

BAB II

TINJAUAN PUSTAKA

II.1. Sistem

Siklus hidup sistem adalah proses evolusioner yang diikuti dalam menerapkan sistem atau subsistem informasi berbasis komputer. Defenisi tentang sistem cukup banyak, untuk mengetahui lebih jelasnya tentang defenisi sistem ini diambil beberapa pernyataan dari beberapa ahli berikut ini:

Sistem adalah Entitas yang terdiri dari dua komponen atau lebih komponen atau subsistem yang berinteraksi untuk mencapai suatu tujuan. Perlengkapan dan program yang terdiri dari instalasi komputer lengkap. Program dan prosedur terkait yang menjalankan suatu tugas dalam sebuah komputer (**Marshall B. Romney, Dkk ; 2006 : 473**).

II.2. Informasi

Informasi adalah data yang sudah dibentuk ke dalam sebuah formulir bentuk yang bermanfaat dan dapat digunakan untuk manusia (**Marshall B. Romney, Dkk ; 2006 : 474**).

II.3. Sistem Informasi

Sistem Informasi adalah sekumpulan *hardware, software, brainware*, prosedur atau aturan yang diorganisasikan secara integral untuk mengolah data

menjadi informasi yang bermanfaat guna memecahkan masalah dan pengambilan keputusan.

Sistem Informasi adalah cara teratur untuk mengumpulkan, memproses, mengelola, dan melaporkan informasi agar organisasi dapat mencapai tujuan dan sasarannya (**Marshall B. Romney, Dkk ; 2006 : 473**).

Ada 2 (dua) jenis Sistem informasi yaitu:

1. Sistem informasi formal memiliki tanggung jawab jelas untuk memproduksi informasi.
2. sistem informasi informal adalah sistem yang muncul dari adanya kebutuhan yang tidak dipuaskan oleh saluran formal. Sistem ini berjalan tanpa adanya penugasan formal tanggung jawab.

II.4. Sistem Informasi Akuntansi

Sistem Informasi Akuntansi adalah sekumpulan *hardware, software, brainware*, prosedur atau aturan yang diorganisasikan secara integral untuk mengolah data menjadi informasi yang bermanfaat guna memecahkan masalah dan pengambilan keputusan (**Marshall B. Romney, Dkk ; 2006 : 473**).

Sistem Informasi Akuntansi (SIA) terdiri dari lima (5) komponen yaitu:

1. Orang-orang yang mengoperasikan sistem tersebut dan melaksanakan berbagai fungsi.
2. Prosedur-prosedur baik manual maupun yang terotomatisasi, yang dilibatkan dalam mengumpulkan, memproses dan menyimpan data tentang aktifitas-aktivitas organisasi.
3. Data tentang proses-proses bisnis organisasi.

4. *Software* yang dipakai untuk memproses dan organisasi.
5. Infrastruktur teknologi informasi, termasuk komputer, peralatan pendukung(*Peripheral device*) dan peralatan untuk komunikasi jaringan.

Kelima komponen ini secara bersama-sama memungkinkan suatu SIA memenuhi tiga fungsi pentingnya dalam organisasi, yaitu:

1. Mengumpulkan dan menyimpan data tentang aktivitas-aktivitas yang dilaksanakan oleh organisasi, sumber daya yang dipengaruhi oleh aktivitas-aktivitas tersebut, dan para pelaku yang terlibat dalam berbagai aktivitas tersebut, agar pihak manajemen, dan pihak-pihak luar yang berkepentingan dapat meninjau ulang (*review*) hal-hal yang telah terjadi.
2. Mengubah data menjadi informasi yang berguna bagi pihak manajemen untuk membuat keputusan dalam aktivitas perencanaan, pelaksanaan, dan pengawasan.
3. Menyediakan pengendalian yang memadai untuk menjaga aset-aset organisasi, termasuk data organisasi, untuk memastikan bahwa data tersebut tersedia dibutuhkan, akurat dan andal.

