



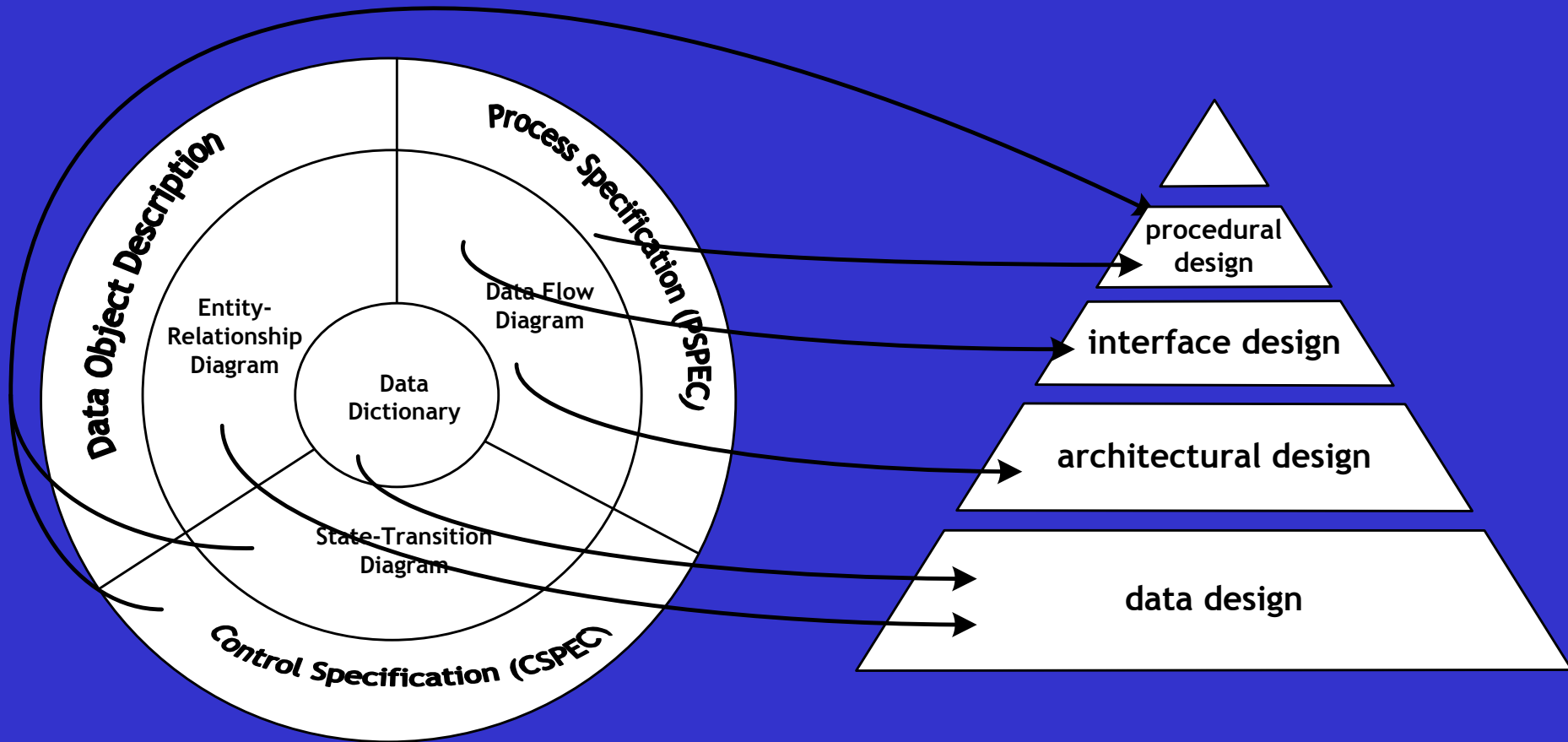
REKAYASA PERANGKAT LUNAK



“Perangkat Lunak ”

Ratna Wardani

Model to Design





Model to Design

- **Data design**

- mengubah model informasi (entity relationship diagram dan data dictionary) menjadi struktur data

- **Architectural design**

- berisi hubungan antar elemen dalam program

- **Interface design**

- menjelaskan bagaimana komunikasi di dalam perangkat lunak, dengan sistem, dan dengan manusia yang menggunakannya.

