

BAB II

LANDASAN TEORI

II.1. Aplikasi

Program aplikasi adalah program siap pakai atau program yang dirancang untuk melaksanakan suatu fungsi bagi pengguna atau aplikasi yang lain. Aplikasi juga diartikan sebagai penggunaan atau penerapan suatu konsep yang menjadi pokok pembahasan atau sebagai program komputer yang dibuat untuk menolong manusia dalam melaksanakan tugas tertentu. Aplikasi software yang dirancang untuk penggunaan praktisi khusus, klasifikasi luas ini dapat dibagi menjadi 2 (dua) yaitu :

1. Aplikasi *software* spesialis, program dengan dokumentasi tergabung yang dirancang untuk menjalankan tugas tertentu.
2. Aplikasi paket, suatu program dengan dokumentasi tergabung yang dirancang untuk jenis masalah tertentu (Rahmatillah ; 2011: 3).

Dan Aplikasi berasal dari kata *application* yang artinya penerapan, lamaran, penggunaan. Secara istilah aplikasi adalah program yang siap pakai yang direka untuk melaksanakan suatu fungsi bagi pengguna atau aplikasi yang lain dan dapat digunakan oleh sasaran yang dituju. Aplikasi merupakan perangkat lunak yang dimasukkan atau terdapat dalam komputer dan memiliki fungsi-fungsi khusus. (Iko Ari Guna ; 2014 : 10)

Aplikasi merupakan salah satu subkelas perangkat lunak komputer yang memanfaatkan kemampuan komputer langsung untuk melakukan suatu tugas yang diinginkan pengguna. Contoh utama aplikasi adalah pengolah kata, lembar kerja,

memanipulasi foto, merancang rumah dan pemutar media. Beberapa aplikasi yang digabung bersama menjadi suatu paket atau *suite* aplikasi (*application suite*). Contohnya adalah Microsoft Office dan OpenOffice.org, yang menggabungkan suatu aplikasi pengolah kata, lembar kerja dan beberapa aplikasi lainnya. Aplikasi-aplikasi dalam suatu paket biasanya memiliki antarmuka pengguna yang memiliki kesamaan sehingga memudahkan pengguna untuk mempelajari dan menggunakan tiap aplikasi. Sering kali, mereka memiliki kemampuan untuk saling berinteraksi satu sama lain sehingga menguntungkan pengguna. Contohnya, suatu lembar kerja dapat dibenamkan dalam suatu dokument pengolah kata walaupun dibuat pada aplikasi lembar kerja yang terpisah. (Verawaty Simbolon : 2014 : 10)

II.2. Android

Android adalah sistem operasi disematkan pada gadget, baik itu handphone, tablet, juga sekarang sudah merambah ke kamera digital dan jam tangan. Saat ini gadget berbasis android, baik itu tablet atau handphone, begitu digandrungi. Selain harganya yang semakin terjangkau, juga banyak varian spesifikasi yang bisa dipilih sesuai kebutuhan. Perkembangan android sangat cepat. Di awal tahun 2002 ada 200 juta pengguna aktif android, dan google play mampu menampung 400.000 aplikasi yang siap digunakan, dan total mencapai 10 triliun kali aplikasi yang sudah di download lewat android market, pertumbuhan yang luar biasa. Jumlah ini diyakini akan terus bertambah seiring waktu dan perkembangan teknologi (Agus Wahadyo ; 2013 : 3).

II.2.1. Arsitektur Android

Secara garis besar Arsitektur Android dapat dijelaskan di gambarkan sebagai berikut :

1. *Applications* dan *Widgets*

Application dan *Widgets* ini adalah *layer* dimana kita berhubungan dengan aplikasi saja, dimana biasanya kita *download* aplikasi kemudian kita lakukan instalasi dan jalankan aplikasi tersebut. Di *layer* terdapat aplikasi inti termasuk *client email*, program SMS, kalender, peta, *browser*, kontak, dan lain-lain. Semua aplikasi ditulis menggunakan bahasa pemrograman Java.

2. *Applications Frameworks*

Android adalah *Open Development Platform* yaitu Android menawarkan kepada pengembang atau member kemampuan kepada pengembang untuk membangun aplikasi yang bagus dan inovatif. Pengembang bebas untuk mengakses perangkat keras, akses informasi *resources*, menjalankan *service background*, mengatur alarm, dan menambahkan status *notification*, dan sebagainya. Pengembang memiliki akses penuh menuju *API framework* seperti dirancang supaya kita dengan mudah dapat menggunakan kembali komponen yang sudah digunakan (*reuse*).

3. *Libraries*

Libraries adalah *layer* dimana fitur-fitur Android berada, biasanya para pembuat aplikasi mengakses *libraries* untuk menjalankan aplikasinya.

