

BAB II

TINJAUAN PUSTAKA

II.1. Sistem

Dengan berbagai pendekatan, beragam pula istilah “sistem” didefinisikan. Sistem adalah suatu pengorganisasian yang saling berinteraksi, saling bergantung dan terintegrasi dalam kesatuan variabel atau komponen. Terdapat dua kelompok pendekatan sistem, yaitu menekankan pada prosedur dan komponen atau elemennya. Pendekatan sistem yang lebih menekankan pada prosedur mendefinisikan sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkelompok dan bekerjasama untuk melakukan kegiatan pencapaian sasaran tertentu. Makna dari prosedur sendiri, yaitu urutan yang tepat dari tahapan-tahapan instruksi. Sedangkan pendekatan yang menekankan pada komponen mendefinisikan sistem sebagai kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. *“Serangkaian atau tatanan elemen-elemen yang diatur untuk mencapai tujuan yang ditentukan sebelumnya melalui pemrosesan informasi”* (Riyanto, dkk; 2009 : 22).

II.2. Siklus Informasi

Menurut Riyanto, dkk. (2009 : 24), Data merupakan bentuk yang masih mentah yang belum dapat bercerita banyak, sehingga perlu diolah lebih lanjut. Data diolah melalui model tertentu menjadi informasi yang dapat dimanfaatkan oleh penerima dalam membuat keputusan dan melakukan tindakan, yang berarti

melakukan suatu tindakan lain yang akan membuat sejumlah data kembali. Data yang masih belum diolah akan disimpan dalam bentuk *database*. Data yang disimpan ini nantinya dapat diambil kembali untuk diolah kembali menjadi informasi. Data tersebut akan ditangkap sebagai input, diproses kembali lewat suatu model tertentu dan seterusnya membentuk suatu siklus. Siklus ini oleh John Burch disebut dengan siklus informasi (*information cycle*).

II.3. Sistem Informasi Akuntansi

Menurut Anastasia Diana dan Lilis Setiawati (2011 : 2) Tujuan dasar suatu sistem tergantung pada jenis sistem itu sendiri. Sebagai contoh, sistem peredaran darah manusia merupakan sistem biologi yang memiliki tujuan untuk mengedarkan darah yang mengandung oksigen dan sari makanan ke seluruh tubuh. Sedangkan sistem buatan manusia seperti sistem yang terdapat di sekolah, organisasi bisnis, atau instansi pemerintah juga mempunyai tujuan yang berbeda-beda. Organisasi bisnis biasanya memiliki tujuan yang lebih jelas.

II.4. UML (*Unified Modelling Language*)

Menurut Prabowo Pudjo Widodo dan Herlawati (2011 : 6-7), *UML* singkatan dari *Unified Modelling Language* yang berarti bahasa permodelan standar. UML diaplikasikan untuk maksud tertentu, biasanya antara lain untuk :

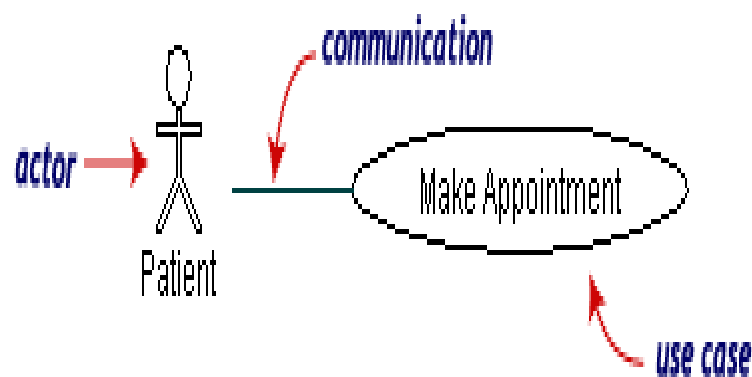
1. Merancang perangkat lunak
2. Sarana Komunikasi antara perangkat lunak dengan proses bisnis.

