

BAB II

TINJAUAN PUSTAKA

II.1. Sistem

Sistem merupakan suatu bentuk *integrasi* antara satu komponen dengan komponen yang lain karena sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi di dalam sistem tersebut. Oleh karena itu sistem dapat diklasifikasikan dari beberapa sudut pandang, seperti contoh sistem yang bersifat abstrak, sistem alamiah, sistem yang bersifat *deterministik* dan sistem yang bersifat terbuka dan tertutup.

1. Sistem Abstrak dan Sistem Fisik

Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik, misalnya sistem teologia. Sedangkan sistem fisik merupakan sistem yang ada secara fisik seperti sistem komputer, sistem produksi, dan lain sebagainya.

2. Sistem Alamiah dan Sistem Buatan Manusia

Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia, misalnya sistem perputaran bumi, terjadinya siang dan malam, pergantian musim. Sedangkan sistem buatan manusia merupakan sistem yang melibatkan hubungan manusia dengan mesin, yang disebut dengan *human machine system*. Sistem informasi berbasis komputer merupakan contohnya, karena menyangkut penggunaan komputer yang berinteraksi dengan manusia.

3. Sistem Deterministik dan Sistem Probabilistik

Sistem yang beroperasi dengan tingkah laku yang dapat diprediksi disebut sistem *deterministik*. Sedangkan sistem yang bersifat *probabilistik* adalah sistem yang kondisi masa depannya tidak dapat diprediksi, karena mengandung unsur *probabilitas* (Tata Sutabri ; 2012 : 25).

II.2. Informasi

Sistem informasi manajemen berhubungan dengan informasi. Berapa banyak informasi yang diberikan oleh sebuah sistem informasi ? belum ada metode untuk mengukur informasi dalam sebuah sistem untuk menghitung isinya. Informasi adalah sebuah istilah yang tepat dalam pemakaian umum. Informasi dapat mengenai data mentah, data tersusun, kapasitas sebuah saluran komunikasi , dan lain sebagainya. Informasi ibarat darah yang mengalir dalam tubuh suatu organisasi sehingga informasi ini sangat penting di dalam suatu organisasi. Suatu sistem yang kurang mendapatkan informasi akan menjadi luruh, kerdil, dan akhirnya mati.

Belum ada metode untuk mengukur informasi dalam sebuah sistem dan kerumitan informasi tidak memungkinkan adanya suatu rumus atau *algoritma* untuk menghitung isinya. Informasi adalah sebuah istilah yang tidak tepat dalam pemakaiannya secara umum. Informasi dapat mengenai data mentah, data tersusun, kapasitas sebuah saluran komunikasi, dan lain sebagainya. Suatu sistem yang kurang mendapatkan informasi akan menjadi luruh, kerdil, dan akhirnya mati. Sistem informasi manajemen berhubungan dengan informasi. Informasi

adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan. Sistem pengolahan informasi akan mengolah data menjadi informasi atau mengolah data dari bentuk tak berguna menjadi bentuk berguna bagi yang menerimanya. Nilai informasi berhubungan dengan keputusan (Tata Sutabri ; 2012 : 29).

II.3. Sistem Informasi Akuntansi

Organisasi tergantung pada sistem informasi untuk dapat berdaya saing. Informasi juga merupakan sumber daya, sama seperti pada pabrik dan peralatan. Produktivitas, sebagai faktor yang penting untuk mempertahankan daya saing perusahaan, dapat ditingkatkan dengan sistem informasi yang lebih baik. Akuntansi, sebagai suatu sistem informasi, mengidentifikasi, mengumpulkan, memproses dan mengkomunikasikan informasi ekonomi mengenai suatu entitas ke berbagai kelompok orang. Informasi merupakan suatu data yang diorganisasi yang dapat mendukung ketepatan pengambilan keputusan. Sistem merupakan sekumpulan sumber daya yang dapat terkait untuk mencapai suatu tujuan. (George H. Bodnar;2007:3)

SIA (Sistem Informasi Akuntansi) merupakan kumpulan sumber daya, seperti manusia dan peralatan, yang dirancang untuk mengubah data keuangan dan data lainnya ke dalam informasi. Informasi tersebut dikomunikasikan kepada para pembuat keputusan. Sistem informasi akuntansi melakukan hal tersebut dengan sistem manual atau melalui sistem terkomputerisasi. (George H. Bodnar;2007:3)

Analog dengan defenisi sebelumnya, SIA (Sistem Informasi Akuntansi) adalah sistem berbasis computer yang dirancang untuk mentransformasi data akuntansi menjadi infomasi. (George H. Bodnar; 2007: 8)

