorientasi objek oleh Grady Booch, Jim Rumbaugh, dan Ivar Jacobson.Namun demikian UML dapat digunakan untuk memahami dan mendokumentasikan setiap sistem informasi.Penggunaan UML dalam industri terus meningkat. Ini merupakan standar terbuka yang menjadikannya sebagai bahasa pemodelan yang umum dalam industri peranti lunak dan pengembangan sistem.

A. Diagram UML

UML menyediakan 10 macam diagram untuk memodelkan aplikasi berorientasi objek, yaitu:

1. *Use Case Diagram* untuk memodelkan proses bisnis.
2. *Conceptual Diagram* untuk memodelkan konsep-konsep yang ada di dalam aplikasi.

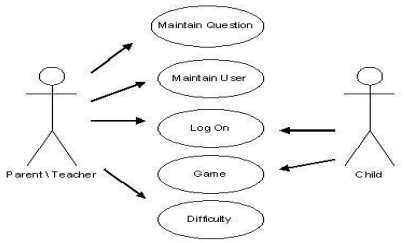
3. *Sequence diagram* untuk memodelkan pengiriman pesan *(message)* antar objek.
4. *Collaboration Diagram* untuk memodelkan interaksi antar objek.
5. *State Diagram* untuk memodelkan perilaku objek di dalam sistem.
6. *Activity diagram* untuk memodelkan perilaku *user*dan objek di dalam sistem.
7. *Class Diagram* untuk memodelkan struktur kelas.
8. *Objek Diagram* untuk memodelkan struktur objek.
9. *Component Diagram* untuk memodelkan komponen objek.
10. *Deployment Diagram* untuk memodelkan distribusi aplikasi.

Berikut akan dijelaskan 4 macam diagram yang paling sering digunakan dalam pembangunan aplikasi berorientasi objek, yaitu *use case diagram, sequence diagram, collaboration diagram,* dan*class diagram.*

B. *Use Case Diagram*

Use case diagram digunakan untuk memodelkan bisnis proses berdasarkan perspektif pengguna sistem. *Use case diagram* terdiri atas diagram untuk *use case* dan *actor*. Actormerepresentasikan orang yang akan mengoperasikan atau orang yang berinteraksi dengan sistem aplikasi.

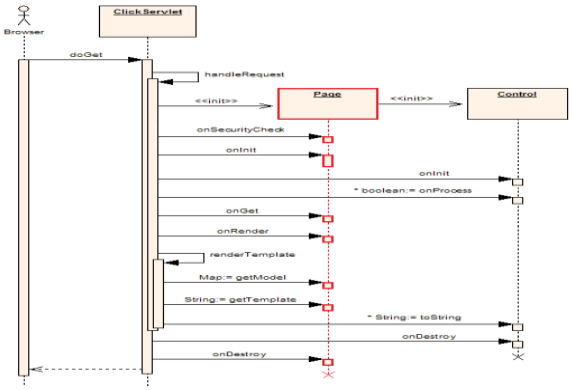
Use case merepresentasikan operasi-operasi yang dilakukan oleh *actor.Use case* digambarkan berbentuk elips dengan nama operasi dituliskan didalamnya. Actor yang melakukan operasi dihubungkan dengan garis lurus ke *use case.*



Gambar 2. 19 Diagram Use Case

C. *Sequence diagram*

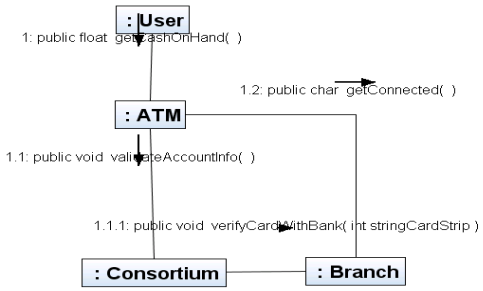
Sequence diagram menjelaskan secara detil urutan proses yang dilakukan dalam sistem untuk mencapai tujuan dari *use case*. Interaksi yang terjadi antar *class*, operasi apa saja yang terlibat, urutan antar operasi, dan informasi yang diperlukan oleh masing-masing operasi.