3. Menjabarkan sistem secara rinci untuk analisa dan mencari apa yang diperlukan sistem.
4. Mendokumentasi sistem yang ada, proses-proses dan organisasinya.

Menurut Prastuti Sulistyorini (2009) 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 :

II.4.1. Use Case Diagram

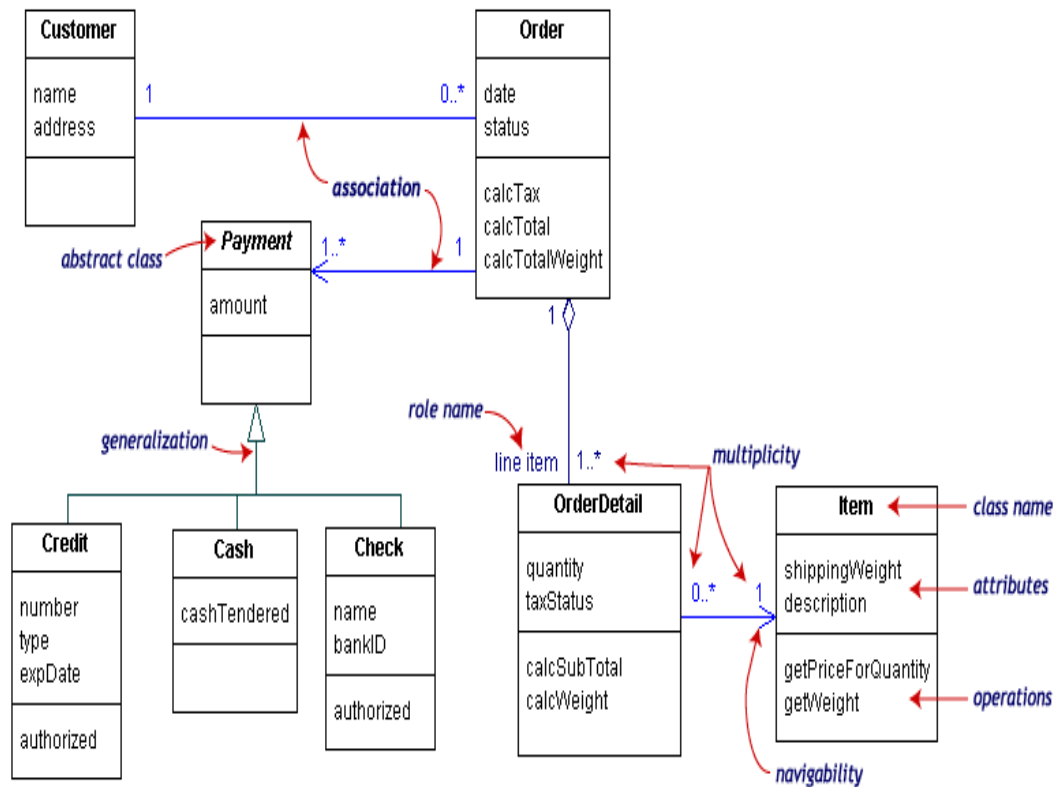
Menurut Prastuti Sulistyorini (2009) Diagram ini bersifat statis. Diagram ini memperlihatkan himpunan use case dan aktor-aktor (suatu jenis khusus dari kelas). Diagram ini terutama sangat penting untuk mengorganisasi dan memodelkan perilaku dari suatu sistem yang dibutuhkan serta diharapkan pengguna.



Gambar II.1. Contoh Use Case Diagram
(sumber : Prastuti Sulistyorini : 2009)

