

BAB II

TINJAUAN PUSTAKA

II.1. Pengertian 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 : 21-22).

II.2. Data Dan Informasi

Istilah data dan informasi sering digunakan secara bergantian. Ada yang menyebut data, padahal informasi, sebaliknya ada yang mengatakan informasi, padahal data. Gordon B. Davis menjelaskan kaitannya dengan informasi dalam bentuk defenisi “informasi adalah data yang telah diproses kedalam suatu bentuk

yang mempunyai arti bagi si penerima dan mempunyai nilai nyata dan terasa bagi keputusan saat itu atau keputusan mendatang”. Sumber dari informasi adalah data. Data merupakan bentuk jamak dari bentuk tunggal datum. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Kejadian-kejadian adalah sesuatu yang terjadi pada saat tertentu di dalam dunia bisnis. Bisnis adalah perubahan dari suatu nilai yang disebut transaksi. Misalnya, penjualan adalah transaksi perubahan nilai barang menjadi nilai uang atau nilai piutang. Kesatuan nyata adalah berupa suatu objek yang nyata seperti tempat, benda dan orang yang betul-betul ada dan terjadi (Tata Sutabri : 2013 : 1).

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 (Tata Sutabri ; 2013 : 29).

II.3. Pengertian Pakar

Pakar adalah seorang individu yang memiliki pengetahuan khusus, pemahaman pengalaman dan metode-metode yang digunakan untuk

memecahkan persoalan dalam bidang tertentu. Seorang pakar memiliki kemampuan seorang kepakaran, yaitu :

- a. Dapat mengenali dan merumuskan suatu masalah.
- b. Menjelaskan solusi dari suatu masalah.
- c. Restrukturisasi pengetahuan.
- d. Belajar dari pengalaman.
- e. Memahami batas kemampuan (Rika Rosnelly ; 2012: 10).

II.3.1. Internet Speedy

Menurut Bapak Adewin Lubis selaku Supervisor dan Koordinator CBOC Provisioning Center PT. Telkom Indonesia (pakar) Speedy adalah merupakan layanan broadband akses internet dari Telkom Indonesia berkualitas tinggi bagi perumahan serta *SME (Small Medium Enterprise)*. Speedy menggunakan teknologi *ADSL (Asymmetric Digital Line Subscriber)*, *MSAN (Multi Service Access Node)*, dan *GPON (Gigabit Passive Optical Network)*, yang menghantarkan sinyal digital berkecepatan tinggi melalui jaringan telefoni secara optimal bagi keperluan konsumsi konten internet, dengan kecepatan data dari 384 kbps hingga 100 Mbps. Tetapi tidak menutup kemungkinan penggunaan teknologi baru lain yang lebih baik/terbaru. Saat ini Telkom menyediakan 3 (tiga) jenis layanan Speedy:

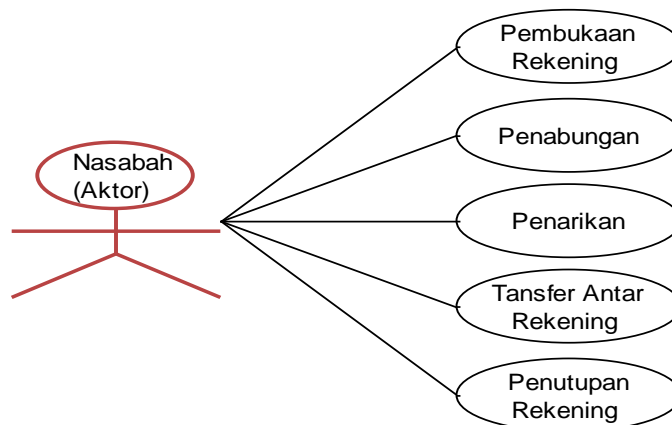
1. Speedy reguler
2. Speedy instan
3. Speedy gold

II.4. Unified Modelling Language

II.4.1. Use Case Diagram

Segala sesuatu yang secara akademis dikembangkan pada umumnya berawal dari suatu konsep. 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.

