

BAB II

TINJAUAN PUSTAKA

II.1. Sistem

Menurut Anastasia Diana dan Lilis Setiawati (2011;2) Sistem merupakan serangkaian bagian saling berhubungan dan bekerja sama untuk mencapai tujuan tertentu. Suatu sistem pasti tersusun dari sub-sub sistem yang lebih kecil yang juga saling tergantung dan bekerja sama untuk mencapai tujuan.

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 ” (Riyanto, dkk; 2009 : 21-22).

II.2. 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 definisi “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. 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).

Jadi dapat ditarik sebuah kesimpulan bahwa informasi adalah sesuatu data yang sudah memiliki arti dapat digunakan untuk kepentingan hal-hal tertentu guna untuk kemajuan suatu sistem.

II.3. Pakar

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. 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.

II.4. Forward Chaining

Menurut Rika Rosnelly (2012: 57) Forward chaining merupakan pencocokan fakta atau pernyataan dimulai dari bagian sebelah kiri (IF dulu).

Dengan kata lain, penalaran dimulai dari fakta terlebih dahulu untuk menguji hipotesis.

Tabel II.1. Contoh Aturan-aturan

No	Aturan
R-1	IF A & B THEN C
R-2	IF C THEN D
R-3	IF A & E THEN F
R-4	IF A THEN G
R-5	IF F & G THEN D
R-6	IF G & E THEN H
R-7	IF C & H THEN I
R-8	IF I & A THEN J
R-9	IF G THEN J
R-10	IF J THEN K

II.5. Perkembangan Janin

Proses perkembangan janin : manusia adalah salah satu hasil karya Tuhan paling menakjubkan, manusia adalah makhluk di bumi paling sempurna, karena memiliki kecerdasan untuk bersosial dan membangun peradaban, selama ribuan tahun manusia juga mempelajari ilmu kesehatan dimana orang-orang berharap bisa mengetahui proses perkembangan janin. Awal perkembangan janin , ketika embrio baru saja terbentuk, besarnya kira-kira 9mm. Minggu ke 8, janin menyerap makanan dan pertumbuhan berkembang sangat pesat. Minggu ke 10, mata mulai terbentuk dan saat telah terbentuk, mata tersebut akan tertutup oleh semacam selaput. Minggu ke 16, lengan, jari-jari tangan dan kaki mulai terbentuk dengan sempurna. Minggu ke 18, proses pembentukan semua organ tubuh dengan lengkap. Umur 6,5 bulan, rambut-rambut halus tumbuh pada tubuh janin , dan untuk seterusnya perkembangannya akan terus berlanjut sampai ia siap dilahirkan.

II.6. 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).

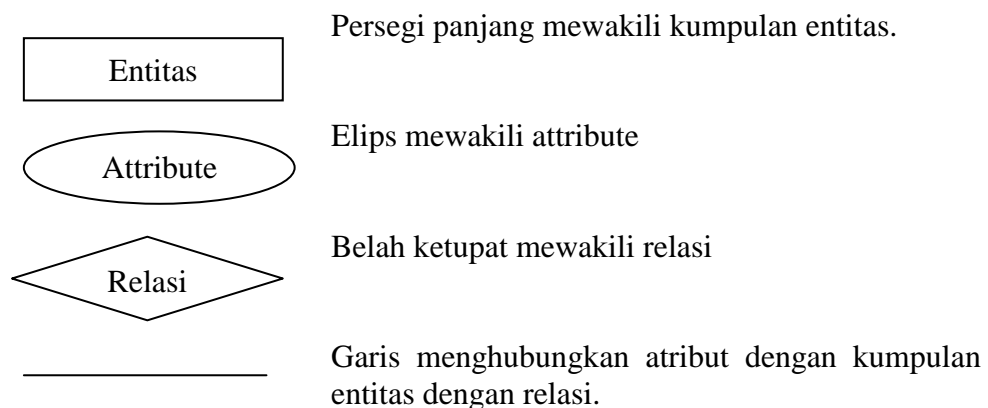
II.7. Microsoft SQL Server

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.8. 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.9. 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.10. 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*).

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 Depedency*) 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* :

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*).

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

II.11. Unified Modelling Language

Pemodelan (*modeling*) sesungguhnya digunakan untuk penyederhanaan permasalahan. Adapun tujuan pemodelan (dalam kerangka pengembangan sistem/perangkat lunak aplikasi) adalah sebagai sarana visualisasi dan komunikasi antar anggota tim pengembang (saat seorang analis/perancang sistem/perangkat