4. *Android Run Time*

Layer yang membuat aplikasi Android dapat dijalankan dimana prosesnya menggunakan implementasi *Linux*. *Dalvik Virtual Machine* (DVM) merupakan mesin yang membentuk dasar kerangka aplikasi Android. Di dalam *Android Run Time* dibagi menjadi dua bagian yaitu :

- a. *Core Libraries* adalah aplikasi Android dibangun dalam bahasa java sementara *Dalvik* sebagai virtual mesinnya bukan *Virtual Machine Java*, sehingga diperlukan sebuah *libraries* yang berfungsi untuk menterjemahkan bahasa java yang ditanda tangani oleh *Core Libraries*.
- b. *Dalvik Virtual Machine* adalah virtual mesin berbasis register yang dioptimalkan untuk menjalankan fungsi-fungsi secara efisien, dimana merupakan pengembangan yang mampu membuat *linux karnel* untuk melakukan *threading* dan manajemen tingkat rendah.

5. *Linux Karnel*

Linux karnel adalah inti dari *operating system* dari Android itu berada. Berisi file-file system yang mengatur system *processing*, *memory*, *resource*, *drivers*, dan system-sistem operasi Android lainnya. *Linux karnel* yang digunakan Android adalah *linux karnel relase 2.6* (Nazruddin Safaat H,2011:6-8).

II.2.2. Versi Android

Telepon pertama yang memakai system operasi Android adalah HTC Dream, yang dirilis pada 22 Oktober 2008, Pada penghujung tahun 2010 diperkirakan hampir semua vendor seluler didunia menggunakan Android sebagai *operating system*. Adapun versi-versi Android yang pernah dirilis sebagai berikut (Nazruddin Safaat H,2011:11-12) :

1. Android 1.1 *Bender*

Pada 9 Maret 2009, *Google* merilis Android versi 1.1. Android versi ini dilengkapi dengan pembaruan estetis pada aplikasi, jam, alarm, *voice search* (pencarian suara), pengiriman pesan dengan *Gmail*, dan pemberitahuan *email*.

2. Android 1.5 *Cupcake*

Pada pertengahan Mei 2009, *Google* kembali merilis telepon seluler dengan menggunakan Android dan SDK (*Software Development Kit*) dengan versi 1.5 (*Cupcake*). Terdapat beberapa pembaruan termasuk juga penambahan beberapa fitur dalam seluler versi ini yakni kemampuan merekam dan menonton video dengan modus kamera, mengupload video ke *youtube* dan gambar ke *pisaca* langsung dari telepon, dukungan *Bluetooth A2DP*, kemampuan terhubung secara otomatis ke *headset Bluetooth*, animasi layar, dan *keyboard* pada layar yang dapat disesuaikan dengan system.

3. Android 1.6 *Donut*

Donut dirilis pada September dengan menampilkan proses pencarian yang baik disbanding sebelumnya, penggunaan baterai indicator dan control applet *VPN*. Fitur lainnya adalah galeri yang memungkinkan pengguna untuk memilih foto yang akan dihapus, kamera, *camcorder* dan galeri yang diintegrasikan, *CDMA/EVDO*, *802.IX*, *VPN*, *Gestures*, dan *Text to engine*, kemampuan dial kontak, teknologi *text to change speech* (tidak tersedia pada semua ponsel, pengadaan resolusi *VWGA*).

4. Android 2.0/2.1 *Éclair*

Pada 3 Desember 2009 kembali diluncurkan ponsel Android dengan versi 2.0/2.1 (*Éclair*), perubahan yang dilakukan adalah pengoptimalan *hardware*,

peningkatan *Google Maps* 3.1.2, perubahan *UI* dengan *browser* baru dan dukungan *HTML5*, daftar kontak yang baru, dukungan *flash* untuk kamera 3.2 MP, digital *Zoom*, dan *Bluetooth* 2.1.

5. Android 2.2 *Froyo*

Pada bulan Mei 2010 Android versi 2.2 Rev 1 diluncurkan. Android inilah yang sekarang sangat banyak beredar dipasaran, salah satunya adalah dipakai di *Samsung FX Tab* yang sudah ada dipasaran.

6. Android 2.3 *Gingerbread*

Android versi 2.3 diluncurkan pada Desember 2010, hal-hal yang direvisi dari versi sebelumnya adalah kemampuan seperti berikut *SIP based VoIP*, *Near Field Communications*, *Gyroscope* dan *sensor*, *Multiple cameras support*, *Mixable audio effects*, *Download manager*.

II.3. Database

Database adalah sekumpulan data yang disimpan secara teratur sehingga memudahkan dalam pencarian kembali, pengelompokkan, dan pengolahannya menjadi informasi. *Database* dapat dibayangkan sebagai sebuah lemari arsip yang memiliki prinsip kerja dan tujuan yang sama. Prinsip utamanya adalah pengaturan data/arsip. Dan tujuan utamanya adalah kemudahan dan kecepatan dalam pengambilan kembali data/arsip. Perbedaannya hanya terletak pada media penyimpanan yang digunakan. Jika lemari arsip menggunakan lemari dari besi atau kayu sebagai media penyimpanan, maka basis data menggunakan media penyimpanan elektronik seperti *magnetic disk*. (Fathansyah, 2012 : 3)

Database juga merupakan “jiwa” dari sebuah aplikasi. Sebab dengan memanfaatkan database, semua fitur, tool, menu, dan fasilitas lainnya yang ada

didalam aplikasi, dapat terhubung satu dengan lainnya. database tidak hanya sekedar tempat untuk menyimpan data. Database bisa digunakan untuk memfasilitasi user yang membutuhkan pemrosesan data baik untuk analisa maupun evaluasi. (Gregorius Agung, 2015: 2).