II.4. Visual Basic

Visual basic merupakan salah satu bahasa pemograman yang andal dan banyak digunakan oleh pengembang untukmembangun berbagai macam aplikasi *windows*. *Visual basic 2008* merupakan aplikasi pemograman yang menggunakan teknologi. *NET Framework 3.5*. Teknologi. *NET Framework 3.5* merupakan komponen *windows* yang terintegrasi serta mendukung pembuatan, penggunaan aplikasi, dan halaman *web*. Teknologi *.Net Framework 3.5* mempunyai 2 komponen utama, yaitu *CLR (Common Language Runtime)* dan *Class Library*. *CLR* digunakan untuk menjalankan aplikasi yang berbasis.*NET*, sedangkan *Library* adalah kelas pustaka atau perintah yang digunakan untuk membangun aplikasi (Wahana Komputer ; 2010 : 2)

II.5. Microsoft SQL Server

Bahasa query merupakan bahasa khusu yang digunakan untuk melakukan manipulasi dan menanyakan pertanyaan (query) yang berhubungan dengan bahasa pemrograman, dimana bahasa query tidak memiliki kemampuan untuk menyelesaikan banyak masalah seperti bahasa pemrograman pada umumnya. Dalam pemrograman basis data, salah satu bahasa yang harus kita kuasai adalah SQL. SQL merupakan bahasa komputer standar yang digunakan untuk

berkomunikasi dengan sistem manajemen basis data relasional (RDBMS) (Ema Utami dan Anggi Dwi Hartanto ; 2012 : 63)

II.6. Kamus Data

Kamus data adalah suatu file yang terpisah yang menyimpan informasi seperti :

1. Nama setiap *item*/jenis/kolom data
2. Struktur data untuk tiap *item*
3. Program yang menggunakan tiap *item*
4. Tingkat keamanan untuk setiap *item* (Zulkifli Amsyah : 2008 : 17)

Sebelum memperoleh defenisi formal basis data, kita akan mencoba memahaminya secara sederhana terlebih dahulu. Istilah basis data tersusun atas dua suku kata, yaitu basis dan data (basis data = basis + data). Dalam sistem bilangan biner, kita dapat menuliskan beberapa contoh bilangan sebagai berikut.

0 → sama dengan 0 dalam sistem bilangan desimal

1 → sama dengan 1 dalam sistem bilangan desimal

10 → sama dengan 2 dalam sistem bilangan desimal

11 → sama dengan 3 dalam sistem bilangan desimal

100 → sama dengan 4 dalam sistem bilangan desimal (Edhy Sutanta ; 2011 : 25)

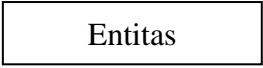
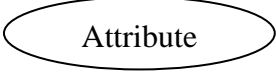


II.7. Entity Relationship Diagram

Entity Relationship Diagram/ER_M merupakan suatu model data yang dikembangkan berdasarkan objek. ER_M digunakan untuk menjelaskan hubungan antara data dalam basis data kepada pengguna secara logik. ER_M didasarkan

pada suatu persepsi bahwa *real world* terdiri atas objek-objek dasar yang mempunyai hubungan/kerelasian antar objek-objek dasar tersebut. ER_M digambarkan dalam bentuk diagram yang disebut dengan ER (*ER_Diagram/ER_D*). Untuk menggambarkan ER_D digunakan simbol-simbol grafis tertentu (Edhy Sutanta ; 2011 : 91)

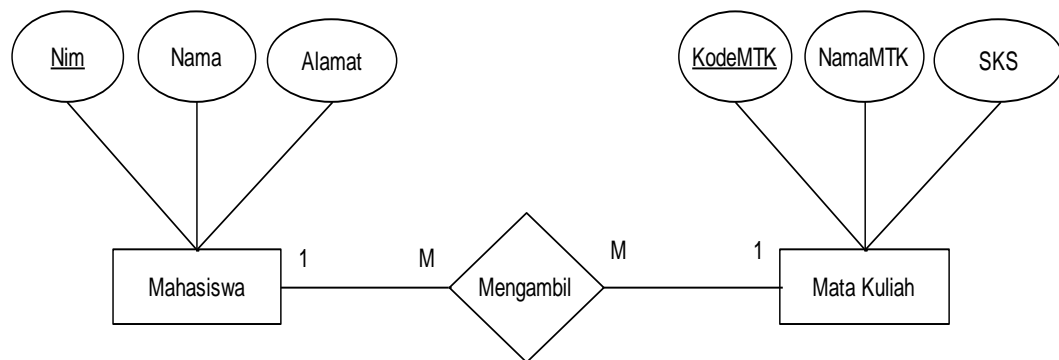
Struktur *logis* (skema *database* dapat ditunjukkan secara *grafis* dengan ER yang dibentuk dari komponen-komponen berikut ini (Rosa A.S dan M. Shalahuddin ; 2011 : 60)

Tabel II. 1. Komponen ERD

Simbol	Keterangan
	Persegi panjang mewakili kumpulan entitas.
	Elips mewakili attribute
	Belah ketupat mewakili relasi
	Garis menghubungkan atribut dengan kumpulan entitas dengan relasi.