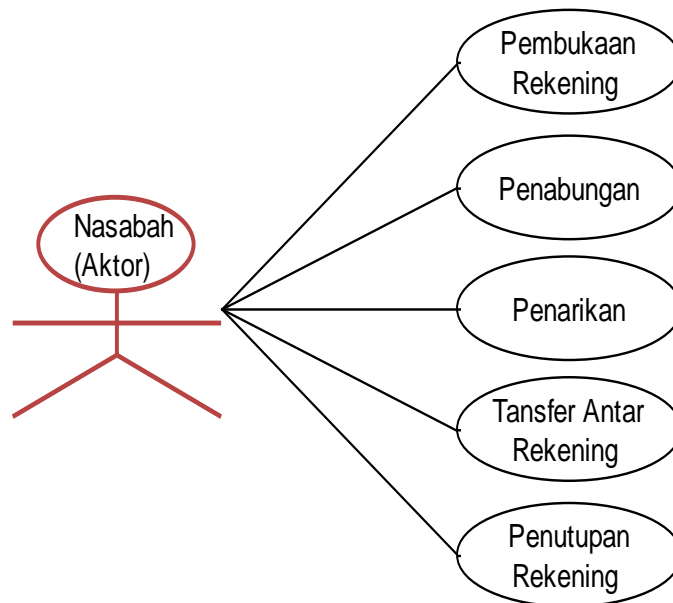
lunak bekerja dalam tim yang beranggotakan beberapa/banyak anggota) serta sebagai sarana dokumentasi (bermanfaat untuk menelaah perilaku sistem secara seksama dan bermanfaat untuk menguji (*testing*) sistem yang telah selesai dikembangkan).

II.11.1. 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)



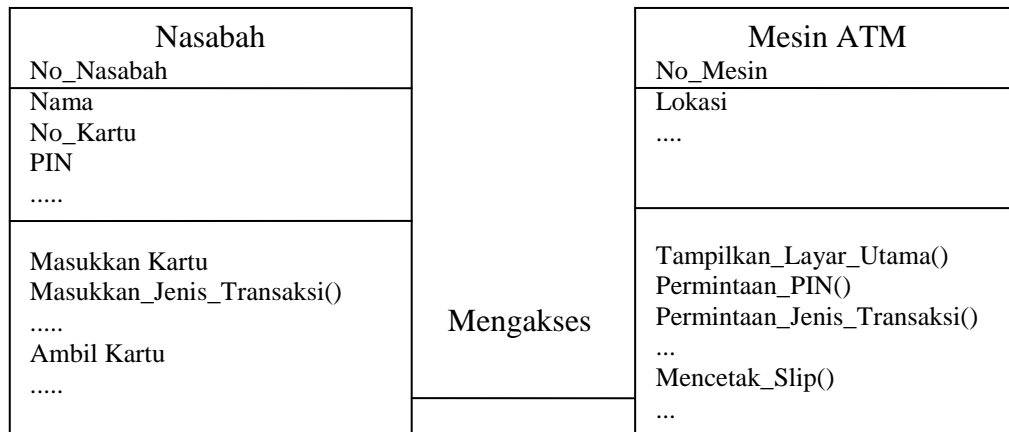
Gambar II.1. Contoh Use Case Diagram
(Sumber : Adi Nugroho ; 2009 : 8)

II.11.2. 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 :

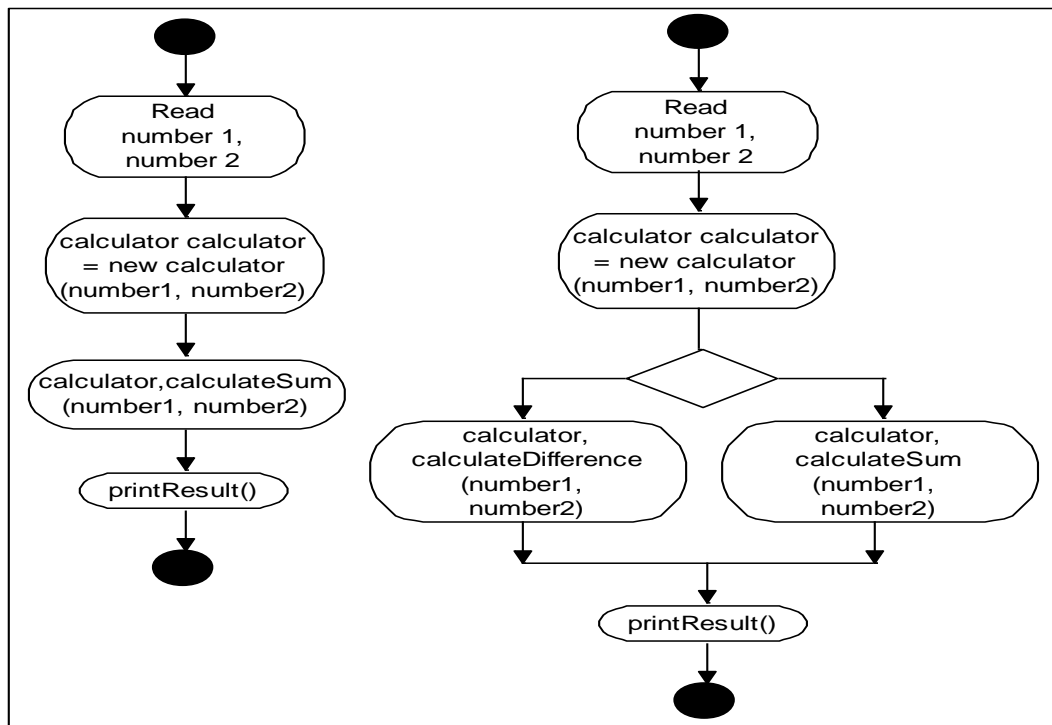
- a. Memudahkan untuk mempelajari secara seksama hal-hal yang ada di dunia nyata.
- b. Menyediakan suatu dasar yang kuat dalam implementasi ke dalam sistem terkomputerisasi (Adi Nugroho ; 2009 :17).