II.4.2. Class Diagram

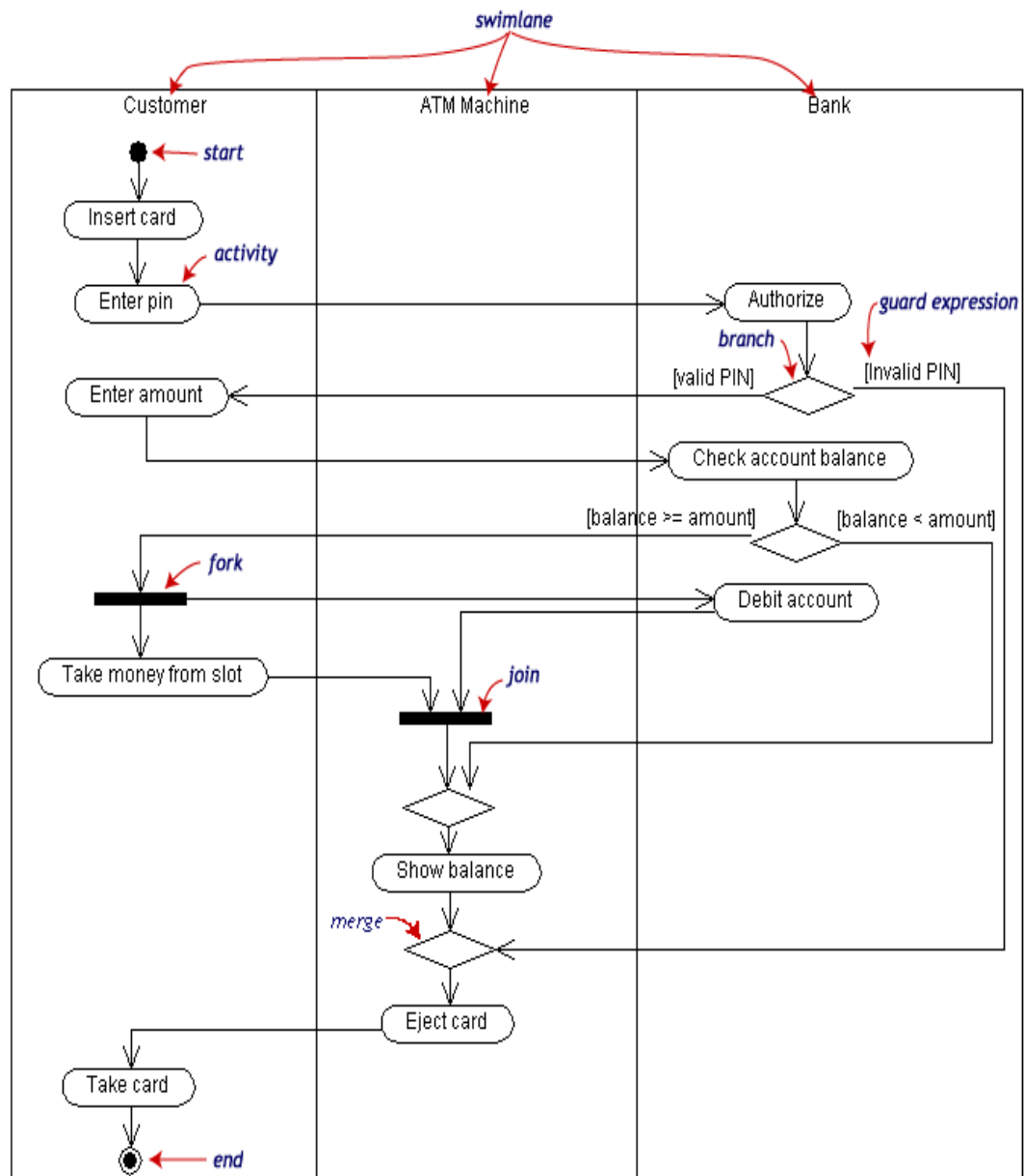
Menurut Prastuti Sulistyorini (2009) Diagram kelas bersifat statis. Diagram ini memperlihatkan himpunan kelas-kelas, antarmuka antarmuka, kolaborasi-kolaborasi serta relasi.