Mempelajari SIA adalah hal yang penting dalam akuntansi, Dalam *Statement Of Financial Accounting Concepts no.2, Financial Accounting Standards Board* mendefenisikan akuntansi sebagai sistem informasi. Di dalam standar akuntansi keuangan tersebut juga disebutkan bahwa tujuan utama akuntansi adalah untuk menyediakan informasi yang berguna bagi para para

pengambil keputusan. Oleh sebab itu, bukanlah hal yang mengherankan apabila *accounting education change commission* merekomendasikan bahwa kurikulum akuntansi harus menekankan bahwa akuntansi adalah suatu proses identifikasi, pengembangan, pengukuran dan komunikasi informasi. Komisi tersebut menyarankan agar kurikulum akuntansi harus dirancang untuk memberi para mahasiswa sebuah pemahaman yang kuat atas tiga komponen dasar berikut:

1. Pemakaian informasi didalam pengambilan keputusan
2. Sifat, desain, pemakaian dan implementasi SIA
3. Pelaporan informasi keuangan

II.5. Penjualan Konsinyasi

Penjualan Konsinyasi merupakan suatu perjanjian dimana salah satu pihak yang memiliki barang menyerahkan sejumlah barang kepada pihak tertentu untuk dijualkan dengan memberikan komisi (tertentu). (**Hadori Yunus, Dkk ; 2009 : 141**)

Dalam sistem penjualan konsinyasi pihak yang menyerahkan barang (pemilik) disebut *consignor* atau pengamanat. Sedangkan pihak yang menerima barang disebut *consignee* atau komisioner. Dari segi pengamat transaksi pengiriman-pengiriman barang kepada komisioner, biasa disebut sebagai “Barang-barang Konsinyasi” (*Consignment Out*). Sedang bagi komisioner untuk barang- barang yang diterimanya disebut sebagai “Barang-barang Komisi” (*Consignment in*).

Terdapat perbedaan prinsipal antara transaksi penjualan dengan transaksi konsinyasi, yaitu dalam hubungannya perpindahan hak milik atas barang-barang yang bersangkutan. Di dalam transaksi konsinyasi penyerahan barang dari pengamanat kepada komisioner tidak diikuti (tidak berarti) adanya penyerahan hak milik atas barang yang bersangkutan. Meskipun diakui bahwa dalam transaksi konsinyasi itu telah terjadi perpindahan (penyerahan) terhadap pengelolaan dan penyimpanan barang pada komisioner, namun demikian “hak milik” atas barang yang bersangkutan tetap berada pada pengamanat (*consignor*). Hak milik atas barang itu akan berpindah dari pengamanat apabila komisioner telah berhasil menjual barang tersebut kepada pihak ketiga.

Alasan-alasan bagi pengamant (*consigner*) untuk mengadakan perjanjian konsinyasi :

1. Konsinyasi merupakan suatu cara untuk lebih memperluas pasaran yang dapat dijamin oleh seorang produsen, pabrikan atau distributor, terutama apabila :
 - a. Barang- barang yang bersangkutan baru diperkenalkan, permintaan produk tidak tertentu dan belum terkenal.
 - b. Penjualan pada masa-masa yang lalu dengan melalui dealer tidak menguntungkan.
 - c. Harga barang menjadi mahal dan membutuhkan investasi yang cukup besar bagi pihak dealer apabila dia harus membeli barang-barang bersangkutan.

2. Resiko-resiko tertentu dapat dihindarkan oleh pengamanat. Barang-barang konsinyasi tidak ikut disita apabila terjadi kebangkrutan pada diri komisioner. Jadi lain sifatnya dengan perjanjian keagenan atau dealer.
3. Mungkin pengamanat ingin mendapatkan penjual khusus (*specialist*) dalam perdagangan barang-barangnya.
4. Harga eceran barang – barang yang bersangkutan tetap dapat dikontrol oleh pengamanat; demikian pula terhadap jumlah barang-barang yang siap dipasarkan dan stok barang tersebut.