- Sebuah interface mengandung maksud sebuah aliran informasi.



Model to Design

- **Procedural design**

→ mengubah elemen struktural dari arsitektur program menjadi deskripsi prosedural dari komponen perangkat lunak



Model to Design

- Sebuah desain harus menunjukkan organisasi secara hirarkis
- Sebuah desain harus bersifat modular; jadi, sebuah perangkat lunak seharusnya dapat dibagi-bagi secara logik menjadi beberapa elemen yang melakukan fungsi atau subfungsi secara spesifik
- Sebuah desain harus mengandung abstraksi data dan prosedural
- Sebuah desain harus mengarah pada modul-modul (prosedur atau subrutin) yang menunjukkan karakteristik fungsional



Model to Design

- Sebuah desain harus mengarah pada antarmuka yang mengurangi kompleksitas hubungan antar modul dan dengan lingkungan luar
- Sebuah desain harus diturunkan menggunakan metode yang berulang yang diarahkan oleh informasi yang dihasilkan pada tahap analisis perangkat lunak



Model to Design

- Proses desain tidak boleh mengalami "*tunnel vision*"
- Desain harus dapat dilacak ke model analisis
- Tidak melakukan desain pada hal yang sama berulang-ulang
- Desain harus merepresentasikan masalah pada keadaan nyata
- Desain harus memperlihatkan keseragaman dan integrasi



Model to Design

- Desain harus terstruktur untuk mengantisipasi adanya perubahan
- Desain bukan coding, coding bukan desain
- Penilaian kualitas desain harus dilaksanakan pada saat desain tersebut dibuat
- Desain harus di-review untuk meminimasi kesalahan konseptual



Konsep Desain

- Abstraksi
- Modularitas
- Arsitektur software
- Hirarki kontrol
- Pembagian struktural
- Data struktur
- Software procedure
- Penyembunyian informasi



Dokumentasi Desain

- I. Lingkup Sistem
- II. Desain Data
- III. Desain Architectural
- IV. Desain Antarmuka
- V. Desain Prosedural
- VI. Catatan Khusus
- VII. Appendix



Data Design

- Mengubah objek data yang didefinisikan pada model analisis menjadi struktur data yang ada dalam perangkat lunak
- Atribut yang dimiliki objek data, hubungan di antara objek data, dan penggunaannya dalam program, semuanya mempengaruhi pemilihan struktur data



Architectural Design

- Menggunakan karakteristik aliran informasi dalam model analisis untuk menghasilkan struktur program
- Sebuah data flow diagram dipetakan menjadi struktur program menggunakan dua pendekatan :
 - Transform mapping
 - Transaction mapping
- Transform mapping : diterapkan untuk sebuah aliran data yang menunjukkan batas yang jelas antara data yang masuk dan yang keluar



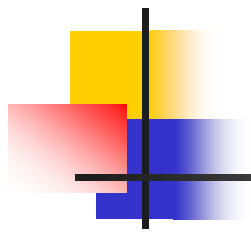
Architectural Design

- DFD dipetakan menjadi sebuah struktur yang mengalokasikan kontrol menjadi input, pemrosesan, dan output bersama dengan hirarki modul
- Transaction mapping : diterapkan jika sebuah item informasi menyebabkan percabangan
- DFD dipetakan menjadi sebuah struktur yang mengalokasikan kontrol menjadi sebuah substruktur yang mendapatkan dan mengevaluasi sebuah transaksi