Gambar II.2. Contoh Class Diagram
(sumber : Prastuti Sulistyorini : 2009)

II.4.3. Activity Diagram

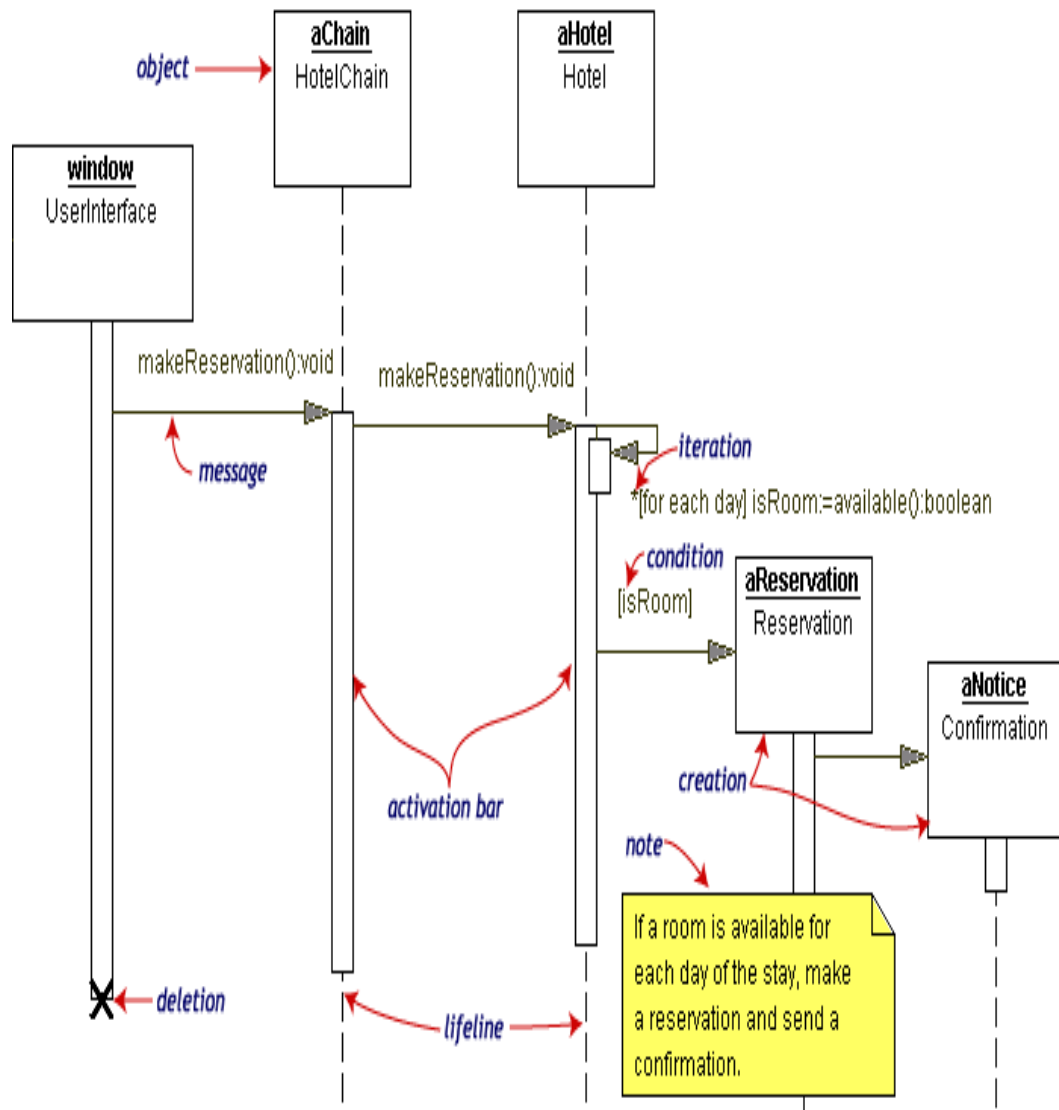
Menurut Prastuti Sulistyorini (2009) Diagram ini bersifat dinamis. Diagram ini adalah tipe khusus dari diagram state yang memperlihatkan aliran dari suatu aktifitas ke aktifitas lainnya dari suatu sistem. Diagram ini terutama penting dalam pemodelan fungsi – fungsi dalam suatu sistem dan memberi tekanan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