Prinsip dasar pemrograman database sebenarnya sederhana. Pemrograman database menggunakan empat aksi dasar yang disebut dengan istilah CRUD. Istilah ini sebenarnya mengacu kepada empat aksi tersebut, yaitu Create, Read, Update, dan Delete. Artinya:

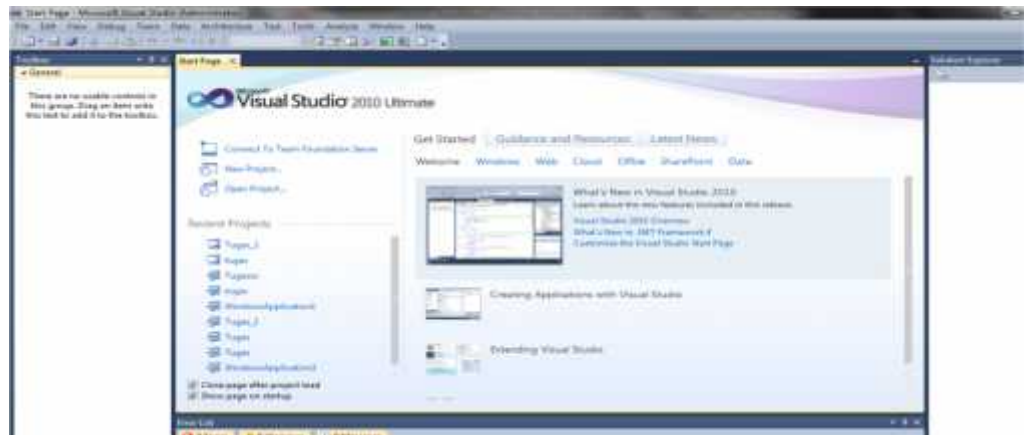
- a. Create: data diinput ke dalam sistem database baik oleh user maupun sistem.
- b. Read: data dibaca dari sistem database untuk diolah dan ditampilkan kedalam sebuah aplikasi sesuai perintah user .
- c. Update: data yang tersimpan disistem database diperbarui mengikuti perintah user.
- d. Delete: data yang tersimpan didalam sistem database dihapus sebagian atau seluruhnya sesuai dengan perintah user.

Aplikasi sederhana maupun kompleks, selama memanfaatkan database, akan selalu mengikuti “rutinitas diatas. Oleh karena itu dapat dikatakan, dasar dari pemrograman database adalah CRUD.

II.4. VB.Net dan *Microsoft Visual Basic 2010*

VB .Net adalah bahasa pemrograman untuk membuat aplikasi yang nantinya dengan aplikasi tersebut akan mempermudah bagi pengguna baik untuk melihat data, laporan, dan lain-lain. dan dewasa ini Programmer VB .Net banyak

dipakai diperusahaan kecil maupun besar sehingga jika anda belajar VB .Net akan memudahkan anda dalam berkarir maupun mencari pekerjaan.



Gambar II.1. Tampilan awal Visual Basic 2010

Identik dengan menulis kode di Visual Studio 2010, Visual Basic adalah bahasa pemrograman yang sangat populer. Kecepatan dan kemudahan penggunaan membuat sering pilihan pertama bagi programmer baru, serta pilihan sangat disukai untuk lebih berpengalaman mengatur bersemangat untuk belajar iterasi terbaru Visual Basic. Panduan mulai ini menyediakan dengan dasar yang kuat, membuka kekuatan dan kemungkinan Visual Basic 2010 dan memberikan langkah-langkah rinci untuk dengan cepat dan mudah menulis program yang berguna. (Thearon, Bryann : 2011).

Sebelum kita lebih jauh belajar mengenai pemrograman Visual Basic 2010 alangkah baiknya jika kita mengenal lebih dekat apa itu Visual Basic serta perkembangan versi-versi terbarunya. Visual Basic adalah salah satu bahasa pemrograman berbasis desktop yang dikeluarkan (diproduksi) oleh perusahaan perangkat lunak komputer terbesar yaitu Microsoft. Visual Basic merupakan salah

satu bahasa pemrograman paling laris dan paling sukses didunia. Dimana tercatat sampai tahun 2005 Visual Basic merupakan bahasa pemrograman yang yang paling banyak dipakai oleh para programmer bahkan diyakini sampai saat ini. Menjadi pilihan berbagai kalangan tentunya Visual Basic memiliki berbagai hal yang patut dijadikan alasan, selain bahasa pemrograman yang sangat (paling) mudah dipelajari oleh berbagai kalangan baik awam maupun ahli Visual Basic yang didukung penuh oleh produsennya (Microsoft) selalu dikembangkan dan disesuaikan dengan kebutuhan zaman seperti penyesuaian model pemrograman modern yang berbasis OOP (Object Oriented Programing).

