

## **BAB II**

### **TINJAUAN PUSTAKA**

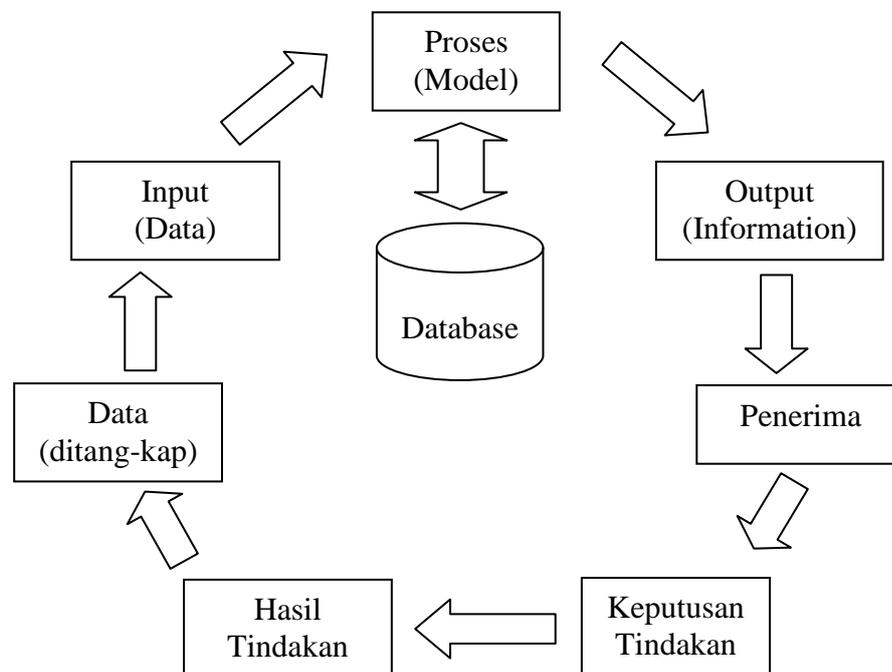
#### **II.1. Sistem**

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. Data Dan Informasi**

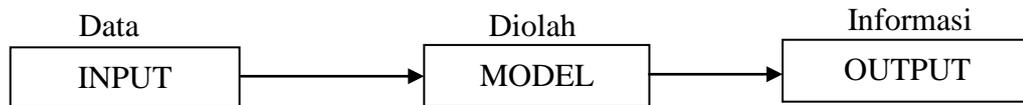
Menurut Riyanto, dkk. (2009 : 24), Data merupakan bentuk yang masih mentah yang belum dapat bercerita banyak, sehingga perlu diolah lebih lanjut. Data diolah melalui model tertentu menjadi informasi yang dapat dimanfaatkan oleh penerima dalam membuat keputusan dan melakukan tindakan, yang berarti

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*). Agar menjadi informasi yang berguna, data perlu diolah melalui sebuah siklus. Siklus ini disebut siklus pengolahan data (*data processing life cycle*).



**Gambar II.1. Siklus Informasi**  
(Sumber : Riyanto, dkk. : 2009 : 24)

Menurut Riyanto, dkk. (2009 : 23), Agar menjadi informasi yang berguna, data perlu diolah melalui sebuah siklus. Siklus ini disebut siklus pengolahan data (*data processing life cycle*).



**Gambar II.2. Siklus Pengolahan Data**  
(Sumber : Riyanto, dkk. : 2009 : 23)

### II.3. Sistem Informasi

Menurut Tata Sutabri (2012:58) Tujuan sistem informasi adalah memenuhi kebutuhan informasi semua manajer dalam perusahaan atau dalam sub-unit organisasi perusahaan. Sistem informasi menyediakan informasi bagi pemakai dalam bentuk laporan dan output dari berbagai simulasi model matematika, dimana proses manajemen didefinisikan sebagai aktivitas-aktivitas :

1. Perencanaan, formulasi terinci untuk mencapai suatu tujuan akhir tertentu adalah aktivitas manajemen yang disebut perencanaan.
2. Pengendalian, perencanaan hanyalah setengah dari pertempuran.

### II.4. 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. 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;2006:3).

Analog dengan definisi sebelumnya, SIA (Sistem Informasi Akuntansi) adalah sistem berbasis computer yang dirancang untuk mentransformasi data akuntansi menjadi informasi (George H. Bodnar; 2006: 8).