Interface Design

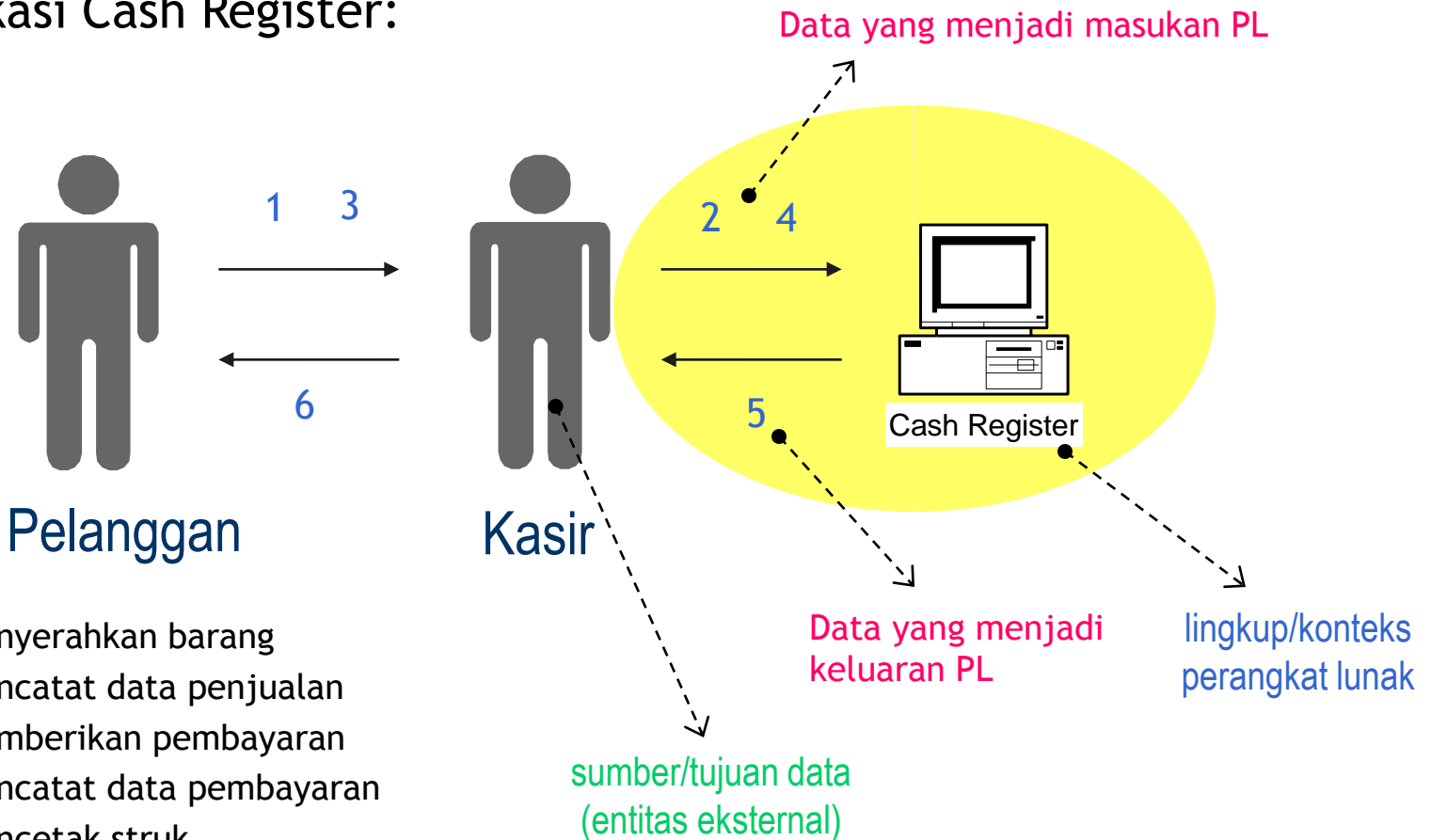
- Meliputi antarmuka program internal dan eksternal serta desain untuk antarmuka pengguna
- Desain antarmuka internal dan eksternal diarahkan oleh informasi yang diperoleh dari model analisis