Visual Basic diturunkan dari bahasa *BASIC*. *Visual Basic* terkenal sebagai bahasa pemrograman yang mudah digunakan terutama untuk membuat aplikasi yang berjalan di atas *platform Windows*. Pada tahun 90an, *Visual Basic* menjadi bahasa pemrograman yang paling populer dan menjadi pilihan utama untuk mengembangkan program berbasis *Windows*. Visual Basic termasuk bahasa pemrograman event – driven generasi ketiga dan Integrated Development Environment (IDE) dari Microsoft yang mulai diperkenalkan pada tahun 1991. Visual Basic merupakan pengembangan dari BASIC yang dibuat saebagai bahasa pemrograman yang mudah dipelajari dan digunakan. Visual Basic memungkinkan proses Rapid Application Development (RAD) dari aplikasi antarmuka, mengakses database, dan membuat kontrol dan objek. . (Gregorius Agung, 2015: 2).

VB .NET adalah salah satu bahasa pemrograman komputer tingkat tinggi salah satu bahasa pemrograman yang *Object Oriented Program (OOP)* atau pemrograman yang berorientasi pada object. Kata “*Visual*” menunjukkan cara

yang digunakan untuk membuat *Graphical User Interface (GUI)* tidak perlu lagi menuliskan instruksi pemrograman dalam kode-kode baris hanya untuk membuat sebuah desain aplikasi cukup melakukan *drag and drop object-object* yang akan digunakan. (Muhammad Soufian ; Skripsi, 2014 : 19).

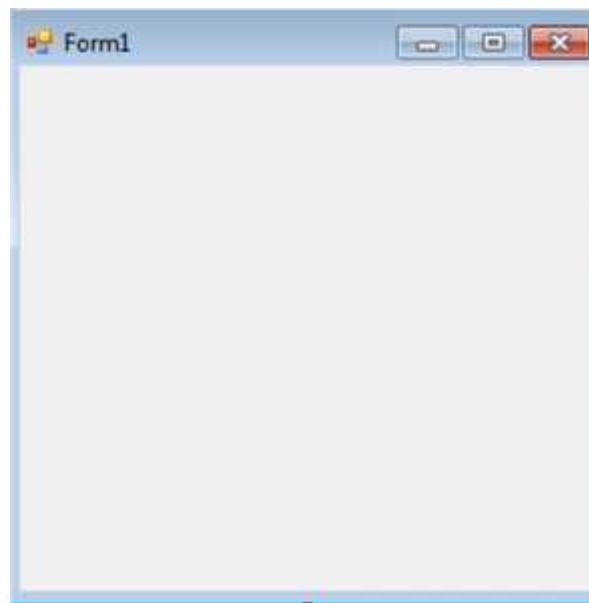
Adapun objek-objek yang digunakan dalam program ini :

1. Form

Form merupakan komponen VB yang memiliki sifat container, karena fungsi utama dari form adalah sebagai tempat komponen VB yang lain.

Pembuatan form dalam VB dapat kita lakukan dengan menggunakan menu

Project > Add Windows Form



Gambar II.2. Tampilan Form

2. ToolBox

ToolBox merupakan jendela yang berisikan group dari bermacam-macam control dan komponen yang dapat dipasang di dalam form. kontrol seperti Textbox, Button, Radio Button, Checkbox, Combo box dan lain-lain

ditambahkan ke dalam form dengan cara drag dan drop, atau mengklik dan menggoreskannya pada form.



Gambar II.3. Tampilan Toolbox

Beberapa kontrol pada ToolBox :

- a. Button : komponen yang digunakan untuk memberikan aksi saat di tekan. Button menjalankan proses menyimpan, mengubah, menghapus, dan lain-lain.
- b. Label : untuk menampilkan teks. Biasa digunakan untuk memberikan informasi pada kontrol lain.
- c. Textbox : untuk input data. Komponen ini paling sering digunakan bersama button.
- d. Checkbox: digunakan untuk memberikan pilihan input kepada user user dapat memilih lebih dari 1 item data.
- e. Combobox : komponen yang menampilkan pilihan secara drop down.
- f. Listbox : sebuah kotak yang di dalamnya berisi item-item , listbox menampilkan item lebih dari 1.

menyimpan banyak data dan mengimplementasikannya untuk kepentingan bisnis dan perusahaan. (Wahana Komputer, 2013 : 2),

SQL Server 2012 sekarang didukung pada saat instalasi Server Core dari Windows Server 2008 R2 dan Windows SQL dan SQL Server memiliki banyak fungsi built in yang bisa dipakai untuk banyak keperluan. Anda juga bisa membuat fungsi anda sendiri kalau mau. Adanya fungsi membuat SQL Server bisa dipakai untuk melakukan banyak hal. Sebuah fungsi di SQL Server bisa bersifat deterministik dan non deterministik. Fungsi deterministik akan mengembalikan kembalian dengan hasil yang sama dengan set input yang dimasukan, tak peduli kapan dieksekusinya. SQL memiliki banyak fungsi built in untuk melakukan kalkulasi data pada data. Ada 2 kategori fungsi, yaitu fungsi agregat dan fungsi skalar. Fungsi aggregate akan mengembalikan satu value yang dihitung dari nilai di kolom, sementara fungsi skalar akan mengembalikan nilai tertentu tergantung kepada nilai inputnya.