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)



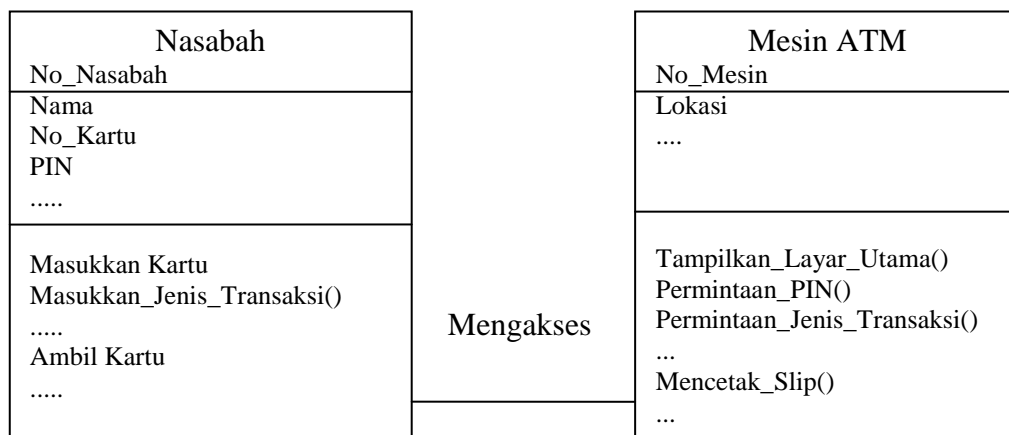
Gambar II.1. Contoh Use Case Diagram
(Sumber : Adi Nugroho ; 2009 : 8)

II.4.2. Class Diagram

Class didefinisikan sebagai kumpulan/himpunan objek yang memiliki kesamaan dalam atribut/properti, perilaku (operasi), serta cara berhubungan dengan objek lain (Adi Nugroho ; 2009 : 18).

Selain itu, kita juga mendefinisikan objek sebagai konsep, abstraksi dari sesuatu dengan batas nyata, sehingga kita dapat menggambarkan secara sistematis. Pemahaman objek memiliki dua fungsi, yaitu :

- a. Memudahkan untuk mempelajari secara seksama hal-hal yang ada di dunia nyata.
- b. Menyediakan suatu dasar yang kuat dalam implementasi ke dalam sistem terkomputerisasi (Adi Nugroho ; 2009 :17).

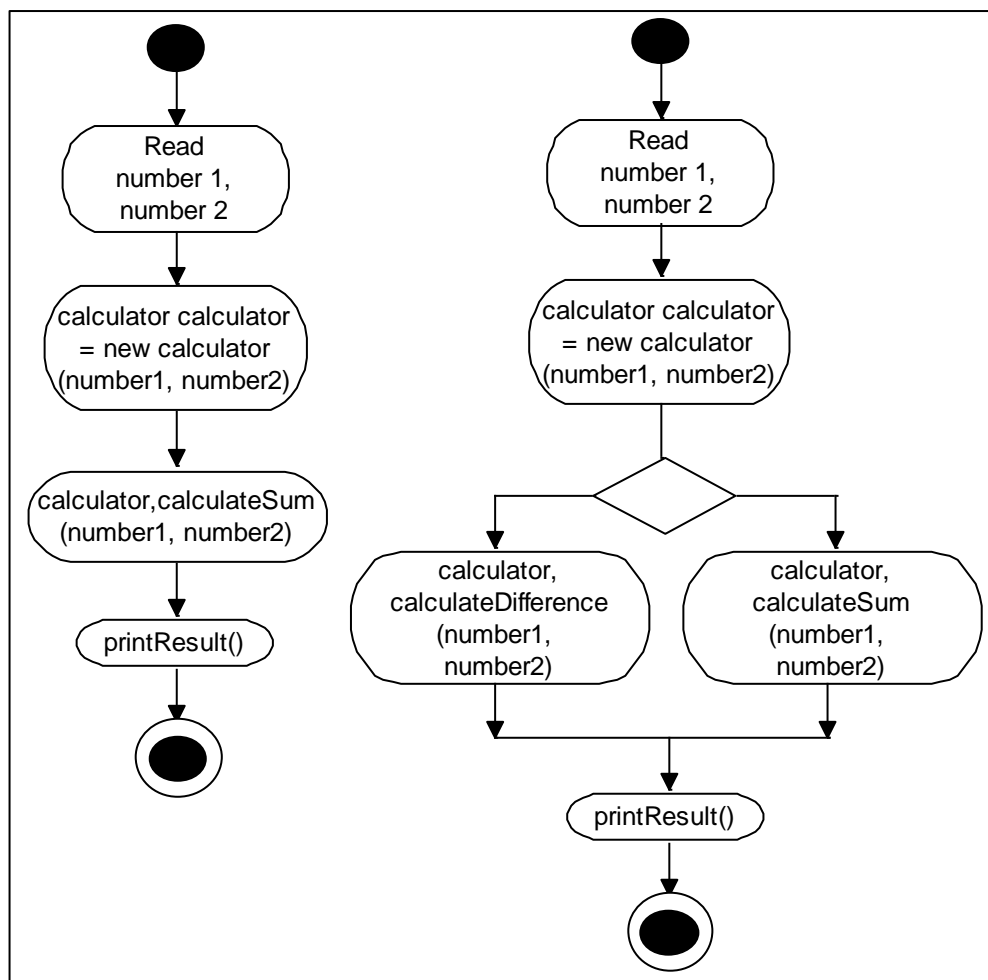


Gambar II.2. Contoh Class Diagram
(Sumber : Adi Nugroho ; 2009: 39)