Gambar II.3. Contoh Activity Diagram
(sumber : Prastuti Sulistyorini : 2009)

II.4.4. Sequence Diagram

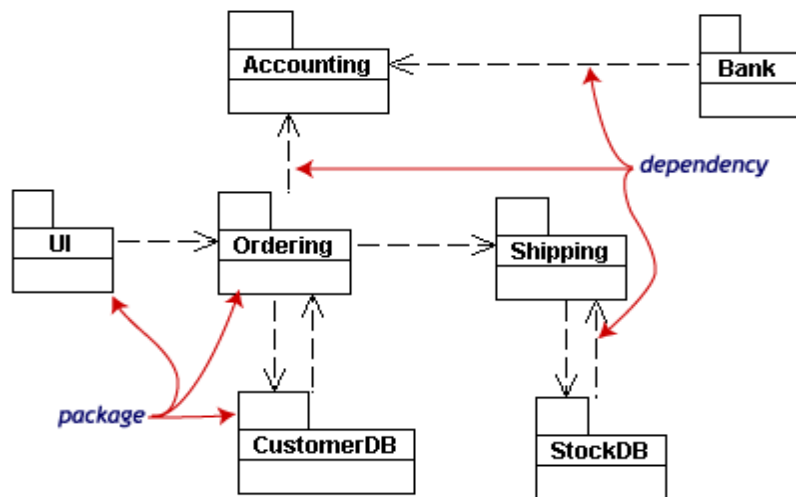
Menurut Prastuti Sulistyorini (2009) Diagram ini bersifat dinamis. Diagram sequence merupakan diagram interaksi yang menekankan pada pengiriman pesan (message) dalam suatu waktu tertentu.



Gambar II.4. Contoh Sequence Diagram
(sumber : Prastuti Sulistyorini : 2009)

II.4.5. Diagram Package Object

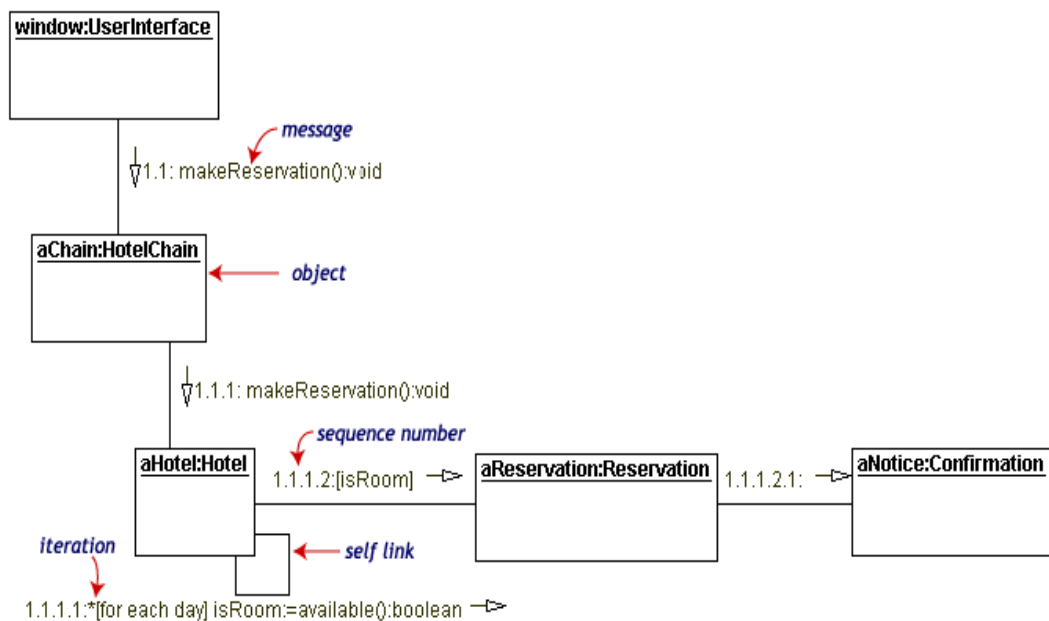
Menurut Prastuti Sulistyorini (2009) Diagram objek bersifat statis. Diagram ini memperlihatkan objek-objek serta relasi antar objek. Diagram objek memperlihatkan instansiasi statis dari segala sesuatu yang dijumpai pada diagram kelas.