Alasan – alasan Komisioner menerima perjanjian konsinyasi, antara lain :

1. Komisioner dilindungi dari kemungkinan resiko gagal untuk memasarkan barang- barang tersebut atau keharusan menjual dengan rugi.
2. Resiko rusaknya barang dan adanya fluktuasi harga dapat dihindarkan.
3. Kebutuhan akan modal kerja dapat dikurangi, sebab adanya barang-barang konsinyasi yang diterima atau dititipkan oleh pengamanat.

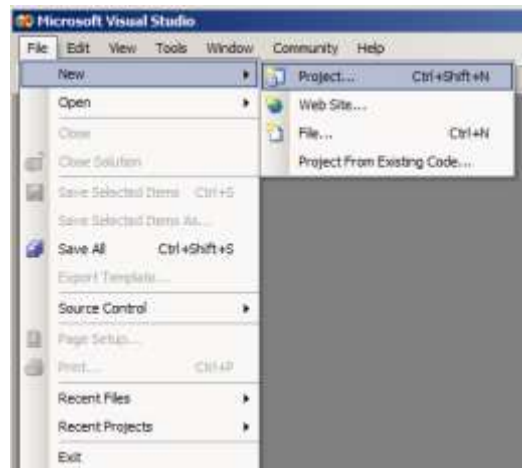
II.6. Visual Basic.Net 2008

Visual Basic, merupakan bahasa dan aturan pemrograman yang harus ditaati dalam menuliskan perintah-perintah agar program dapat dikompilasi.

Visual.Basic 2008 adalah Aplikasi IDE (Integrated Development Environment) yang digunakan untuk mengembangkan software. Di dalam aplikasi

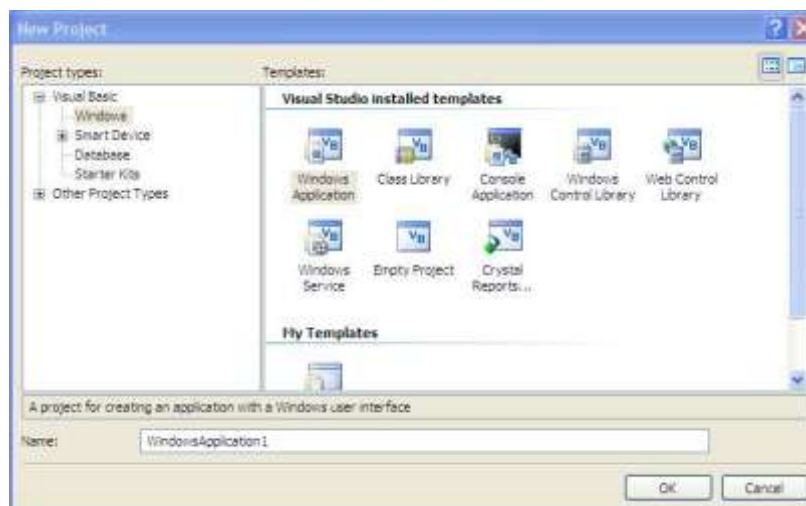
IDE inilah tersedia berbagai fitur yang memudahkan pemrograman, seperti kompilasi debugging, pengaturan proyek, mengedit antarmuka secara visual, dan lain-lain (**Rachmad Hakim S. ; 2009 : 2**).

Gambar dibawah ini merupakan contoh membuat aplikasi windows :



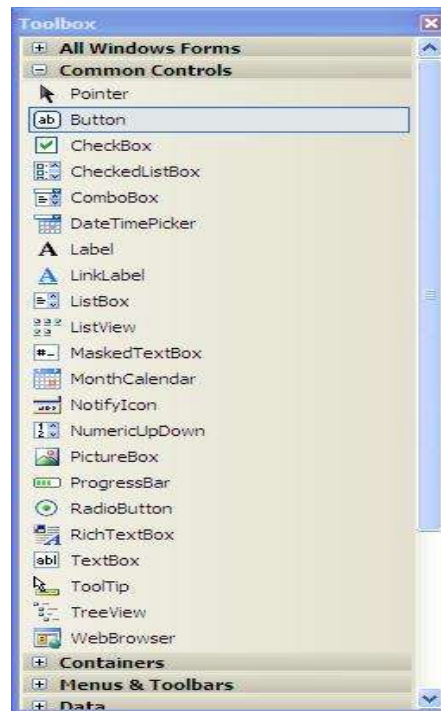
Gambar II.1. New Project

Sumber: (Rachmad Hakim S ; 2009 : 10)