II.4.3. 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.3. Contoh Activity Diagram
(Sumber : Adi Nugroho ; 2009 : 74)

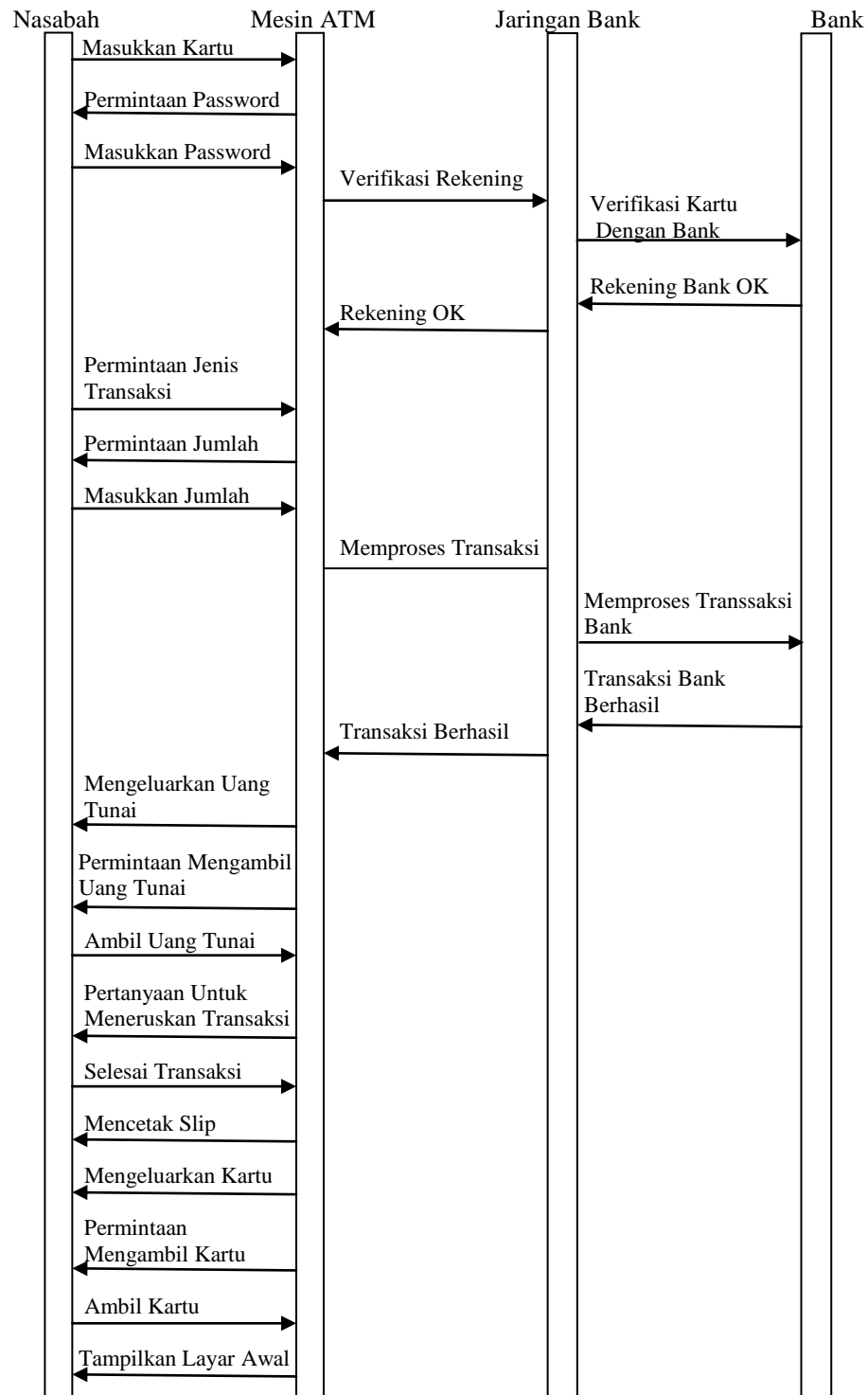
II.4.4. Sequence Diagram

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.