Gambar II.5. Contoh Package Object Diagram
 (sumber : Prastuti Sulistyorini : 2009)

II.4.6. Collaboration Diagram

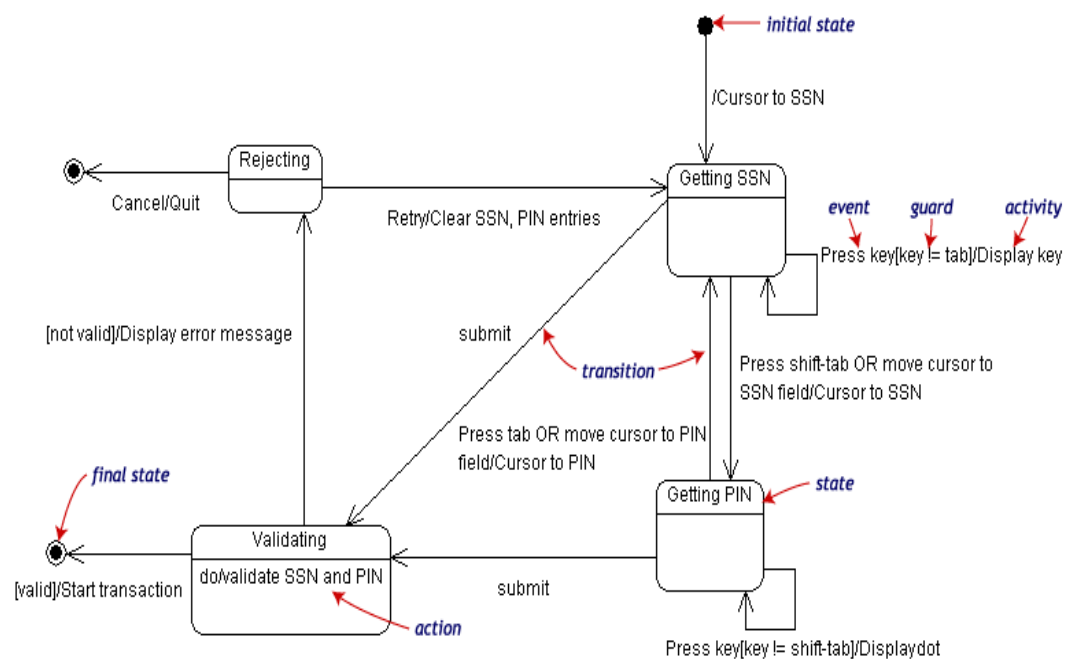
Menurut Prastuti Sulistyorini (2009) Diagram ini bersifat dinamis. Diagram kolaborasi adalah diagram interaksi yang menekankan organisasi struktural dari objek – objek yang menerima serta mengirim pesan (message).



Gambar II.6. Contoh Collaboration Diagram
 (sumber : Prastuti Sulistyorini : 2009)