(Sumber : Rosa A.S dan M. Shalahuddin ; 2011)

Sebagai ilustrasi, bayangan anda mengambil bagian sistem basis data universitas yang terdiri dari mahasiswa dan mata kuliah. Gambar II.1. menunjukkan ER Diagram dari contoh. Diagram menunjukkan bahwa ada dua kumpulan entitas, yaitu mahasiswa dan mata kuliah, dan bahwa relasi mengambil mahasiswa dan mata kuliah.



Gambar II.1. Diagram ER
(Sumber : Rosa A.S dan M. Shalahuddin ; 2011)

II.8. Normalisasi

Normalisasi diartikan sebagai suatu teknik yang menstrukturkan atau mendekomposisikan data dalam cara-cara tertentu untuk mencegah timbulnya permasalahan dalam pengolahan data dalam basis data. Permasalahan yang dimaksud adalah berkaitan dengan penyimpangan-penyimpangan (*anomalies*) yang terjadi akibat adanya kerangkapan data dalam relasi dan in-efisiensi pengolahan. Proses normalisasi menghasilkan relasi yang optimal yaitu :

1. Memiliki struktur record yang konsisten secara logik
2. Memiliki struktur record yang mudah untuk dimengerti
3. Memiliki struktur record yang sederhana dalam pemeliharaan
4. Memiliki struktur record yang mudah ditampilkan kembali untuk memenuhi kebutuhan pengguna
5. Minimalisasi kerangkapan data guna meningkatkan kinerja sistem (Edhy Sutanta ; 2011 : 174).

Tahapan normalisasi terdiri dari beberapa bentuk, yaitu:

1. Bentuk Tak Normal (UNF / *Un Normal Form*).

2. Bentuk Normal Pertama (1 NF / *First Normal Form*).

Bentuk Normal pertama memiliki ciri: Data berbentuk *file-file* (*file* datar), *record* disusun sesuai kedatangan, masih mungkin terjadi penyimpangan data (anomali data). Anomali data dapat berupa insert *anomali*, *delete anomali*, *update anomali*, dan *redundancy data* (data duplikat).

3. Bentuk Normal Kedua (2 NF / *Second Normal Form*).

Bentuk Normal kedua memiliki ciri; Tidak terjadi anomali data, setiap *field*/atribut bukan kunci harus tergantung fungsi (*Functional Dependency*) terhadap *field*/atribut kunci, masih mungkin terjadi *transitive dependency* (*field* bukan kunci tergantung pada *field* bukan kunci dalam satu *table*). Model objek mencapai bentuk normal kedua, sehingga penulis mendesain mulai bentuk normal ketiga dan bentuk normal boyce codd. Sedangkan untuk bentuk tak normal sudah dari dokumen dasar berupa Faktur, Nota, dan laporan *Stock of Name*.

4. Bentuk Normal Ketiga (3 NF / *Third Normal Form*).

Table yang memenuhi Bentuk Normal Ketiga harus tidak terdapat *Transitive Dependency*. Bentuk normal ketiga dari sistem *inventory* :

5. Bentuk Normal *Boyce Codd* (BCNF / *Boyce Codd Normal Form*).

Karena tak ada *field* bukan kunci tergantung secara parsial (bagian) kunci dalam satu tabel, maka bentuk normal ketiga juga merupakan bentuk BCNF.

6. Normal yang lebih tinggi.

Setelah 3NF, semua masalah normalisasi hanya melibatkan tabel yang mempunyai tiga kolom atau lebih dan semua kolom adalah kunci. Bentuk Normal Boyce-Codd (BCNF) adalah versi 3NF yang lebih teliti dan berhubungan dengan tabel relasional yang mempunyai banyak kunci kandidat, kunci kandidat gabungan, dan kunci kandidat yang saling tumpang tindih (Yuniar Supardi ; 2008 : 11)

Tabel II.2. Normalisasi

Pelanggan

*Kode Pelanggan	Nama_Pelanggan	Alt_Pelanggan	Tlp_Pelanggan

Transj

*No_Faktur	Qty_Jual	Harga_Jual	**Kode_Barang

Faktur

*No_Faktur	Tgl_Jual	Total_Jual	Pembuat	Penerima	**Kode_Pelanggan	**No_SO

Barang

*No_Barang	Nama_Barang	Harga_Beli	Harga_Jual	Quantity

(Sumber : Yuniar Supardi ; 2008)

II.9. Unified Modelling Language

UML singkatan dari *Unified Modelling Language* yang berarti bahasa permodelan standar. UML diaplikasikan untuk maksud tertentu, biasanya antara lain untuk :

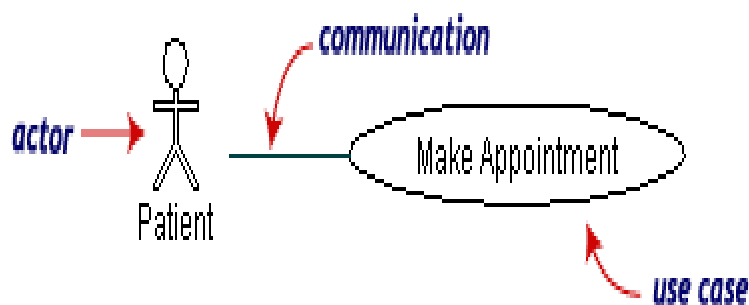
1. Merancang perangkat lunak
2. Sarana Komunikasi antara perangkat lunak dengan proses bisnis.