Dalam aktivitas pengembangan sistem, diharapkan akuntan dan auditor dapat bertindak secara profesional. Akuntansi dapat menjalankan aktivitas pengembangan sistem baik untuk perusahaan mereka sendiri atau untuk perusahaan lain, dalam hal mereka memiliki posisi sebagai konsultan. *Auditor eksternal* maupun *internal* berhadapan dengan aktivitas pengembangan sistem pada saat mereka mengevaluasi pengendalian sistem informasi sebagai bagian dari penugasan audit suatu perusahaan (George H. Bodnar;2006:22).

Sebuah proyek pengembangan sistem biasanya terdiri dari tiga fase, yaitu analisa sistem, desain sistem, dan implementasi sistem. Analisa sistem melibatkan penyusunan solusi dan evaluasi solusi untuk menyelesaikan masalah sistem. Analisa sistem menekankan tujuan sistem secara keseluruhan. Dasar dari analisa sistem ini adalah timbal balik antar tujuan sistem. Tujuan umum analisa sistem, yaitu:

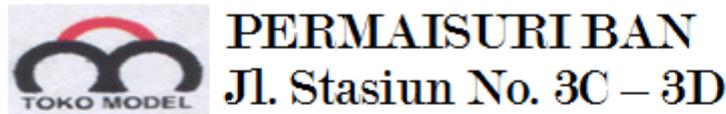
1. Untuk meningkatkan kualitas ekonomi.
2. Untuk meningkatkan pengendalian internal.
3. Untuk meminimalkan biaya, jika memungkinkan.

Desain sistem merupakan proses merancang secara rinci solusi yang telah dipilih dalam proses analisis. Desain sistem mencakup evaluasi efektivitas dan efesiansi alternatif rancangan sistem terkait dengan kebutuhan sistem secara keseluruhan. Implementasi sistem merupakan proses penerapan prosedur dan metode yang telah dirancang ke dalam operasi. Implementasi sistem mencakup pengujian solusi sebelum implementasi, dokumentasi, serta evaluasi sistem pada saat sistem tersebut mulai dioperasikan untuk memastikan bahwa sistem berfungsi sesuai dengan yang telah direncanakan (George H. Bodnar;2006:23).

## **II.5. Penggajian**

Menurut Anastasia Diana Dan Lilis Setiawati (2011:182) Dokumen yang dipakai dalam pembayaran gaji antara lain meliputi :

1. Kartu waktu : Kartu waktu berguna untuk merekam presensi setiap hari, jam berapa karyawan hadir di kantor dan jam berapa pulang dari kantor.
2. Daftar gaji : Daftar gaji memuat gaji seluruh karyawan.
3. Slip gaji : Slip gaji memuat rincian komponen gaji.
4. Daftar transfer : Daftar tansfer berfungsi sebagai surat perintah ke bank untuk mentransfer sejumlah tertentu ke setiap karyawan yang akan menerima gaji.



### Bukti Pembayaran Gaji

Nama Mekanik	: .....
Alamat	: .....
Gaji	: Rp. 50.000,-
Uang Makan	: Rp. 10.000,-
Uang Transportasi	: Rp. 10.000,-
Uang/Kendaraan @ Rp. 5.000	: .....
Uang Kerajinan/10 Kendaraan	: Rp. 25.000,-
Total	: .....

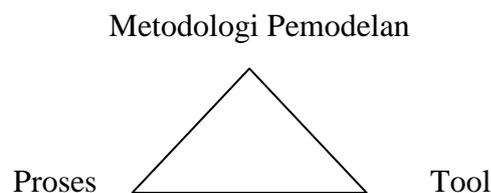
Yang Menerima

(.....)

**Gambar II.3. Slip Gaji**  
(Sumber : Permaisuri Ban)

## II.6. UML (Unnified Modelling Language)

Menurut Prabowo Pudjo Widodo dan Herlawati (2011 : 6-7), *UML* singkatan dari *Unified Modelling Language* yang berarti bahasa permodelan standar. Menurut Yuni Sugiarti (2013 : 34) *Unified Modelling Language (UML)* adalah sebuah “bahasa” yang telah menjadi standar dalam industri untuk *visualisasi*, merancang dan mendokumentasikan sistem piranti lunak. *UML* menawarkan sebuah standar untuk merancang model sebuah sistem.