SADT

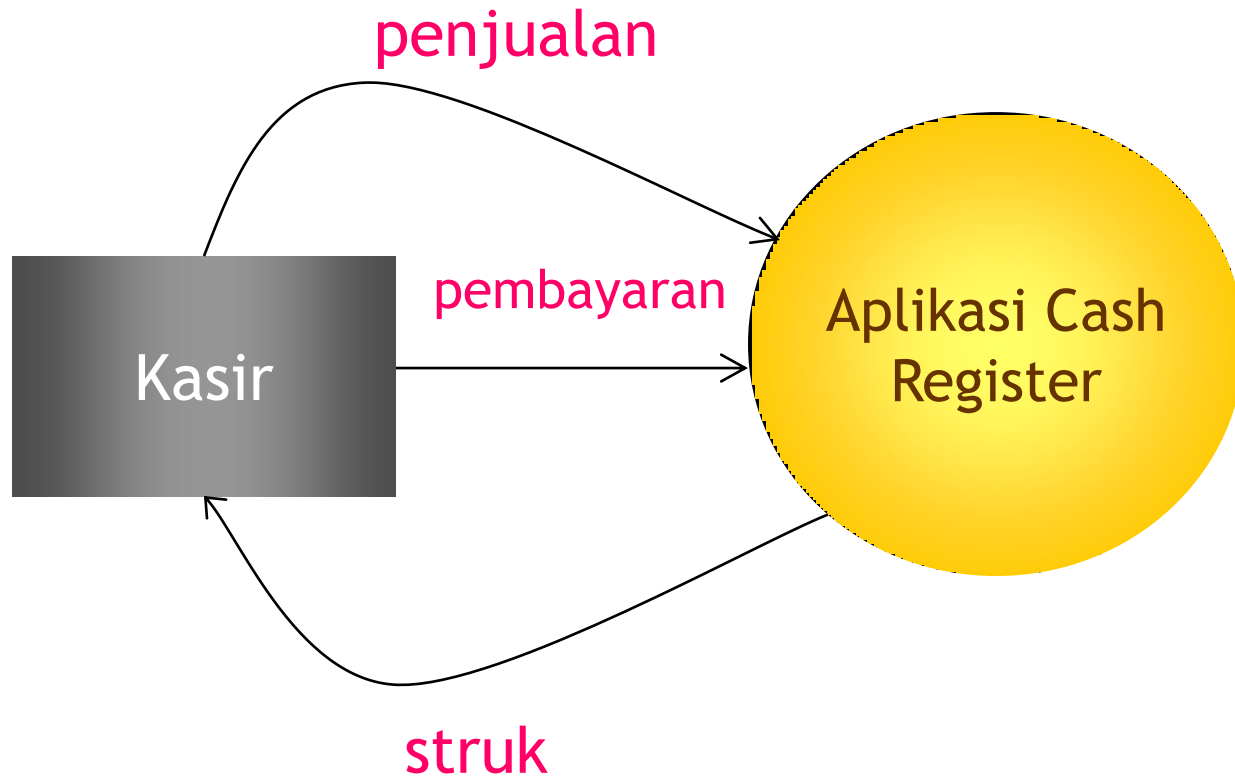
Pemodelan Fungsional dan Aliran Informasi (Analisis Terstruktur)

Aplikasi Cash Register:



1. Menyerahkan barang
2. Mencatat data penjualan
3. Memberikan pembayaran
4. Mencatat data pembayaran
5. Mencetak struk
6. Menerima struk, barang, dan kembalian

Pemodelan Fungsional dan Aliran Informasi (Analisis Terstruktur)





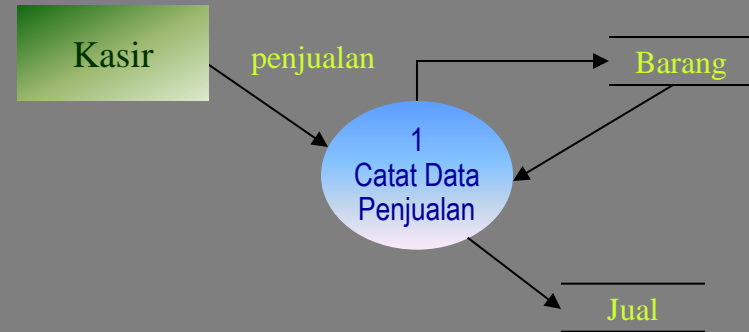
Pemodelan Fungsional dan Aliran Informasi (Analisis Terstruktur)

- Merupakan pemerincian (*break down*) dari Diagram Konteks: level-1, 2, dst.
- Proses-proses yang akan dibuat harus sesuai dengan deskripsi kebutuhan fungsionalnya.
- Alur dan urutan proses mengikuti mekanisme proses pengolahan data yang nanti akan dilakukan oleh perangkat lunak.