3. Menjabarkan sistem secara rinci untuk analisa dan mencari apa yang diperlukan sistem.
4. Mendokumentasi sistem yang ada, proses-proses dan organisasinya.
(Prabowo Pudjo Widodo dan Herlawati ; 2011 : 6-7)

Unified Modelling Language (UML) adalah sebuah “bahasa” yang telah menjadi standar dalam industri untuk visualisasi Setiap sistem yang kompleks seharusnya bisa dipandang dari sudut yang berbeda – beda sehingga bisa mendapatkan pemahaman secara menyeluruh . Untuk upaya tersebut UML menyediakan 9 jenis diagram yang dapat dikelompokkan berdasarkan sifatnya statis atau dinamis. Ke 9 diagram dalam UML itu adalah (Prastuti Sulistyorini 2009)

II.9.1. Use Case Diagram

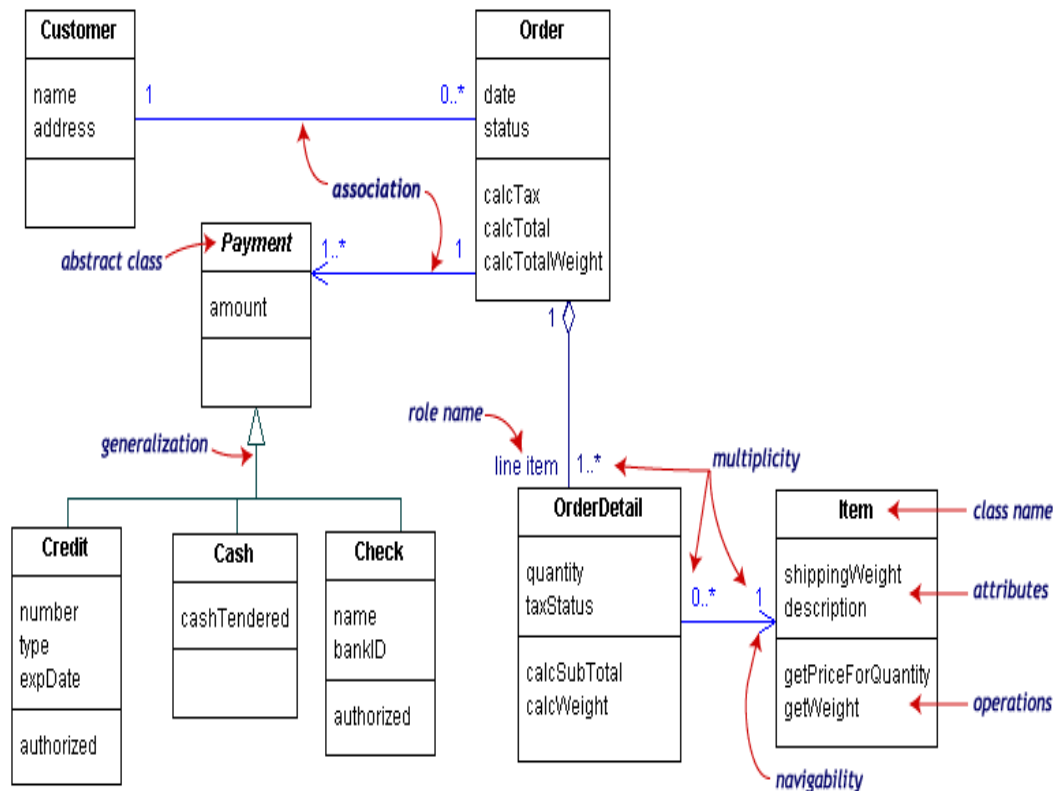
Diagram ini bersifat statis. Diagram ini memperlihatkan himpunan use case dan aktor-aktor (suatu jenis khusus dari kelas). Diagram ini terutama sangat penting untuk mengorganisasi dan memodelkan perilaku dari suatu sistem yang dibutuhkan serta diharapkan pengguna (Prastuti Sulistyorini : 2009).