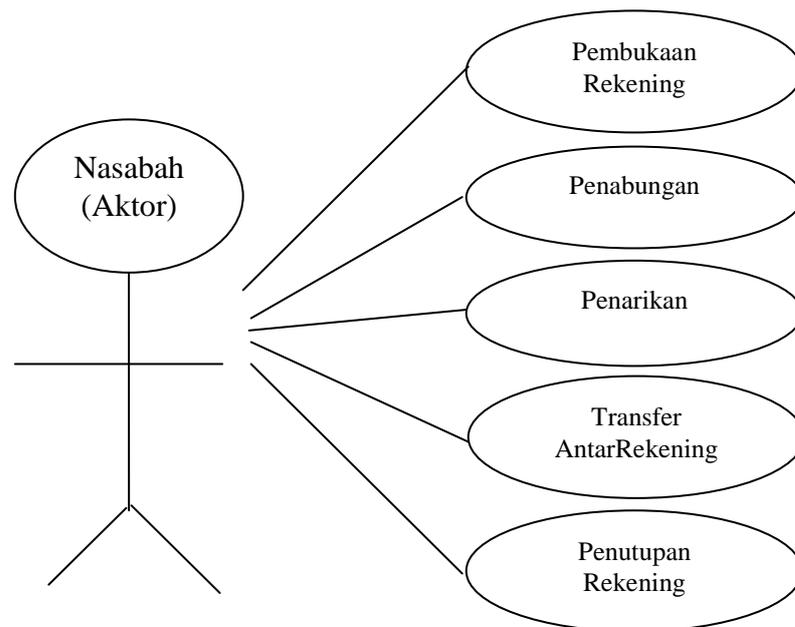


**Gambar II.4. The Triangle For Success**  
(Sumber : Yuni Sugiarti : 2013 ; 34)

### II.6.1. Use Case Diagram

Segala sesuatu yang secara akademis dikembangkan pada umumnya berawal dari suatu konsep. Demikian juga halnya dengan pengembangan sistem pada umumnya dikembangkan berdasarkan analisis kebutuhan. Analisis kebutuhan ini adalah tahap konseptualisasi, yaitu suatu tahap yang mengharuskan analis dan perancang sistem untuk berusaha tahu secara pasti mengenai hal yang menjadi kebutuhan dan harapan pengguna sehingga kelak aplikasi yang dibuat memang akan digunakan oleh pengguna (*user*) serta akan memuaskan kebutuhan dan harapannya.

Dalam konteks UML, tahap konseptualisasi dilakukan dengan pembuatan *use case diagram* yang sesungguhnya merupakan deskripsi peringkat tinggi bagaimana perangkat lunak (aplikasi) akan digunakan oleh penggunanya. Selanjutnya, *use case diagram* tidak hanya sangat penting pada saat analisis, tetapi juga sangat penting dalam tahap perancangan (*design*), untuk mencari kelas-kelas yang terlibat dalam aplikasi, dan untuk melakukan pengujian (*testing*). Saat akan mengembangkan *use case diagram*, hal yang pertama kali harus dilakukan adalah mengenali *actor* untuk sistem yang sedang dikembangkan. Dalam hal ini, ada beberapa karakteristik untuk para *actor*, yaitu *actor* yang ada di luar sistem yang sedang dikembangkan dan *actor* yang berinteraksi dengan sistem yang sedang dikembangkan. (Adi Nugroho ; 2009 : 7)

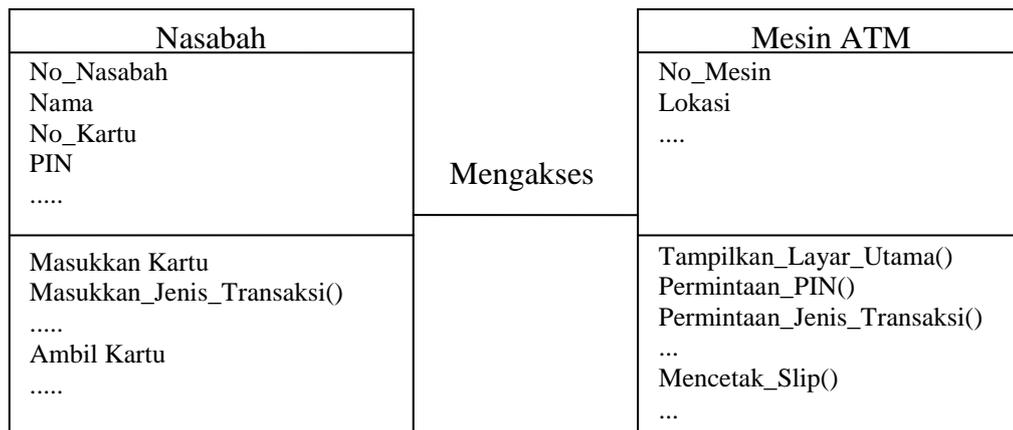


**Gambar II.5. Contoh Use Case Diagram**  
(Sumber : Adi Nugroho ; 2009 : 8)

### II.6.2. Class Diagram

Menurut Rosa A.S dan M. Shalahuddin (2011 : 122), *Class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