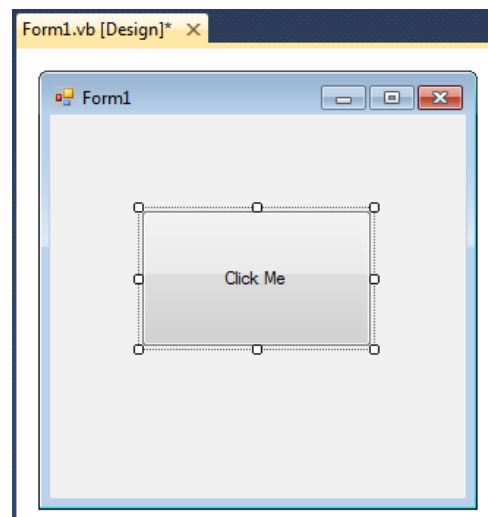
Gambar II.2. Windows Forms Application

Sumber : (Rachmad Hakim S ; 2009 : 10)



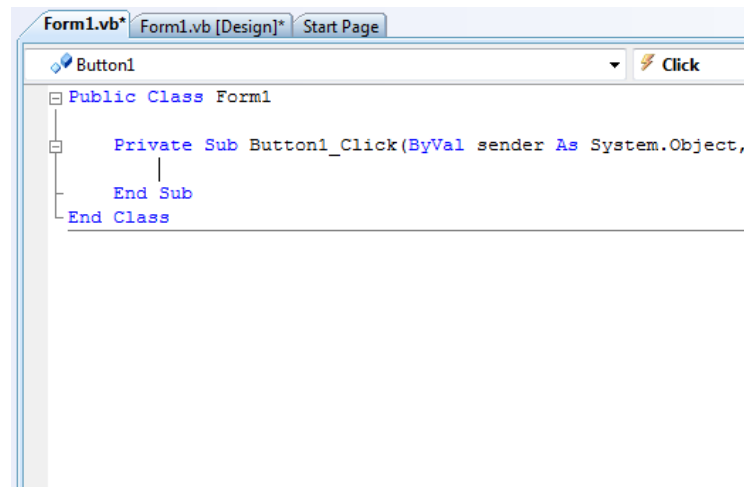
Gambar II.3. Toolbox

Sumber : (Rachmad Hakim S ; 2009 : 11)



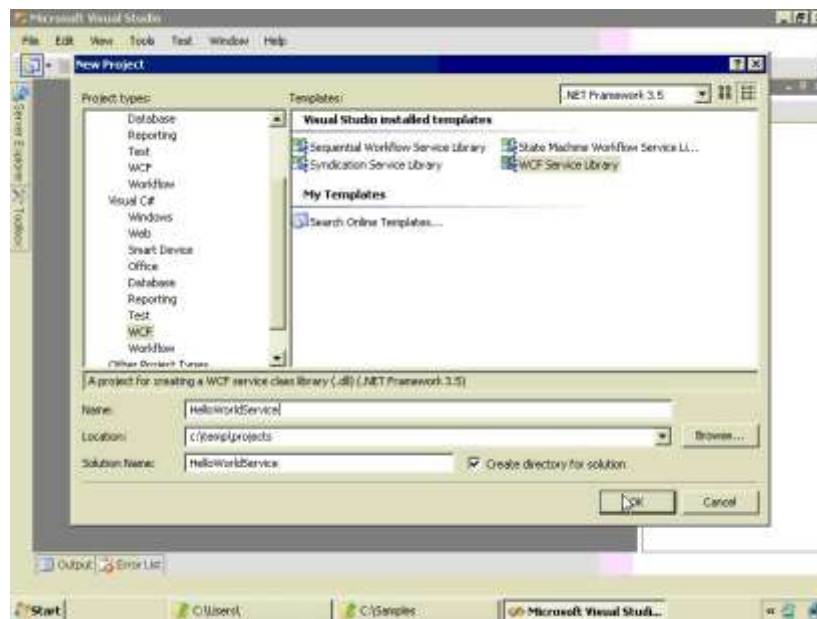
Gambar II.4. Desain

Sumber : (Rachmad Hakim S ; 2009 : 12)