Gambar II.2. Contoh Use Case Diagram
(Sumber : Prastuti Sulistyorini : 2009)

II.9.2. Class Diagram

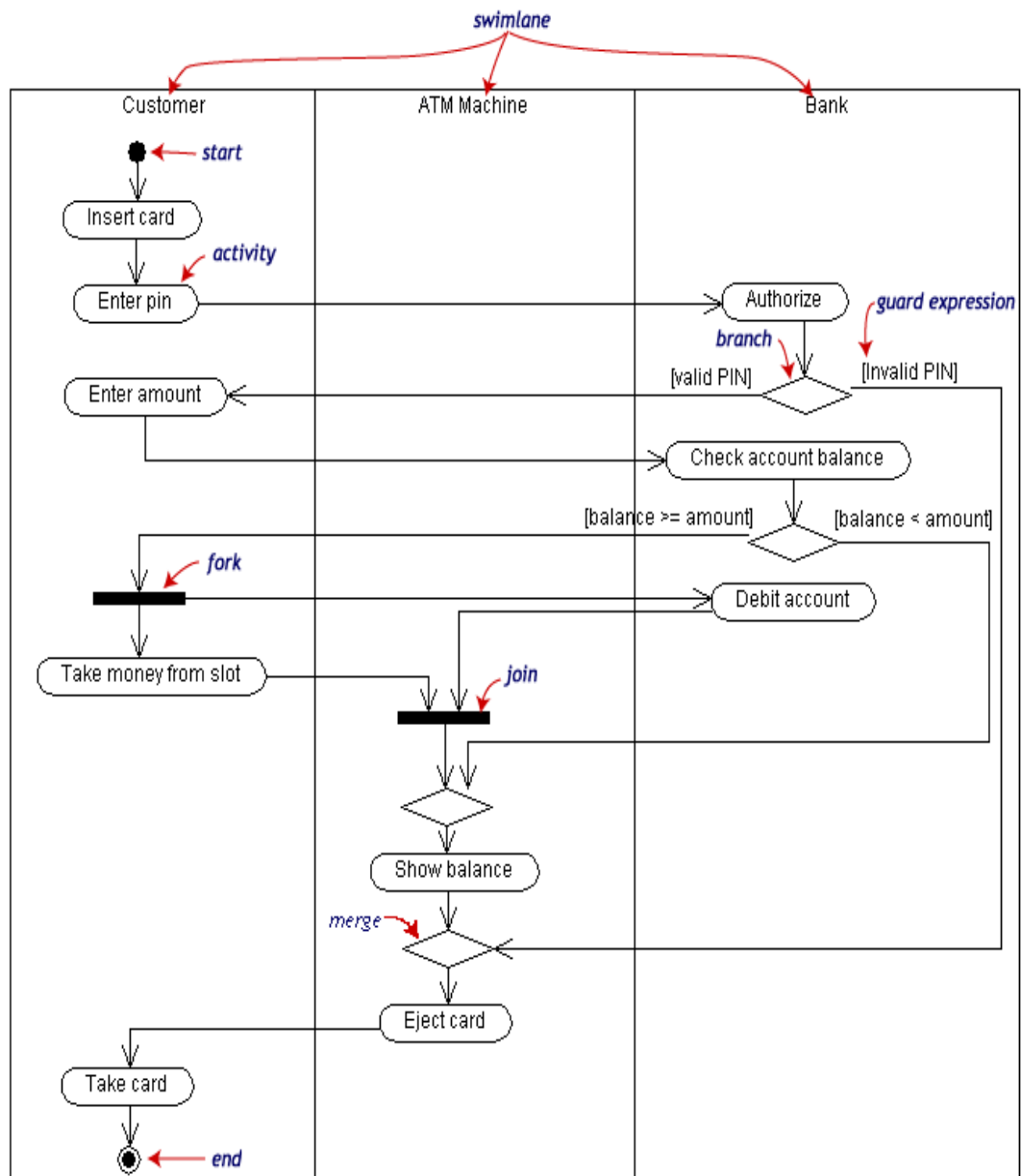
Diagram kelas bersifat statis. Diagram ini memperlihatkan himpunan kelas-kelas, antarmuka antarmuka, kolaborasi-kolaborasi serta relasi (Prastuti Sulistyorini : 2009).



Gambar II.3. Contoh Class Diagram
(Sumber : Prastuti Sulistyorini : 2009)