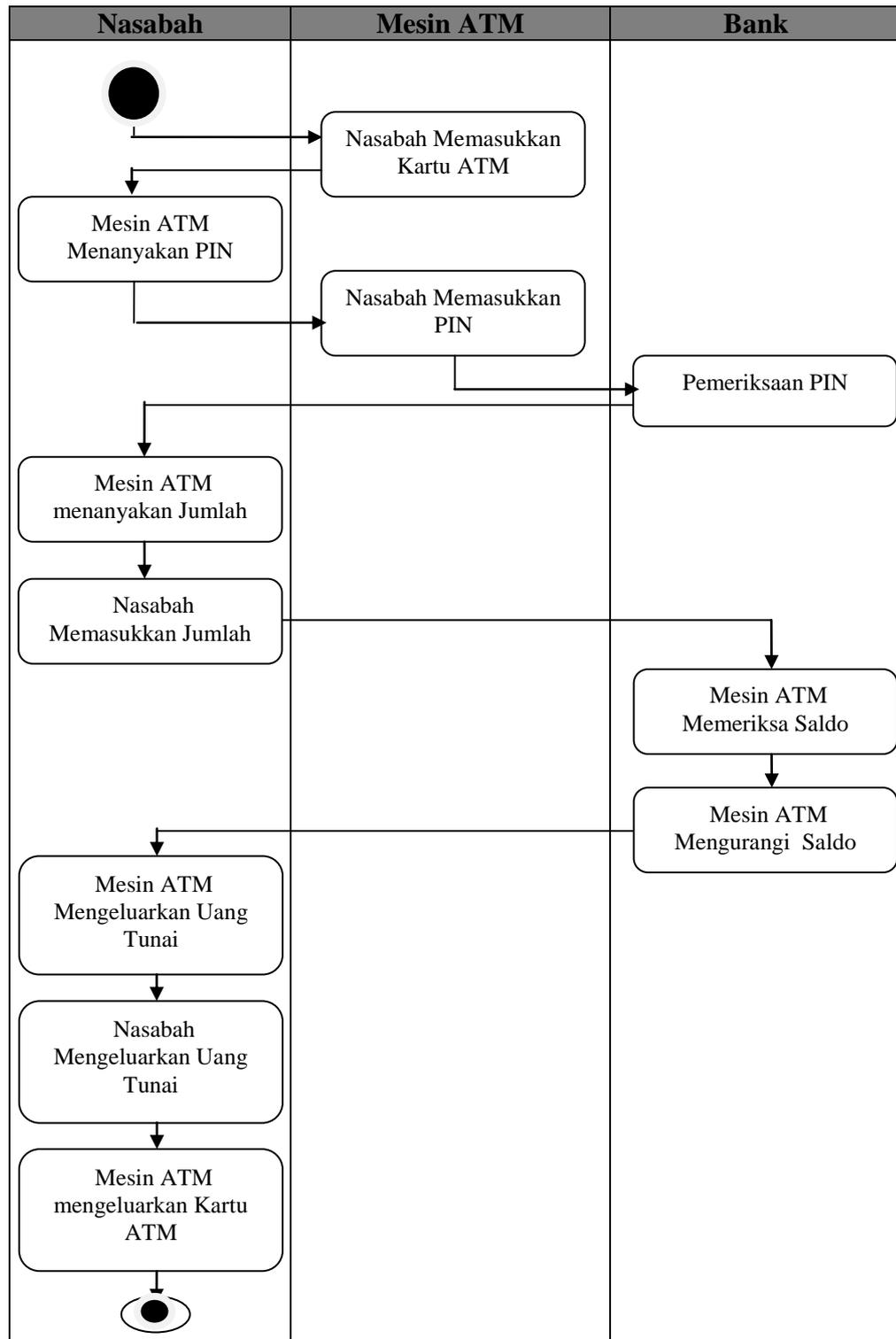
1. Atribut merupakan *varabel-variabel* yang dimiliki oleh suatu kelas.
2. Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas.