Workflow Penjualan Barang



Diagram Aliran Data (DAD)



Kamus Data

- 1. barang yang dibeli
- 2. penjualan = kode_brg + banyak
- 3. Barang = @kode_brg + nama_brg + harga + stok
- 4. Jual = @no_faktur + @kode_brg + banyak

Sketsa Tampilan Layar

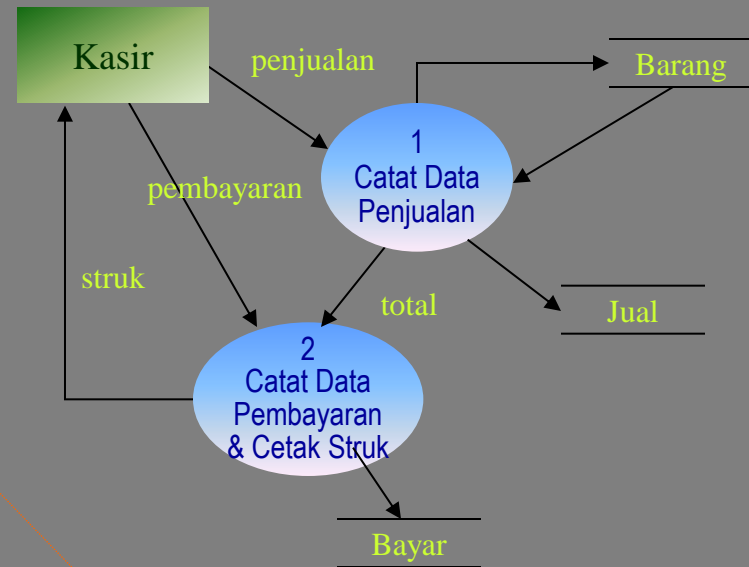
Entry Penjualan Barang X

Kode Barang	BRG-101
Nama Barang	KERTAS A4 80 GR.
Harga (Rp.)	27,500
Banyaknya	2
Jumlah (Rp.)	55,000

Workflow Pembayaran



Diagram Aliran Data (DAD)



Kamus Data

~~1. barang yang dibeli~~

2. penjualan = kode_brg + banyak

3. Barang = @kode_brg + nama_brg + harga + stok

4. Jual = @no_faktur + @kode_brg + banyak

~~5. uang~~

6. pembayaran = jml_bayar

7. Bayar = @no_faktur + tanggal + total

8. struk = no_faktur + tanggal + {nama_brg + harga ± banyak + jumlah} + total + bayar + kembali