II.9.3. Activity Diagram

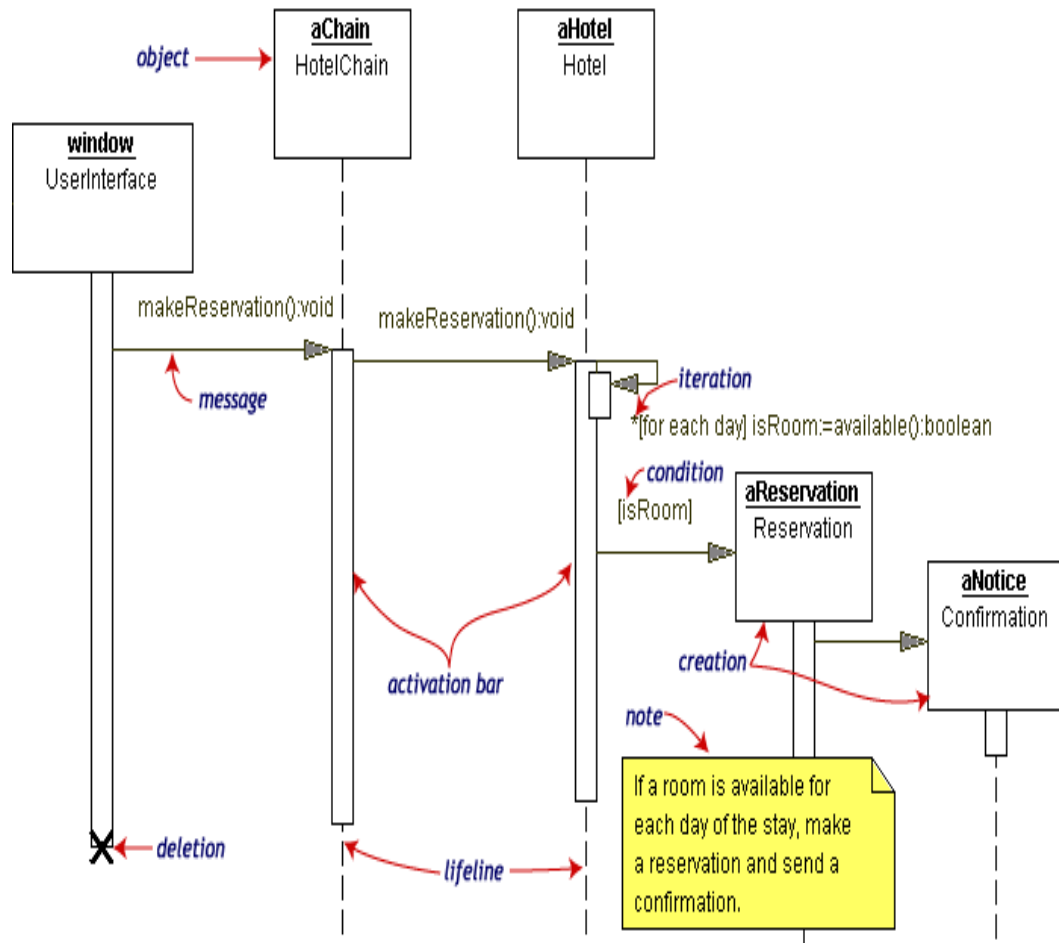
Diagram ini bersifat dinamis. Diagram ini adalah tipe khusus dari diagram state yang memperlihatkan aliran dari suatu aktifitas ke aktifitas lainnya dari suatu sistem. Diagram ini terutama penting dalam pemodelan fungsi – fungsi dalam suatu sistem dan memberi tekanan pada aliran kendali antar objek (Prastuti Sulistyorini : 2009).



Gambar II.4. Contoh Activity Diagram
(Sumber : Prastuti Sulistyorini : 2009)

II.9.4. Sequence Diagram

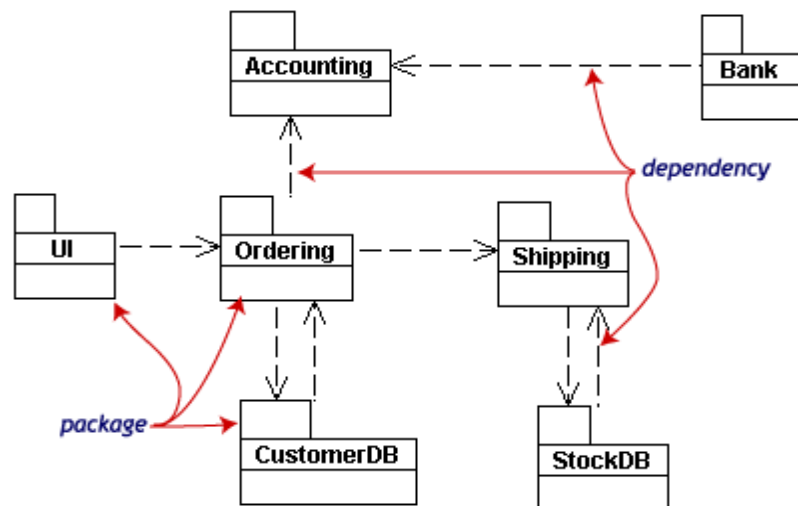
Diagram ini bersifat dinamis. Diagram sequence merupakan diagram interaksi yang menekankan pada pengiriman pesan (message) dalam suatu waktu tertentu (Prastuti Sulistyorini : 2009).



Gambar II.5. Contoh Sequence Diagram
(Sumber : Prastuti Sulistyorini : 2009)