II.4.7. Statechart Diagram

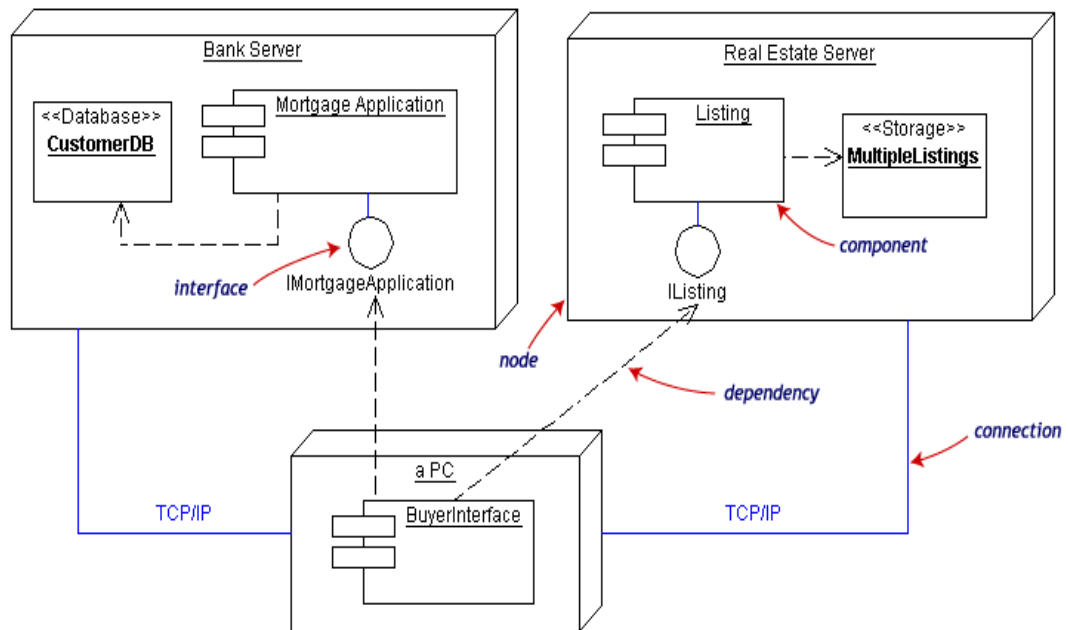
Menurut Prastuti Sulistyorini (2009) Diagram ini bersifat dinamis. Diagram ini memperlihatkan state – state pada sistem, memuat state, transisi, event, serta aktifitas. Diagram ini terutama penting untuk memperlihatkan sifat dinamis dari antarmuka, kelas, kolaborasi dan terutama penting pada pemodelan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



Gambar II.7. Contoh Statechart Diagram
(sumber : Prastuti Sulistyorini : 2009)

II.4.8. Component dan Deployment Diagram

Menurut Prastuti Sulistyorini (2009) Diagram ini bersifat statis. Diagram ini memperlihatkan organisasi serta kebergantungan 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.



Gambar II.8. Contoh Component dan Deployment Diagram
(sumber : Prastuti Sulistyorini : 2009)

II.5. Aktiva Lancar

Menurut Binsar H. Simanjuntak, dkk (2008) Pada dasarnya aktiva dapat diklasifikasikan menjadi dua bagian utama yaitu aktiva lancar dan aktiva tidak lancar. Sebelum lebih jauh kita bicara tentang Aktiva Lancar dan Aktiva tidak Lancar alangkah baiknya kita lihat dulu pengertian aktiva. Aktiva adalah sumber daya yang dikuasai oleh perusahaan sebagai akibat dari peristiwa masa lalu dan dari mana manfaat ekonomis di masa depan diharapkan akan diperoleh perusahaan.

Menurut Binsar H. Simanjuntak, dkk (2008 : 4) Pengertian Aktiva lancar adalah uang kas dan aktiva-aktiva lain atau sumber-sumber yang diharapkan akan direalisasi menjadi uang kas atau dijual atau dikonsumsi selama siklus usaha perusahaan yang normal atau dalam waktu satu tahun, mana yang lebih lama.

II.6. Metode Garis Lurus

Menurut Binsar H. Simanjuntak, dkk (2008 : 4) Dengan metode ini penyusutan tahunan dapat ditentukan dengan dua cara yaitu :

1. (Cost-nilai residu) : umur

Misalkan sebuah peralatan yang diperoleh tahun 2005 senilai Rp. 16.000.000,00 dan masa manfaat ditentukan 5 tahun dengan nilai sisa Rp. 1.000.000,00 besar penyusutan tahun 2006 dapat dihitung sebagai berikut $(16.000.000,00 - 1.000.000,00) / 5 = \text{Rp.}3.000.000,00$

2. Ditentukan % penyusutan, kemudian penyusutan tahunan diperoleh dengan cara mengalikan % tersebut dengan cost yang disusutkan sebagai berikut :