Contoh fungsi aggregate adalah :

- AVG(): Mengembalikan nilai rata-rata.
- STDEV(): Mengembalikan nilai standard deviasi.
- COUNT(): Mengembalikan jumlah baris.
- MAX(): Mengembalikan nilai terbesar.
- MIN(): Mengembalikan nilai terkecil.
- SUM(): Mengembalikan jumlah.
- Dan seterusnya.

Contoh fungsi Scalar antara lain:

- UPPER(): Mengonversi field ke upper case.

- LOWER(): Mengonversi field ke huruf kecil/lower case.
- LEN(): Mengembalikan panjang field text.
- ROUND(): Menggenapkan field numerik ke jumlah desimal yang ditentukan.
- GETDATE(): Mengembalikan waktu dan tanggal sistem sekarang.
- Dan lain sebagainya.

II.5.1. Mengenal Query Sql.

Ketika jendela query sudah ada, berarti Anda siap untuk mengutak atik database dengan query. Query adalah sebuah statement yang dibuat menggunakan bahasa SQL (Structured Query Language). Karena itu agar bisa mengutak-atik query, Anda perlu mengenal bahasa SQL yang merupakan standar pembuatan query. SQL menggunakan bahasa terstandardisasi yang awalnya dikembangkan oleh IBM untuk melakukan mengambil data, mengubah data, dan menentukan database relasional menggunakan bahasa deklaratif yang sudah dipahami. SQL bisa digunakan untuk melakukan berapa hal berikut.

- Mebuat database baru.
- Membuat tabel baru di database.
- Membuat view di database.
- Mengatur permission di tabel, prosedur, dan view

Walaupun SQL ini standar, kebanyakan sistem RDBMS yang ada sekarang mengimplementasikan versi SQL-nya sendiri-sendiri. Begitu pula dengan SQL Server. Microsoft SQL Server menggunakan versi sql yang disebut T-SQL (Transact-SQL). T-SQL adalah versi SQL khusus milik Microsoft yang sangat mirip dengan versi SQL standar, hanya saja ada fungsi tambahan dari microsoft,

dan fungsi built in lainnya. Bagian pertama dari query adalah data definition Language (DDL), Data Definition Language (DDL) ini mengatur struktur tabel dan index. Elemen dasar dari DDL adalah untuk statement CREATE, ALTER, RENAME, dan DROP. Untuk lebih jelas dapat dilihat sebagai berikut.

- CREATE, digunakan untuk membuat objek (seperti tabel) di database.
- DROP, digunakan untuk menghapus objek di database. Sebuah objek yang sudah dihapus biasanya tidak bisa dikembalikan lagi, alias penghapusannya bersifat permanen.
- ALTER, digunakan untuk memodifikasi struktur dari objek yang ada, misalnya untuk menambahkan kolom ke tabel yang masih ada.

II.6. Eclipse

Eclipse adalah sebuah IDE (Integrated Development environment) untuk mengembangkan perangkat lunak dan dapat dijalankan di semua platform (platform independent). Eclipse awalnya dikembangkan oleh IBM pada tanggal 5 November 2001, yang menginvestasikan sebanyak US\$ 40 juta untuk pengembangannya. Semenjak itu konsorsium Eclipse Foundation mengambil alih untuk pengembangan Eclipse lebih lanjut dan pengaturan organisasinya. Eclipse pada saat ini merupakan salah satu IDE favorit dikarenakan gratis dan *open source*, yang berarti setiap orang boleh melihat kode pemrograman perangkat lunak ini. Selain itu, kelebihan dari eclipse yang membuatnya populer adalah kemampuannya untuk dapat dikembangkan oleh pengguna dengan komponen yang dinamakan *plug-in*.

Terdapat dua jenis platform untuk pengembangan aplikasi BlackBerry ini, yaitu berbasis bahasa java (menggunakan Java Mobile Edition for BlackBerry)

dan berbasis Web (menggunakan Widgets for BlackBerry). Pada pengembangan aplikasi ini menggunakan aplikasi BlackBerry yang berbasis java. Dalam pengembangan aplikasi ini, tentu banyak kemiripan dengan melakukan pengembangan di perangkat telepon seluler lainnya yang berbasis java. Beberapa perbedaannya yaitu hasil keluaran dari source code yang dikompilasi berupa file berektensi .cod, yaitu file khusus untuk diinstal pada perangkat BlackBerry. Sedangkan IDE yang digunakan adalah Eclipse Galileo dengan tambahan plugins *JDE* yaitu *BlackBerry java Plug-in* untuk *OS BlackBerry 4.6*.

Dengan menggunakan plugins JDE (Java Development environment) ini, para developer dapat menggunakan banyak API yang disediakan. Adapun API yang dapat digunakan pada JDE ini berasal dari “java”, “javax”, “javax. Microedition”, “net.rim.BlackBerry.API”, “org.w3c.dom”, dan “org.xml”. Untuk API lengkap BlackBerry yang dapat digunakan terdapat di situs resmi BlackBerry.

II.7. Koperasi.

Menurut Johar (2009) ”koperasi merupakan usaha bersama yang berlandaskan asas kekeluargaan untuk meningkatkan kesejahteraan anggotanya, koperasi merupakan badan usaha yang mengorganisir pemanfaatan dan pendayagunaan sumber ekonomi anggota dengan dasar prinsip koperasi”.