**Gambar II.6. Contoh Class Diagram**  
(Sumber : Adi Nugroho ; 2009: 39)

### II.6.3. Activity Diagram

Apakah langkah yang harus kita lakukan selanjutnya setelah kita membuat use case diagram ? use case diagram merupakan gambaran menyeluruh dan pada umumnya sangatlah tidak terperinci. Oleh karena itu, kita harus memperinci lagi perilaku sistem untuk masing-masing use case yang ada. Apa perkakas (tool) yang bisa kita gunakan ? jika kasus kita cukup sederhana, mungkin kita bisa menggunakan skenario seperti yang tercantum berikut, sementara jika kasusnya cukup kompleks, kita mungkin bisa menggunakan activity diagram agar bisa mendapatkan gambaran yang lebih menyeluruh (Adi Nugroho ; 2009 : 10).



**Gambar II.7. Contoh Activity Diagram**  
 (Sumber : Adi Nugroho ; 2009 : 11)

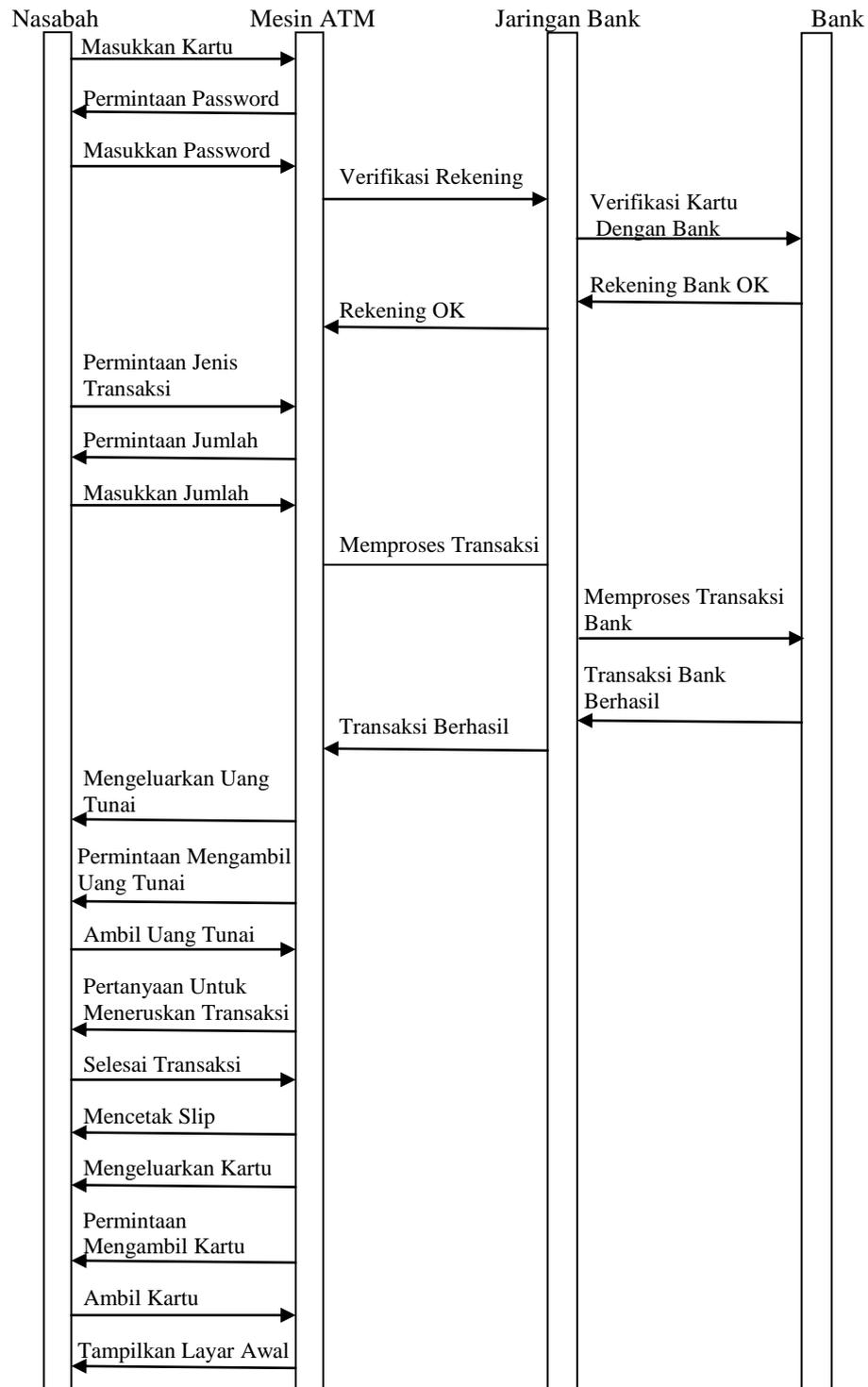
#### II.6.4. Sequence Diagram

Diagram sekuensial atau *sequence diagram* digunakan untuk menunjukkan aliran fungsionalitas dalam *use case*. Diagram sekuensial adalah diagram yang disusun berdasarkan urutan waktu. Kita membaca diagram sekuensial dari atas ke bawah. Setiap diagram sekuensial mempresentasikan suatu aliran dari beberapa aliran di dalam *use case*.

Jadi dengan kata lain sekuensial diagram menunjukkan aliran fungsionalitas berdasarkan urutan waktu serta kejadian yang nantinya akan menentukan metode/fungsi atribut masing-masing. Dimana fungsi-fungsi tersebut akan diterapkan pada suatu kelas/objek.

Pada gambar II.6. terlihat pengelompokkan *event-event* serta fungsi masing-masing atribut tersebut. Di dalam diagram terlihat jelas bagaimana aliran suatu proses kejadian dimana seorang nasabah yang akan melakukan transaksi dengan sebuah mesin ATM. Dari diagram tersebut kita mengetahui *event-event* yang terjadi, seperti : Nasabah memasukkan kartu ATM, Mesin ATM merespon dengan meminta *password* atau PIN, dan selanjutnya.

Kita dapat melihat setiap fungsi atribut dan *event-event* apa saja yang terjadi. Sehingga melalui diagram sekuensial ini kita dapat merancang suatu program aplikasi yang baik, sehingga dalam menghadapi sebuah kasus yang benar-benar kompleks diagram sekuensial ini sangat membantu.