- a. Persentase penyusutan tahunan = $100\% : \text{umur}$, jadi = $100\% : 5 = 20\%$
- b. Dihitung penyusutan = $20\% \times (16.000.000,00 - 1.000.000,00) = \text{Rp.}3.000.000,00$

II.7. Desain Database

Menurut Yuniar Supardi (2008 : 9) *Desain database* merupakan pekerjaan yang penting dalam pembuatan atau pengembangan sistem, karena desain *database* akan mendapatkan susunan data atau *table* yang efektif dan efisien. Alat desain *database* yang populer ada dua, yaitu : ERD (*Entity Relationship Diagram*) dan Normalisasi. Jika memakai *Normalisasi* harus mendapatkan Data Dasar (Dokumen Dasar), sedangkan ERD tidak perlu. Dalam desain ERD terbagi dua

tahapan yaitu: *Preliminary Design* (Desain Awal) dan *Final Design* (Desain Akhir). Tetapi disain Akhir dari ERD juga berisi Normalisasi.

II.8. Kamus Data Dan Normalisasi

Menurut Budi Raharjo (2011 : 59) dalam suatu rancangan *database*, *data dictionary* digunakan untuk menjelaskan atau mendeskripsikan kolom-kolom pada masing-masing tabel yang akan dibuat dalam *database*. Deskripsi kolom yang dimaksud di sini meliputi tipe data, lebar karakter atau digit, serta keterangan tentang kunci relasi.

Tabel II.1. Tabel kategori

Nama Kolom	Tipe Data	Lebar	NULL?	Kunci
Kategori_id	INT	11	NOT NULL	Primary Key
Kategori_nama	VARCHAR	25		

Tabel II.2. Tabel pengarang

Nama Kolom	Tipe Data	Lebar	NULL?	Kunci
Pengarang_id	CHAR	3	NOT NULL	Primary Key
Pengarang_nama	VARCHAR	30		

Tabel II.3. Tabel penerbit

Nama Kolom	Tipe Data	Lebar	NULL?	Kunci
Penerbit_id	CHAR	4	NOT NULL	Primary Key
Penerbit_nama	VARCHAR	50		

Tabel II.4. Tabel buku

Nama Kolom	Tipe Data	Lebar	NULL?	Kunci
Buku_isbn	CHAR	13	NOT NULL	Primary Key
Buku_judul	VARCHAR	75		
Penerbit_id	CHAR	4		
Buku_tglterbit	DATE	-		
Buku_jmlhalaman	INT	11		
Buku_deskripsi	TEXT	-		
Buku_harga	DECIMAL	10,0		

Tabel II.5. Tabel link_buku_pengarang

Nama Kolom	Tipe Data	Lebar	NULL?	Kunci
Buku_isbn	CHAR	13	NOT NULL	Primary Key dan Forign Key
Pengarang_id	CHAR	3	NOT NULL	Primary Key dan Forign Key

Tabel II.6. Tabel link_buku_kategori

Nama Kolom	Tipe Data	Lebar	NULL?	Kunci
Buku_isbn	CHAR	13	NOT NULL	Primary Key dan Forign Key
kategori_id	CHAR	11	NOT NULL	Primary Key dan Forign Key

(Sumber : Budi Raharjo : 2011 : 59)

II.9. Entity Relationship Diagram (ERD)

Menurut Budi Raharjo (2011 : 57) *entity relationship diagram* merupakan salah satu alat bantu (berupa gambar) dalam *database relasional* yang berguna untuk menjelaskan hubungan atau relasi antartabel yang terdapat di dalam *database*. Dalam ERD kita juga dapat melihat daftar kolom yang menyusun masing-masing tabel. ERD inilah yang akan kita gunakan sebagai acuan untuk membahas materi dalam buku ini. Berikut keterangan tanda yang digunakan :

- 1-∞ menunjukkan relasi 1-ke-banyak
- n-∞ menunjukkan relasi banyak-ke-banyak