Koperasi merupakan organisasi ekonomi yang berusaha menggerakkan sumber daya ekonomi demi memajukan kesejahteraan anggota, karena sumber daya ekonomi tersebut terbatas dan dalam mengembangkan koperasi harus mengutamakan kepentingan anggota maka koperasi harus mampu bekerja secara efisien dan mengikuti prinsip – prinsip dan kaidah – kaidah ekonomi.

Berikut ini beberapa prinsip koperasi :

1. Keanggotaan koperasi bersifat sukarela dan terbuka.
2. Pengelolaan koperasi dilakukan secara demokratis.
3. Sisa hasil usaha (SHU) yang merupakan keuntungan dari usaha yang dilakukn oleh koperasi dibagi berdasarkan besarnya jasa masing-masing anggota.
4. Modal diberi balas jasa secara terbatas.
5. Koperasi bersifat mandiri.

Koperasi adalah badan usaha yang beranggotakan orang-orang atau badan hukum koperasi dengan melandaskan kegiatannya berdasarkan prinsip koperasi sekaligus sebagai gerakan ekonomi rakyat yang berdasarkan asas kekeluargaan. Koperasi bertujuan untuk menyejahterakan anggotanya.

Didalam bukunya “ *Histoire des Doctrines Cooperative* “ mengemukakan bahwa koperasi terdiri atas produsen produsen kecil yang tergabung secara sukarela untuk mencapai tujuan bersama ,dengan saling bertukar jasa secara kolektif dan menanggung resiko bersama dengan mengerjakan sumber sumber yang disumbangkan oleh anggota.(*Dr. G. Mladenata 2012*).

II.8. Kontrak Kerja

Kontrak Kerja adalah suatu perjanjian antara pekerja dan pengusaha secara lisan dan/atau tulisan, baik untuk waktu tertentu maupun untuk waktu tidak tertentu yang memuat syarat-syarat kerja, hak dan kewajiban pekerja dan perusahaan, Kontrak Kerja/Perjanjian Kerja menurut Undang-Undang No.13/2003 tentang Ketenagakerjaan adalah perjanjian antara pekerja/buruh dengan

pengusaha atau pemberi kerja yang memuat syarat-syarat kerja, hak, dan kewajiban para pihak.

Robinson dan Rousseau (2008) menyatakan bahwa kontrak kerja secara umum mengacu pada dokumen tertulis yang mengatur hak dan kewajiban seorang karyawan dan tunduk pada peraturan perusahaan. Selanjutnya Robinson dan Rousseau (2008) menjelaskan bahwa kontrak mengikat karyawan dan perusahaan dalam suatu persatuan kerja, mengatur perilaku masing-masing pihak dalam perusahaan serta memungkinkan pencapaian tujuan perusahaan (Robinson dan Rousseau, 2008)

Selanjutnya Rousseau (dalam Conway dan Briner, 2009) mengemukakan bahwa kontrak kerja merupakan keyakinan individu, yang dibentuk dari organisasi, keyakinan tersebut mengacu pada persetujuan antara individu dan organisasinya. Sedangkan Menurut Herriot dan Pemberton (dalam Conway dan Briner, 2009) kontrak kerja merupakan persepsi organisasi dan individu tentang kewajiban masing-masing pihak yang terbentuk secara tidak langsung dalam hubungan kerja. Lebih jelasnya, Morrison and Robinson (dalam Conway dan Briner, 2009) mengemukakan bahwa kontrak kerja mengacu pada keyakinan-keyakinan karyawan mengenai kewajiban-kewajiban yang bersifat timbal balik antara karyawan dan organisasinya, di mana kewajiban tersebut didasarkan pada janji-janji yang dipersepsikan dan tidak disadari dengan penting oleh agen-agen yang ada pada organisasi.

Walaupun status hubungan kerja kontrak kesannya menguntungkan bagi Perusahaan, tapi ada aturan kontrak kerja yang harus dipatuhi dengan disiplin oleh pihak Perusahaan. Apabila ada kesalahan dalam penerapannya, maka status

Karyawan kontrak tersebut otomatis akan berubah menjadi Karyawan tetap. Seperti yang diatur dalam Undang-undang Tenaga Kerja No.013 Tahun 2003.

II.9.Sistem Terdistribusi

Sistem distribusi adalah sebuah sistem yang komponennya berada pada jaringan komputer. Komponen tersebut saling berkomunikasi dan melakukan koordinasi hanya dengan pengiriman pesan (message passing).

Sistem terdistribusi merupakan kebalikan dari Sistem Operasi Prosesor Jamak. Pada sistem tersebut, setiap prosesor memiliki memori lokal tersendiri. Kumpulan prosesornya saling berinteraksi melalui saluran komunikasi seperti LAN dan WAN menggunakan protokol standar seperti TCP/IP. Karena saling berkomunikasi, kumpulan prosesor tersebut mampu saling berbagi beban kerja, data, serta sumber daya lainnya.

Sistem terdistribusi dapat dikatakan sebagai suatu keberadaan beberapa komputer yang bersifat transparan dan secara normal, setiap sistem terdistribusi mengandalkan layanan yang disediakan oleh jaringan komputer.