**Gambar II.8. Contoh Sequence Diagram**  
 (Sumber : Adi Nugroho ; 2009 : 36)

## II.7. Database

Menurut Budi Raharjo (2011 : 245) dalam bekerja dengan *database*, anda perlu memiliki pengetahuan yang cukup tentang konsep *client/server* dan tipe-tipe arsitektur aplikasi *database* yang ada. Dengan demikian, anda akan mengetahui dengan pasti tujuan dan fungsi (untuk apa) anda membuat dan mengolah *database*.

## II.8. Basis Data

Menurut Janner Simarmata (2007 : 1), Kata "*basis data*" bisa digunakan untuk menguraikan segala sesuatu dari sekumpulan data tunggal, seperti daftar telepon. *Basis data* terdiri dari *file-file* fisik yang ditetapkan berdasarkan komputer saat menerapkan perangkat lunak basis data. Sedangkan menurut *Stephens* dan *Plew* (Dalam Janner Simarmata dan Iman Paryudi; 2006 : 1), Basis Data adalah mekanisme yang digunakan untuk menyimpan informasi atau data. Informasi adalah sesuatu yang kita gunakan sehari-hari untuk berbagai alasan. Dengan basis data, pengguna dapat menyimpan data secara terorganisasi. Setelah data disimpan, informasi harus mudah diambil. Kriteria dapat digunakan untuk mengambil informasi. Cara data disimpan dalam basis data menentukan seberapa mudah mencari informasi berdasarkan banyak kriteria. Data pun harus mudah ditambahkan kedalam basis data, dimodifikasi, dan dihapus.

## II.9. 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

### II.10. Entity Relationship Diagram (ERD)

Menurut Budi Raharjo (2011 : 57) *entity relationship diagram* merupakan salah satu alat bantu (berupa gambar) dalam *database relational* 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. 1-∞ menunjukkan relasi 1-ke-banyak
2. n-∞ menunjukkan relasi banyak-ke-banyak

### II.11. Microsoft Visual Basic

*Visual basic* merupakan salah satu bahasa pemrograman yang andal dan banyak digunakan oleh pengembang untuk membangun berbagai macam aplikasi *windows*. *Visual basic 2008* merupakan aplikasi pemrograman 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.12. Microsoft SQL Server**

Bahasa query merupakan bahasa khusus 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)