Gambar II.5. Kode Program

Sumber : (Rachmad Hakim S ; 2009 : 13)



Gambar II.6. Lokasi Penyimpanan

Sumber : (Rachmad Hakim S ; 2009 : 14)

Pemrograman dengan Visual Basic dibuat dengan beberapa tahap berikut:

- Menuliskan kode program dengan bantuan aplikasi IDE. Artinya, Anda menuliskan program menggunakan aplikasi Microsoft Visual Basic 2008 Express Edition.
- Mengompilasi kode program tersebut menjadi program yang dapat dijalankan/dieksekusi. Hasil kompilasi adalah intruksi CIL (*Common Intermediate Language*) atau MSIL (*Microsoft Intermediate Language*) yang hanya dimengerti oleh kompiler JIT (*Just In Time*) dan tidak dapat dieksekusi langsung pada komputer.
- Penyebaran (distribusi) program di komputer dengan .Net Framework.

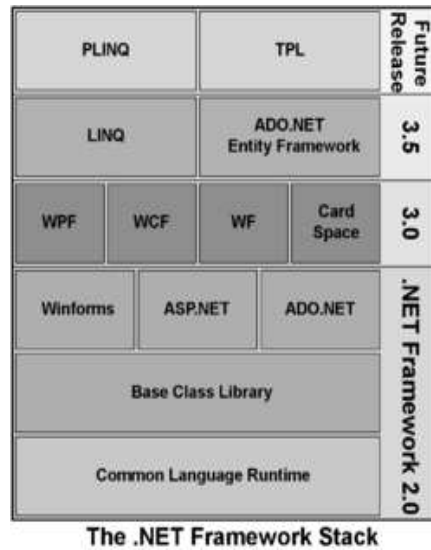
.NET Framework, merupakan *library* dan *virtual machine* yang terus berkembang mengikuti teknologi terbaru. Versi .NET Framework dimulai dari versi 1.0, 1.1, 2.0, 3.0 dan 3.5. Versi .NET yang biasanya dirilis dengan perbaikan serta dukungan terhadap teknologi baru sehingga semakin memudahkan pengembangan software.

.NET Framework, merupakan software kerangka kerja yang menghubungkan antara aplikasi .NET dengan sistem operasi, yang secara garis besar terdiri atas:

- *Library*, berisi kode-kode siap pakai dan banyak dibutuhkan oleh programmer.

- *Virtual machine*, berupa aplikasi yang digunakan untuk menjalankan program hasil kompi

Gambar dibawah ini merupakan bagian .NET Framework :



Gambar II.7. .NET Framework Stack

Sumber : (Rachmad Hakim S ; 2009 : 06)