Dalam penggunaannya sistem terdistribusi sangat diperlukan karena:

1. *Performance*

Sekumpulan prosesor dapat menyediakan kinerja yang lebih tinggi daripada komputer yang terpusat

2. *Distribution*

Banyak aplikasi yang terlibat, sehingga lebih baik jika dipisah dalam mesin yang berbeda (contoh: aplikasi perbankan, komersial)

3. *Reliability*

Jika terjadi kerusakan pada salah satu mesin, tidak akan mempengaruhi kinerja system secara keseluruhan

3. *Incremental Growth*

Mesin baru dapat ditambahkan jika kebutuhan proses meningkat

4. *Sharing Data/Resource*

Resource adalah:

- Segala hal yang dapat digunakan bersama dalam jaringan komputer.
- Meliputi hardware (e.g. *disk, printer, scanner*), juga software (berkas, basis data, obyek data).

5. *Communication*

Menyediakan fasilitas komunikasi antar manusia

Beberapa contoh dari sistem terdistribusi yaitu :

1. Internet, merupakan suatu bentuk jaringan global yang menghubungkan komputer dengan satu sama lainnya, yang dapat berkomunikasi dengan media IP sebagai protokol.
2. Intranet
 - Jaringan yang teradministrasi secara lokal
 - Biasanya proprietary
 - Terhubung ke internet (melalui firewall)
 - Menyediakan layanan internal dan eksternal
3. Sistem terdistribusi multimedia Biasanya digunakan pada infrastruktur internet
 - Karakteristik

Sumber data yang heterogen dan memerlukan sinkronisasi secara real time

- Video, audio, text Multicast

4. Mobile dan sistem komputasi ubiquitous

- Sistem telepon Cellular (e.g., GSM)

Resources dishare : frekuensi radio, waktu transmisi dalam satu frekuensi, bergerak

- Komputer laptop, ubiquitous computing

- Handheld devices, PDA, etc

5. World wide web

- Arsitektur client/server terbuka yang diterapkan di atas infrastruktur internet

- Shared resources (melalui URL)

6. Contoh distribusi yang lainnya seperti

- Sistem telepon seperti ISDN, PSTN

- Manajemen jaringan seperti Administrasi sumber jaringan

- Network File System (NFS) seperti Arsitektur untuk mengakses sistem file melalui jaringan.

II.9.1. Karakteristik Sistem Terdistribusi

Dalam system terdistribusi terdapat beberapa karakteristik yaitu :

1. *No global clock*

- Terdapat batasan pada ketepatan proses sinkronisasi clock pada sistem terdistribusi, oleh karena *asynchronous message passing*

- Pada sistem terdistribusi, tidak ada satu proses tunggal yang mengetahui *global state* sistem saat ini (disebabkan oleh *concurrency* dan *message passing*)

2. *Independent failure*

- Kemungkinan adanya kegagalan proses tunggal yang tidak diketahui
- Proses tunggal mungkin tidak peduli pada kegagalan sistem keseluruhan

3. *Concurrency of components*

- E.g. Beberapa pemakai *browser* mengakses suatu halaman web secara bersamaan.
- Bagaimana jika ada operasi update?

II.9.2 Model Sistem Terdistribusi

Dalam pelaksanaannya sistem terdistribusi memiliki berbagai bentuk (model), yaitu :

1. Sistem *client - server*

Merupakan bagian dari model sistem terdistribusi yang membagi jaringan berdasarkan pemberi dan penerima jasa layanan. Pada sebuah jaringan akan didapatkan: *file server*, *time server*, *directory server*, *printer server*, dan seterusnya.

2. Sistem *point to point*

Merupakan bagian dari model sistem terdistribusi dimana sistem dapat sekaligus berfungsi sebagai *client* maupun *server*.

3. Sistem terkluster

Adalah gabungan dari beberapa sistem individual (komputer) yang dikumpulkan pada suatu lokasi, saling berbagi tempat penyimpanan data

(*storage*), dan saling terhubung dalam jaringan lokal (*Local Area Network*). Sistem kluster memiliki persamaan dengan sistem paralel dalam hal menggabungkan beberapa CPU untuk meningkatkan kinerja komputasi. Jika salah satu mesin mengalami masalah dalam menjalankan tugas maka mesin lain dapat mengambil alih pelaksanaan tugas itu. Dengan demikian, sistem akan lebih handal dan *fault tolerant* dalam melakukan komputasi. Dalam hal jaringan, sistem kluster mirip dengan sistem terdistribusi (*distributed system*). Bedanya, jika jaringan pada sistem terdistribusi melingkupi komputer-komputer yang lokasinya tersebar maka jaringan pada sistem kluster menghubungkan banyak komputer yang dikumpulkan dalam satu tempat.

II.9.3. Permasalahan Sistem Terdistribusi

Masalah dengan sistem terdistribusi yang dapat dimunculkan antara lain berkaitan dengan :

- Software - bagaimana merancang dan mengatur software dalam Distribusi Sistem
- Ketergantungan pada infrastruktur jaringan
- Kemudahan akses ke data yang di share, memunculkan masalah keamanan

Dalam setiap penggunaan suatu sistem, banyak sekali ditemui permasalahan – permasalahan yang muncul, begitu juga dengan sistem terdistribusi. Selain permasalahan – permasalahan yang akan dihadapi terdapat tantangan – tantangan dalam sistem terdistribusi.