~~9. struk, barang dan kembalian~~

total = no_faktur + {kode_brg + nama_brg + harga + banyak} + total

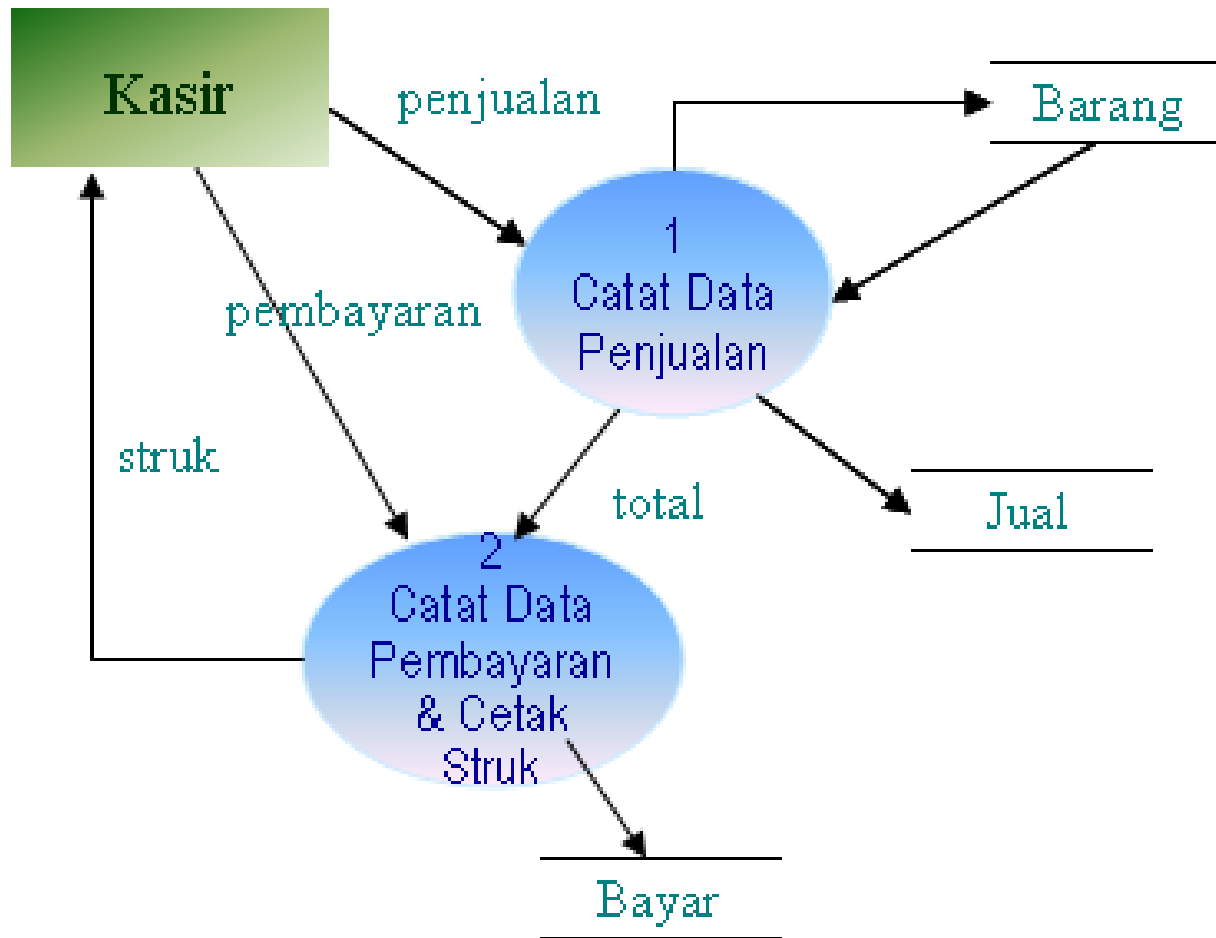
Sketsa Tampilan Layar

Entry Pembayaran
X

Total (Rp.)	55,000
Jumlah Bayar	60,000
Kembali	5,000

Cetak Struk

Pemodelan Fungsional dan Aliran Informasi (Analisis Terstruktur)





Pemodelan Fungsional dan Aliran Informasi (Analisis Terstruktur)

Data Store

1. Barang = @kode_brg + nama_brg + harga + stok
2. Bayar = @no_faktur + tanggal + total
3. Jual = @no_faktur + @kode_brg + banyak

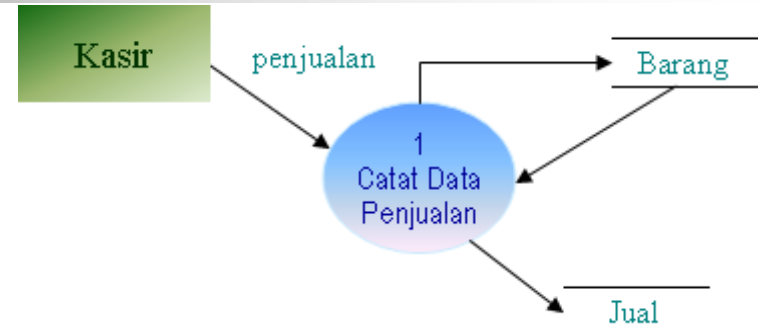
Data Flow

1. pembayaran = jml_bayar
2. penjualan = kode_brg + banyak
3. struk = no_faktur + tanggal + {nama_brg + harga + banyak + jumlah}
+ total + bayar + kembali
4. total = no_faktur + {kode_brg + nama_brg + harga + banyak} + total

Pemodelan Fungsional dan Aliran Informasi (Analisis Terstruktur)

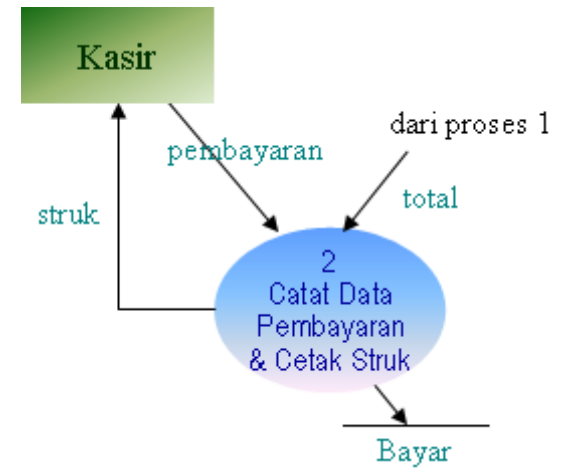
Proses 1: Catat Data Penjualan

1. Baca kode barang
2. Cari dan tampilkan data barang
3. Baca banyak barang;
Hitung dan tampilkan jumlah
4. Rekam data penjualan ke basis data;
Update stok barang