II.9.5. Diagram Package Object

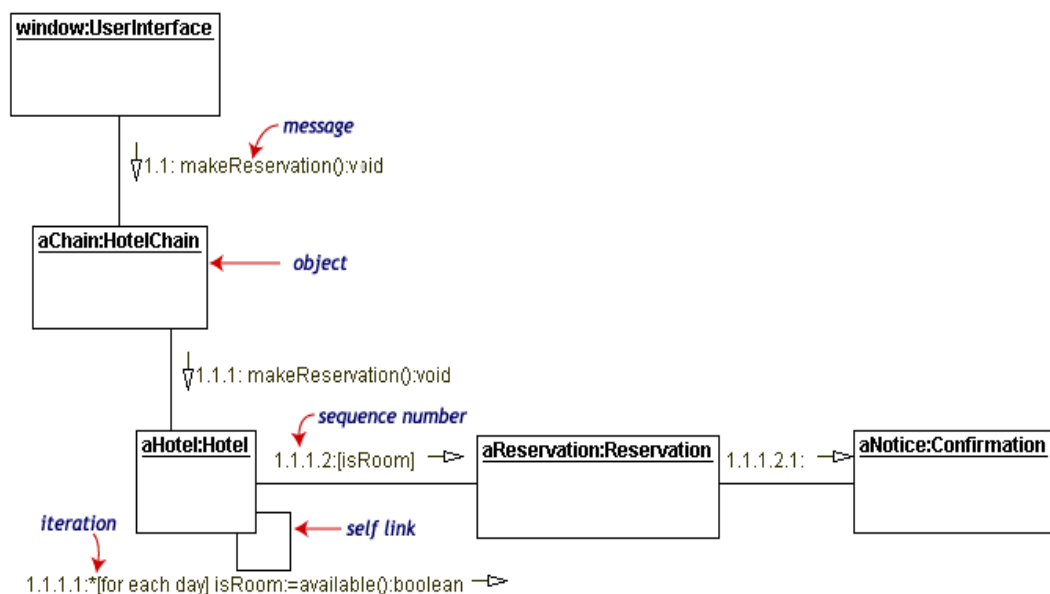
Diagram objek bersifat statis. Diagram ini memperlihatkan objek-objek serta relasi antar objek. Diagram objek memperlihatkan instansiasi statis dari segala sesuatu yang dijumpai pada diagram kelas (Prastuti Sulistyorini : 2009).



Gambar II.6. Contoh Package Object Diagram
(Sumber : Prastuti Sulistyorini : 2009)

II.9.6. Collaboration Diagram

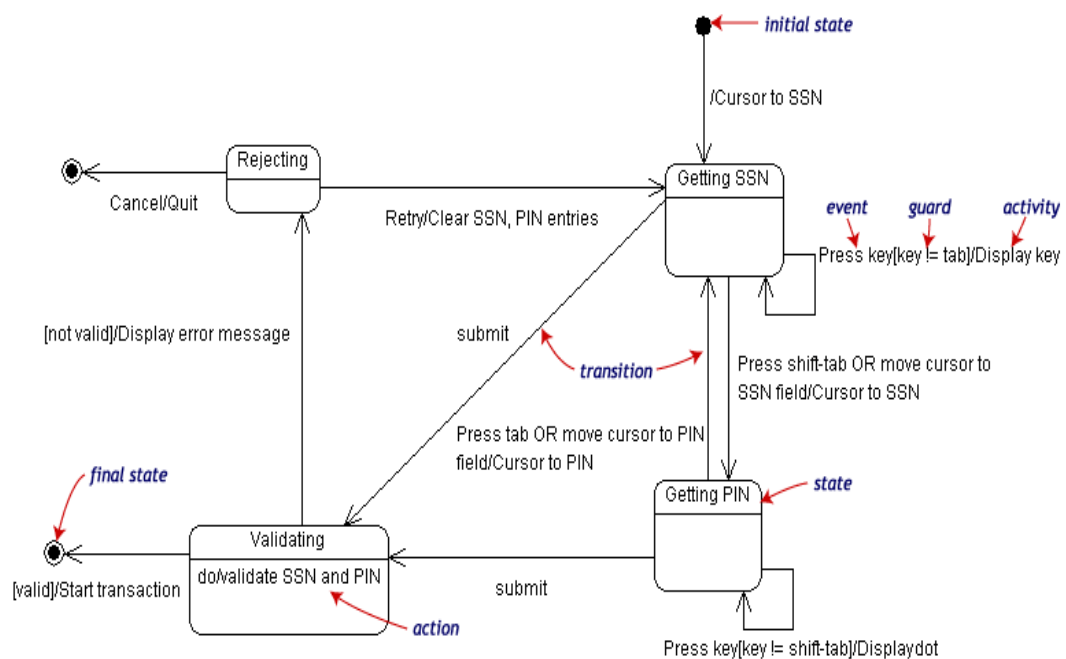
Diagram ini bersifat dinamis. Diagram kolaborasi adalah diagram interaksi yang menekankan organisasi struktural dari objek – objek yang menerima serta mengirim pesan (message) (Prastuti Sulistyorini : 2009).