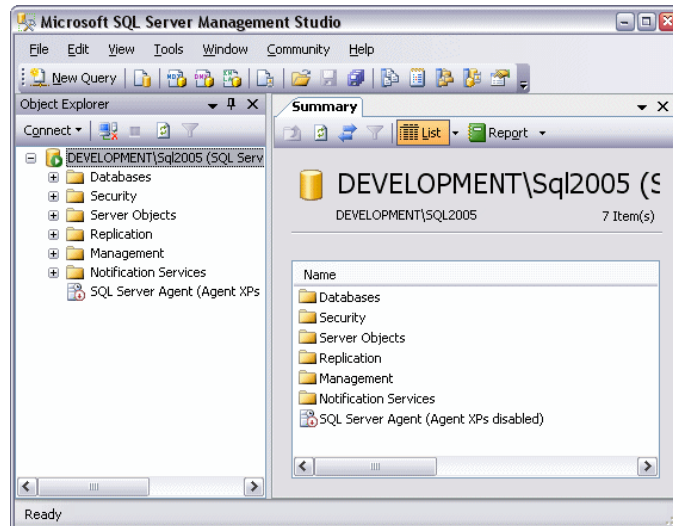
II.7. Pengertian SQL Server 2005

SQL Server 2005 adalah Server basis data yang secara fungsional adalah proses menyediakan layanan basis data. SQL Server 2005 telah mendefinisikan ulang sepenuhnya platform database SQL Server dan menyediakan fondasi yang kuat dimana organisasi-organisasi kecil, sedang, dan besar bisa membangun infrastruktur IT untuk generasi mendatang mereka. (William R. Stanek ; 2009 : 3)

Di dalam inti SQL server 2005, Anda akan menemukan:

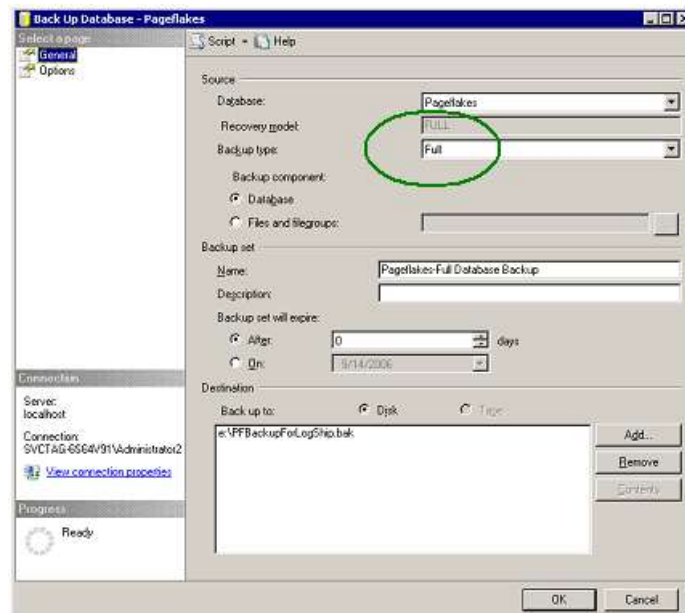
- *SQL Server Database Services* meliputi komponen-komponen database engine, adalah jantung dari Sql.
- *Analysis Services* Menyediakan fungsionalitas *Online Analytical Processing* (OLAP) *data mining* untuk aplikasi-aplikasi *business intelligence*.
- *Data Integration Services* Menyediakan sebuah transformasi data solusi integrasi kelas enterprise untuk mengekstrak dan mentransformasikan data dari berbagai sumber data dan memindahkannya ke satu atau lebih tujuan.
- *Notification Services* meliputi sebuah mesin komponen-komponen klien nontifikasi untuk menghasilkan dan mengirimkan pesan-pesan yang dipersonalisasi pada waktu yang tepat padapengguna-pengguna.
- *Reporting Services* Meliputi Report Manager dan Report Server untuk menyediakan sebuah platform berbasis server yang lengkap untuk menghasilkan, mengelola, dan mendistribusikan laporan-laporan.
- *Service Broker* Menyediakan *Quening* dan *Messaging* sebagai suatu bagian inti dari database.

Gambar dibawah ini merupakan bagian SQL Server 2005 :



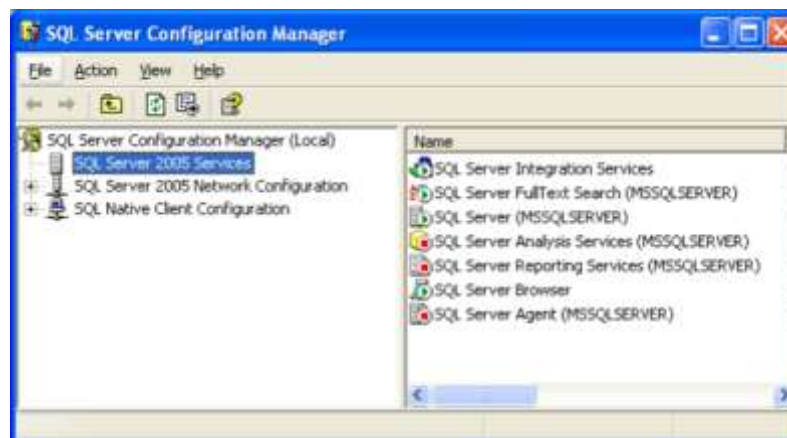
Gambar II.8. Tampilan SQL Server Management Studio