Perhatikan gambar II.4. dimana 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.4. Contoh Sequence Diagram
(Adi Nugroho ; 2009 : 36)**

II.5. 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).

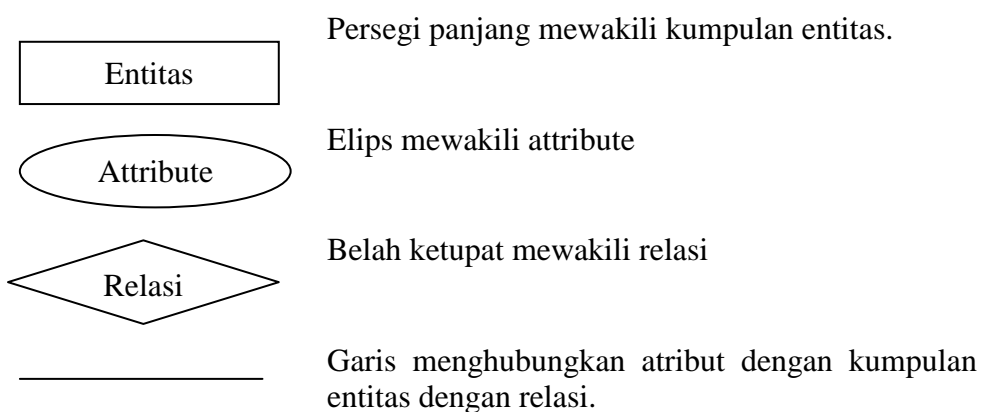
II.6. Microsoft SQL Server

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).

II.7. Database Dan ERD

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* Desain (Disain Awal) dan *Final Design* (Disain Akhir). Tetapi disain Akhir dari ERD juga berisi Normalisasi.

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



II.8. Data Dictionary (kamus data)

Dalam suatu rancangan *database*, *data dictionary* digunakan untuk menjelaskan atau mendeskripsikan kolom-kolom pada masing-masing tabel yang

akan dibuat ke dalam *database*. Deskripsi kolom yang dimaksud di sini meliputi tipe data, lebar karakter atau digit, serta keterangan tentang kunci relasi (Budi Raharjo : 2011 ; 59).

II.9. Normalisasi

Menurut Yuniar Supardi (2008 : 10) tahapan normalisasi terdiri dari beberapa bentuk, yaitu:

1. Bentuk Tak Normal (UNF / *Un Normal Form*).
2. Bentuk Normal Pertama (1 NF / *First Normal Form*).
3. Bentuk Normal Kedua (2 NF / *Second Normal Form*).
4. Bentuk Normal Ketiga (3 NF / *Third Normal Form*).

Tabel II.1 3NF

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

(Yuniar Supardi ; 2008 : 11)

5. Bentuk Normal *Boyce Codd* (BCNF / *Boyce Codd Normal Form*).
6. Normal yang lebih tinggi.

II.10. Metode Forward Chaining

Menurut Rika Rosnelly (2012: 10) Pakar adalah seorang individu yang memiliki pengetahuan khusus, pemahaman pengalaman dan metode-metode yang digunakan untuk memecahkan persoalan dalam bidang tertentu.

Menurut Kusrini (2006:35) *Inferensi* merupakan proses untuk menghasilkan informasi dari fakta yang diketahui atau diasumsikan. *Inferensi* adalah konklusi logis (*logical conclusion*) atau *implikasi* berdasarkan informasi yang berbeda. Dalam sistem pakar, proses *inferensi* dilakukan dalam suatu model yang disebut *inference engine* (mesin inferensi). Ketika *representasi* pengetahuan (RP) pada bagian *knowledge base* telah lengkap, atau paling tidak telah berada pada *level* yang cukup akurat, maka *representasi* pengetahuan tersebut telah siap digunakan. *Inference engine* merupakan modul yang berisi program tentang bagaimana mengendalikan proses *reasoning*.

Runut maju (*forward chaining*) berarti menggunakan himpunan aturan kondisi-aksi. Dalam metode ini, data digunakan menentukan aturan mana yang akan dijalankan, kemudian aturan tersebut dijalankan. Mungkin proses penambahan data ke memori kerja. Proses diulang sampai ditemukan suatu hasil.

Metode *ireferensi* runut maju cocok digunakan untuk menangani masalah pengendalian (*controlling*) dan peramalan (*prognosis*). Berikut ini adalah contoh *inferensi* dengan menggunakan metode runut maju :

Jika : Penderita terkena penyakit epilepsi idiopatik dengan CF antara
0,4 sampai dengan 0,6

Maka : Berikan obat carbamazepine (Kusrini ; 2006 : 36).