Proses 2: Catat Data Pembayaran & Cetak Struk

1. Hitung dan tampilkan total
2. Baca jumlah bayar;
Hitung dan tampilkan jumlah kembalian
3. Rekam data pembayaran ke basis data
4. Cetak struk



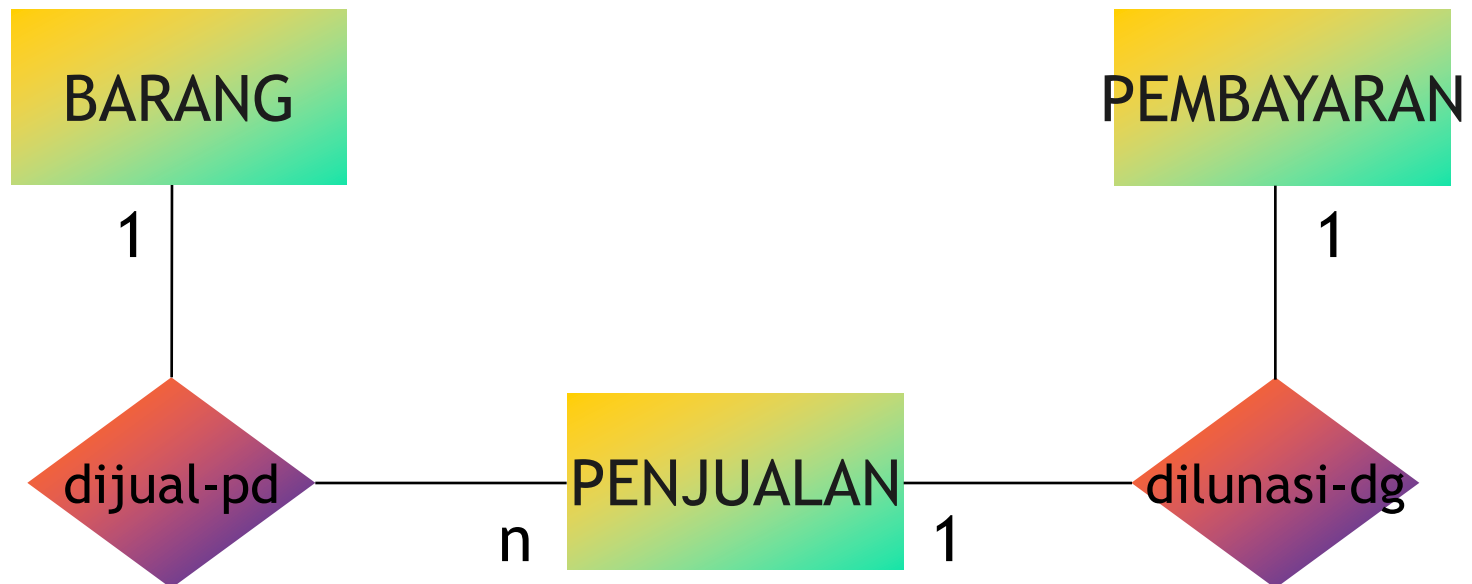


Pemodelan Fungsional dan Aliran Informasi (Analisis Terstruktur)

- Dari DFD yang sudah dibuat, identifikasi data yang akan diolah:
 - Data transaksi penjualan
 - Data transaksi pembayaran
 - Data barang
- Tentukan data mana yang mewakili entitas:
 - Penjualan, pembayaran → event
 - Barang → things
- Tentukan relasi antar entitas.

Pemodelan Fungsional dan Aliran Informasi

ERD (versi Peter Chen)



Pemodelan Fungsional dan Aliran Informasi

ERD (versi James Martin (Conceptual Data Model))