Sumber : (William R. Stanek ; 2009 : 16)



Gambar II.9. Opsi Bar

Sumber : Willam R. Stanek ; 2009 : 17)



Gambar II.10. Konfigurasi Jaringan

Sumber : (William R. Stanek ; 2009 : 18)

II.8. Pengertian UML (Unified Modeling Languages)

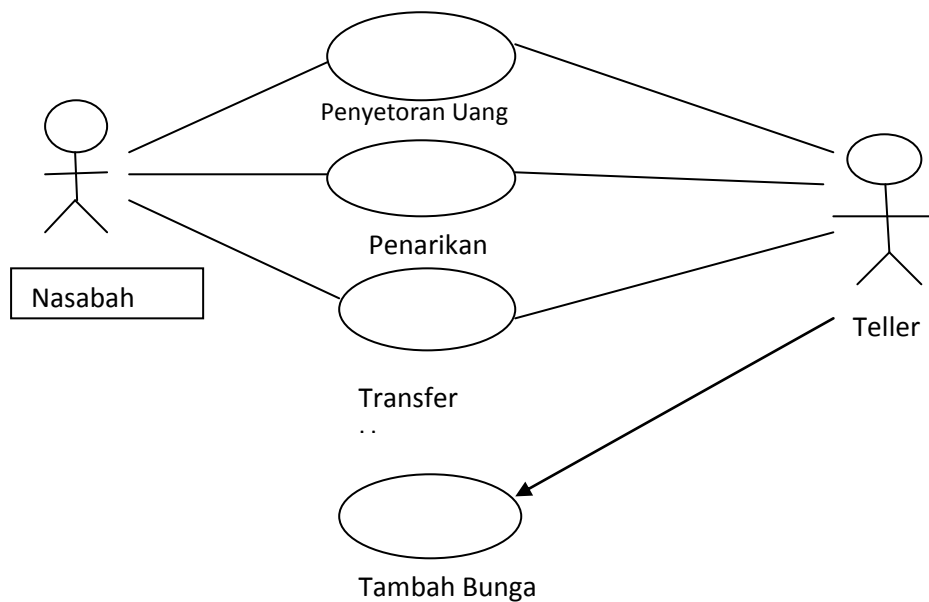
Unified Modeling Languages yang berarti bahasa pemodelan standar. (Chonoles, 2003 : bab I) mengatakan sebagai bahasa, berarti UML memiliki sintaks dan semantik. Ketika kita membuat model menggunakan konsep UML ada aturan-aturan yang harus diikuti. UML bukan hanya sekedar diagram, tetapi juga menceritakan konteks (**Prabowo Pudjo Widodo, Herlawati ; 2011 : 6**)

Salah satu kontributor terhadap diagram use case dalam UML adalah Ivar Jacobsen. Use Case menggambarkan External view dari sistem yang akan kita buat modelnya (Pooley, 2003 : 15) mengatakan bahwa model use case dapat dijabarkan dalam diagram use case, tetapi yang perlu di ingat, diagram tidak identik dengan model karena model lebih luas dari diagram.

Ada tiga(3)Komponen pembentukan digram Use case adalah:

- Aktor (*actor*), menggambarkan pihak-pihak yang berperan dalam sistem.
- Use Case*, aktivitas/sarana yang disiapkan oleh bisnis/sistem.
- Hubungan (*Link*), aktor mana saja yang terlibat dalam use case ini.

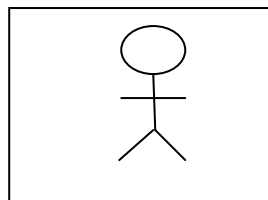
Gambar dibawah ini merupakan salah satu contoh bentuk diagram use case.