II.9.4. Tantangan Sistem Terdistribusi

Tantangan yang ada dalam Sistem Terdistribusi yaitu :

1. Keheterogenan komponen (*heterogeneity*)
2. Keterbukaan (*openness*)
3. Keamanan (*security*)
4. *Scalability*
5. Penanganan kegagalan (*failure handling*)
6. *Concurrency of components*
7. Transparansi

II.10. UML (*Unified Modelling Language*)

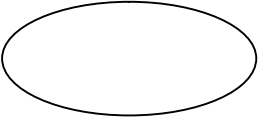
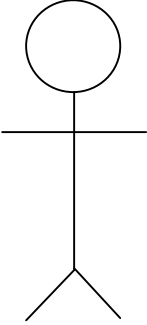


Penjadwalan telah lama diteliti, contohnya dalam penghasilan tenaga oleh *Windu Gata* dan *Grace Gata* (2013), UML (*Unified Modelling Language*) adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML (*Unified Modelling Language*) merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem.

II.10.1 *UseCase Diagram*

UseCase Diagram merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. Mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Dapat dikatakan *usecase* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut.

Simbol-simbol yang digunakan dalam *UseCase Diagram*, yaitu:

Tabel II.1. Diagram UseCase

Gambar	Keterangan
	<p><i>UseCase</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, biasanya dinyatakan dengan menggunakan kata kerja diawal nama <i>UseCase</i>.</p>
	<p><i>Actor</i> atau Aktor adalah <i>abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem. Orang atau sistem bisa muncul dalam beberapa peran. Perlu dicatat bahwa aktor berinteraksi dengan <i>usecase</i>, tetapi tidak memiliki kontrol terhadap <i>usecase</i>.</p>
	<p>Asosiasi antara aktor dan <i>usecase</i>, digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengindikasikan aliran data.</p>
	<p>Asosiasi antara aktor dan <i>usecase</i> yang menggunakan panah terbuka untuk mengindikasikan bila aktor berinteraksi secara pasif dengan sistem.</p>

(Sumber : Windu Gata dan Grace Gata ; 2013)


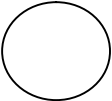

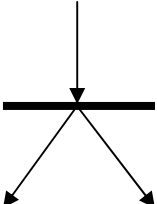
II.10.2 Class Diagram (Diagram Kelas)

Class Diagram merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem. *Class Diagram* juga menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas dan *constraint* yang berhubungan dengan objek yang dikoneksikan.

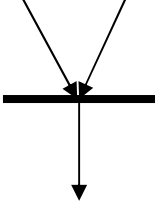
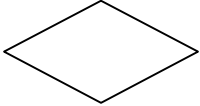
II.10.3 Activity Diagram (Diagram Aktivitas)

Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Simbol-simbol yang digunakan dalam *activity diagram*, yaitu:

Tabel II.2. Diagram Aktivitas

Gambar	Keterangan
	<i>Start Point</i> diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
	<i>End Point</i> , akhir aktifitas.
	<i>Activities</i> , menggambarkan suatu proses/kegiatan bisnis.
	<i>Fork</i> (percabangan), digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu.

Tabel II.2. Diagram Aktivitas Lanjutan

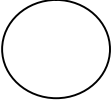
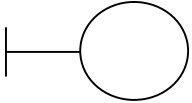
Gambar	Keterangan
	<i>Join</i> (penggabungan) atau <i>Rake</i> , digunakan untuk menunjukkan adanya dekomposisi.
	<i>Decision Points</i> , menggambarkan pilihan untuk pengambilan keputusan, <i>true</i> atau <i>false</i> .

(Sumber : Windu Gata dan Grace Gata ; 2013)

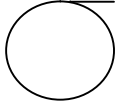

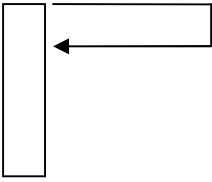
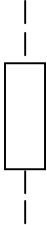

II.10.4 Sequence Diagram (Diagram Urutan)

Sequence Diagram menggambarkan kelakuan objek pada *usecase* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Simbol-simbol yang digunakan dalam *sequence diagram*, yaitu:

Tabel II.3. Diagram Urutan

Gambar	Keterangan
	<i>Entity Class</i> , merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.
	<i>Boundary Class</i> , berisi kumpulan kelas yang menjadi <i>interface</i> atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan <i>formentry</i> dan <i>form</i> cetak.

Tabel II.3. Diagram Urutan Lanjutan

Gambar	Keterangan
	<p><i>Control Class</i>, suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek. <i>Control object</i> mengkoordinir pesan antara <i>boundary</i> dengan entitas.</p>
	<p><i>Message</i>, simbol mengirim pesan antar <i>class</i>.</p>
	<p><i>Recursive</i>, menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.</p>
	<p><i>Activation</i>, mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.</p>
	<p><i>Lifeline</i>, garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i>.</p>

(Sumber : Windu Gata dan Grace Gata ; 2013)