Gambar 2. 20*Sequence diagram*

D. *Collaboration Diagram*

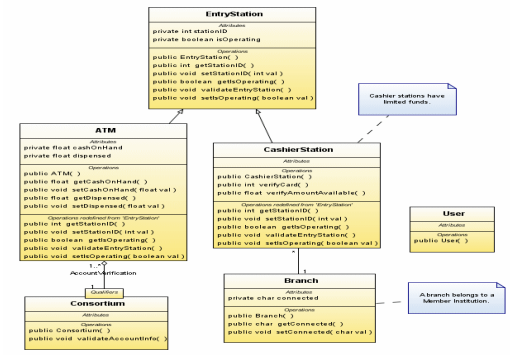
Collaboration diagram dipakai untuk memodelkan interaksi antar objek di dalam sistem.Berbeda dengan *sequence diagram* yang lebih menonjolkan kronologis dari operasi-operasi yang dilakukan, *collaboration diagram* lebih fokus pada pemahaman atas keseluruhan operasi yang dilakukan oleh objek.



Gambar 2. 21*Collaboration Diagram*

E. *Class Diagram*

Class diagram merupakan diagram yang selalu ada di permodelan sistem berorientasi objek. *Class* diagram menunjukkan hubungan antar *class* dalam sistem yang sedang dibangun dan bagaimana mereka saling berkolaborasi untuk mencapai suatu tujuan.



Gambar 2. 22*Class Diagram*

2.2.6 Aplikasi Pendukung

2.2.6.1 NetBeans

NetBeans adalah *sebuah framework platform* untuk aplikasi Java *desktop* dan *Integrated development envirovment (IDE)* untuk mengembangkan aplikasi Java, Javascript, PHP, Ruby dan lain sebagainya.

Netbeans IDE dibuat dengan pemograman Java dan dapat dijalankan dimana saja dengan bantuan JVM, baik itu di Windows, Mac OS, Linux dan Solaris. JDK diperlukan untuk pengembangan apliakasi dengan pemograman Java.

NetBeans IDE adalah sebuah IDE open source. NetBeans IDE mendukung pengembangan aplikasi Java.

2.2.6.2 MySQL

Pada awalnya, *MySQL* merupakan proyek internal sebuah firma asal Swedia, *TcXDataKonsult*. MySQL kemudian dirilis untuk publik pada tahun 1996. Karena *MySQL* menjadi sangat populer, pada tahun 2001 firma tersebut mendirikan sebuah perusahaan baru, *MySQLAB*, yang khusus menawarkan layanan dan produk berbasis *MySQL* (Gilmore, 2006).

Dari awal pembuatannya, para pengembang *MySQL* menitikberatkan pengembangan *MySQL* pada sisi performa dan skalabilitasnya. Hasilnya adalah sebuah perangkat lunak yang sangat teroptimasi, walaupun dari sisi fitur memiliki kekurangan dibandingkan solusi basis data kelas enterprise lain. Akan tetapi *MySQL* menarik minat banyak pengguna. Saat ini, tercatat lebih dari lima juta basis data *MySQL* yang terpasang dan aktif di seluruh dunia. Beberapa perusahaan dan instansi penting dunia seperti *Yahoo!*, *Google* dan *NASA* menggunakan *MySQL* untuk mengolah basis data mereka.

Ada beberapa kelebihan yang dimiliki *MySQL* sehingga dapat menarik banyak pengguna. Kelebihan tersebut yaitu:

1. Fleksibilitas

   Saat ini, *MySQL* telah dioptimasi untuk duabelas platform seperti *HP-UX, Linux, Mac OS X, Novell Netware, OpenBSD, Solaris, Microsoft Windows* dan lain-lain. *MySQL* juga menyediakan *source code* yang dapat diunduh secara gratis, sehingga pengguna dapat mengkompilasi sendiri sesuai platform yang digunakan. Selain itu, *MySQL* juga dapat dikustomisasi sesuai keinginan penggunanya, misalnya mengganti bahasa yang digunakan pada antarmukanya.

2. Performa

   Sejak rilis pertama, pengembang *MySQL* fokus kepada performa. Hal ini masih tetap dipertahankan hingga sekarang dengan terus meningkatkan fiturnya.

3. Lisensi

   MySQL menawarkan berbagai pilihan lisensi kepada penggunanya. Lisensi *open source* yang ditawarkan yaitu lisensi *GNU General Public License* dan *Free/Libre and Open Source Software (FLOSS) License Exception*. Selain itu ditawarkan juga lisensi komersil berbayar yang memiliki fasilitas dukungan teknis.