Gambar II.7. Contoh Collaboration Diagram
(Sumber : Prastuti Sulistyorini : 2009)

II.9.7. Statechart Diagram

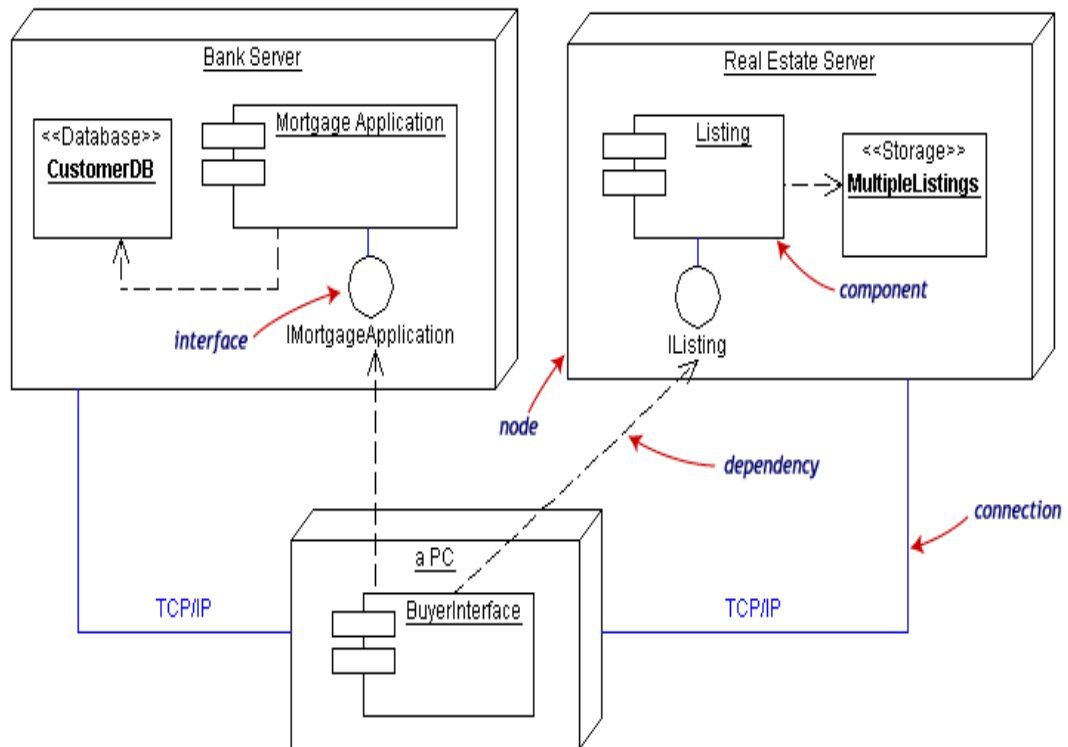
Diagram ini bersifat dinamis. Diagram ini memperlihatkan state – state pada sistem, memuat state, transisi, event, serta aktifitas. Diagram ini terutama penting untuk memperlihatkan sifat dinamis dari antarmuka, kelas, kolaborasi dan terutama penting pada pemodelan sistem – sistem yang reaktif (Prastuti Sulistyorini : 2009).



Gambar II.8. Contoh Statechart Diagram
(Sumber : Prastuti Sulistyorini : 2009)

II.9.8. Component dan Deployment Diagram

Diagram ini bersifat statis. Diagram ini memperlihatkan organisasi serta ketergantungan pada komponen – komponen yang telah ada sebelumnya. Diagram ini berhubungan dengan diagram kelas dimana komponen secara tipikal dipetakan ke dalam satu atau lebih kelas, antarmuka – antarmuka serta kolaborasi – kolaborasi (Prastuti Sulistyorini : 2009).



Gambar II.9. Contoh *Component* dan *Deployment Diagram*
 (Sumber : Prastuti Sulistyorini : 2009)

II. 10. Biaya Masuk Dan Biaya Keluar

Biaya adalah semua pengorbanan yang perlu dilakukan untuk suatu proses produksi, yang dinyatakan dengan satuan uang menurut harga pasar yang berlaku, baik yang sudah terjadi maupun yang akan terjadi. Biaya terbagi menjadi dua, yaitu biaya eksplisit dan biaya implisit. Biaya eksplisit adalah biaya yang terlihat secara fisik, misalnya berupa uang. Sementara itu, yang dimaksud dengan biaya implisit adalah biaya yang tidak terlihat secara langsung, misalnya biaya kesempatan dan penyusutan barang modal.

Dalam akuntansi, yang dimaksud dengan biaya adalah aliran sumberdaya yang dihitung dalam satuan moneter yang dikeluarkan untuk membeli atau

membayar persediaan, jasa, tenaga kerja, produk, peralatan, dan barang lainnya yang digunakan untuk keperluan bisnis atau kepentingan lainnya. Sementara biaya kesempatan merujuk pada setiap alternatif yang dikorbankan untuk melakukan pekerjaan lain yang lebih bernilai. Misalnya, seorang guru dibayar sebesar Rp 500.000,00 per bulan. Jika kemudian ia memutuskan untuk berhenti bekerja dan mencoba berwirausaha, maka ia akan kehilangan pekerjaannya sebagai guru dan mengorbankan *kesempatan* mendapatkan gaji Rp 500.000,00. Dapat disimpulkan bahwa—bagi guru itu—biaya kesempatan untuk menjadi wirausahawan adalah Rp 500.000,00 per bulan (Sulaiman Iskandar ; 2010 : 1).