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

II.11.3. 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).

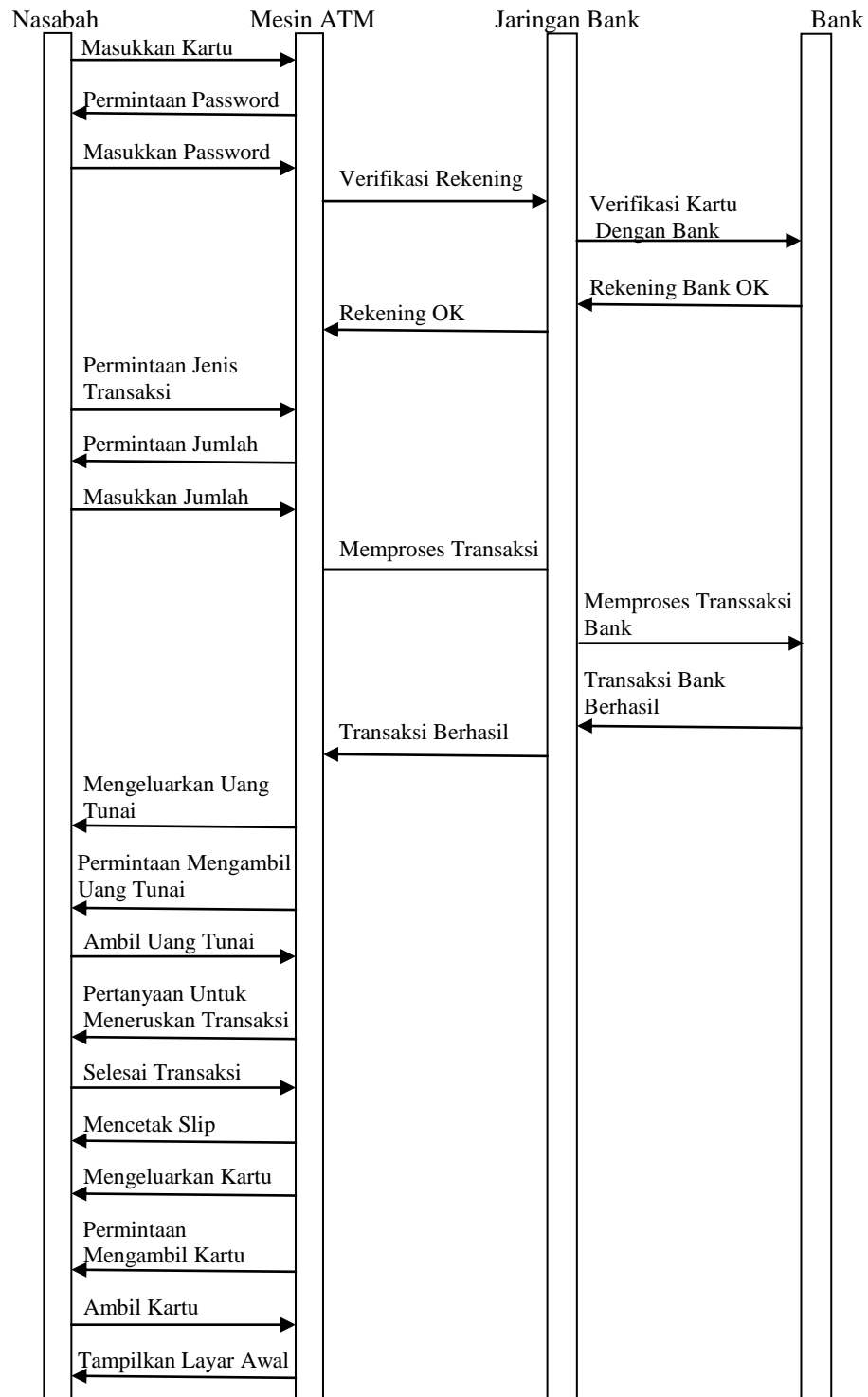


Gambar II.3. Contoh Activity Diagram
(Sumber : Adi Nugroho ; 2009 : 74)

II.11.4. 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.



**Gambar II.4. Contoh Sequence Diagram
(Adi Nugroho ; 2009 : 36)**