Gambar II.11. Diagram Use Case

Sumber : (Prabowo Pudjo Widodo ; 2011 : 17)

II.8.1. Aktor



Gambar II.12 Simbol Aktor

Sumber : (Prabowo Pudjo Widodo ; 2011 : 17)

Gambar II.11 memperlihatkan diagram use case dengan dua aktor (nasabah dan teller) dan empat use case (Penyetoran uang, penarikan uang, transfer uang dan tambah bunga). Simbol aktor adalah gambar orang.

(Chonoles, 2003 : bab 8) menyarankan sebelum membuat use case dan menentukan aktornya, agar mengidentifikasi siapa saja pihak yang terlibat dalam sistem kita. Pihak yang terlibat biasanya dinamakan *Stakeholder*. Langkah awal yang baik adalah mempertimbangkan kebutuhan klien dan pelanggan sebelum menentukan use case. Setiap klien dan pelanggan sebelum membentuk use case. Setiap sistem memiliki *stakeholder* potensial yang harus dipertimbangkan karena berpengaruh terhadap kinerja bisnis/sistem tersebut.

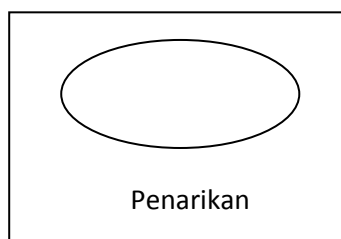
Menurut (Whitten, 2004 : 259) ada empat macam tipe aktor:

1. *Primary business actor* (Aktor bisnis utama) yaitu stakeholder yang terutama mendapatkan keuntungan dari pelaksanaan use case dengan menerima nilai yang terukur atau terobservasi.
2. *Primary sistem actor* (Aktor sistem utama) yaitu stakeholder yang secara langsung berhadapan dengan sistem untuk menginisiasi atau memicu kegiatan atau sistem.
3. *External server actor* (Aktor server eksternal) yaitu stakeholderyang melayani kebutuhan pengguna use case (misalnya biro kredit yang memiliki kuasa atau perubahan kartu kredit).
4. *External receiving actor* (Aktor penerima eksternal) yaitu stakeholder yang bukan pelaku utama, tapi menerima nilai yang terukur atau teramati (*output*) dari *use case*.

II.8.2. Use Case

Menurut (Pilone, 2005 : bab 7.1) use case menggambarkan fungsi tertentu dalam suatu sistem berupa komponen, kejadian atau kelas. Sedangkan (Whitten, 2004 : 258) mengartikan use case sebagai urutan langkah-langkah yang secara tindakan saling terkait (skenario), baik terotomatisasi maupun secara manual, untuk tujuan melengkapi satu tugas bisnis tunggal.

Use Case digambarkan dalam bentuk Elips/Oval.



Gambar II.3. Simbol Use Case

Sumber : (Prabowo Pudjo Widodo ; 2011 : 22)

Use case sangat menentukan karakteristik sistem yang kita buat, oleh karena itu (Chonoles, 2003 : bab 8) menawarkan cara untuk menghasilkan use case yang baik, yaitu :

- Pilihlah nama yang baik. *Use case* adalah sebuah behaviour (Perilaku), jadi seharusnya dalam frase kata kerja.
- Ilustrasikan perilaku dengan lengkap. *Use case* dimulai dari inisiasi oleh aktor primer dan berakhir pada aktor dan menghasilkan tujuan.
- Identifikasi perilaku dengan lengkap. Untuk mencapai tujuan dan menghasilkan tujuan dan menghasilkan nilai tertentu dari aktor, *Use case* harus lengkap.

- Menyediakan Use case lawan (*inverse*). Kita biasanya membutuhkan use case yang membatalkan tujuan, misalnya pada use case pemesanan kamar, dibutuhkan pula use case pembatalan pesanan kamar.
- Batasi *Use case* hingga satu perilaku saja.
- Nyatakan *Use case* dari sudut pandang aktor. Tulislah *Use case* dari sudut pandang aktor bukan dari sistem.