|  |  |  |  |
| --- | --- | --- | --- |
| C:\Users\Admin\Pictures\JATA KPM.png  **BAHAGIAN PENDIDIKAN DAN LATIHAN TEKNIK VOKASIONAL**  **KEMENTERIAN PENDIDIKAN MALAYSIA**  **ARAS 5 & 6, BLOK E14, KOMPLEKS E,**  **PUSAT PENTADBIRAN KERAJAAN PERSEKUTUAN**  **KERTAS PENERANGAN**  ***(INFORMATION SHEET)*** | | | |
| **KOD DAN NAMA PROGRAM NOSS** | | IT-010-3:2016 PEMBANGUNAN APLIKASI | |
| **TAHAP** | | 3 | |
| **KOD, NAMA CU DAN WA NOSS** | | **CU2/WA3 – PLAN MODULE EXPECTED BEHAVIOUR** | |
| **KOD DAN NAMA PROGRAM KV** | | SISTEM PENGURUSAN PANGKALAN DATA DAN APLIKASI WEB | |
| **KOD DAN NAMA KURSUS KSKV** | | KPD 2023 WEB PROGRAMMING | |
| **NO.DAN TAJUK STANDARD KANDUNGAN KSKV** | | K3 PLAN MODULE EXPECTED BEHAVIOUR | |
| **NO. KOD** | **NO. KOD NOSS** | IT-010-3:2016-C02/P(30/70) | Muka Surat : 1 Drp : 9 |
| **NO. KOD KSKV** | KPD 2023 / P(7/11) |

**TAJUK/***TITLE***:**

**KAEDAH *TEST DRIVEN DEVELOPMENT (TDD)- Pseudo code & Data Flow Diagram***

**TUJUAN/***PURPOSE* **:**

Kertas penerangan ini adalah bertujuan menerangkan mengenai :

* Terangkan kaedah *Test Driven Development (TDD)* seperti:
* Kod Pseudo
* *Data flow diagram*

**3.3 KAEDAH TEST DRIVEN DEVELOPMENT (TDD)**

**3.3.1 Pseudo Kod**

1. Pseudo kod ialah satu set aturan yang ditulis dalam bahasa tabii untuk menyelesaikan sesuatu masalah.
2. Merupakan kod palsu yang mirip kod aturcara tetapi ditulis dalam bahasa pertuturan manusia dan mempunyai nombor turutan.
3. Ia menunjukkan langkah demi langkah untuk melaksanakan sesuatu tugas dari mula hingga tamat.
4. Dibuat sebelum pengkodan aturcara sebenar untuk pastikan kaedah penyelesaian masalah yang digunakan adalah betul.
5. Menggunakan ungkapan matematik, ungkapan logik dan struktur kawalan komputer seperti *IF, ELSE, INPUT, OUTPUT*.
6. Contoh pseudo kod bagi sebuah sistem untuk mengira luas bulatan apabila pengguna memasukkan nilai jejari bulatan.

1. Mula

2. INPUT jejari\_bulatan

3. luas\_bulatan = 3.142 \* jejari\_bulatan \* jejari\_bulatan

4. OUTPUT luas\_bulatan

5. Tamat

**3.3.2 *Data Flow Diagram***

**Definisi:**

* DFD merupakan **gambar pergerakan data antara entiti-entiti luaran dan proses-proses serta storan data dalam sebuah sistem**.

**Fungsi DFD:**

1. DFD merupakan perwakilan model logikal yang menunjukkan APA yang dilakukan oleh sistem dan BUKAN BAGAIMANA ia dilakukan.
2. Memberi indikasi mengenai pergerakan data melalui sistem.
3. Menggambarkan fungsi-fungsi sistem.

**Simbol DFD:**

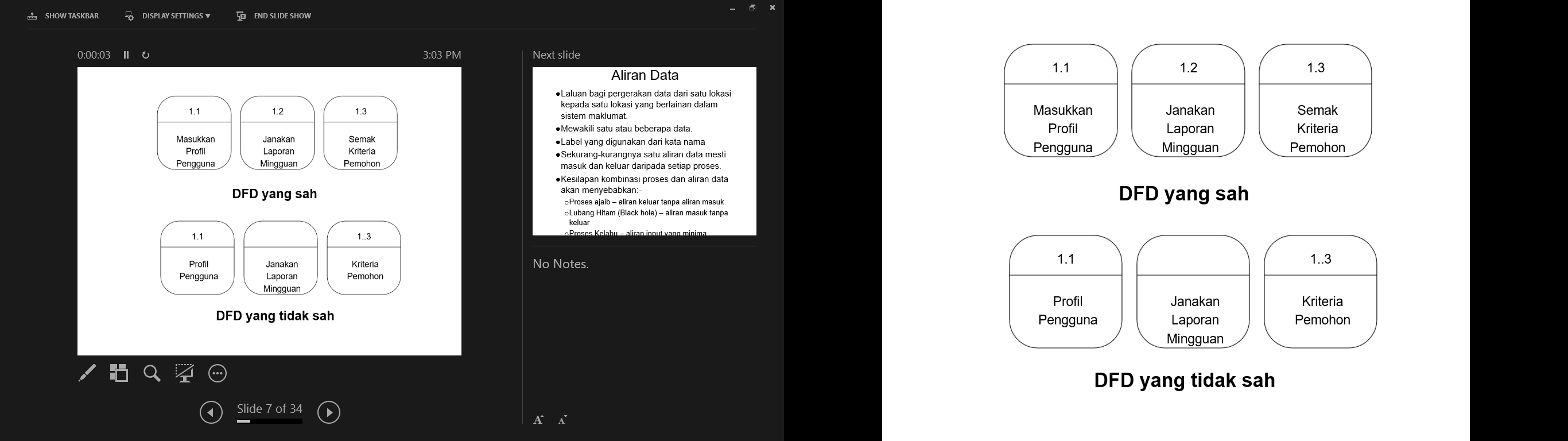
1. Dilukis menggunakan 4 unsur utama iaitu:
2. Entiti luaran
3. Proses
4. Aliran data
5. Storan data
6. Terdapat 2 set simbol yang biasa digunakan:

|  |  |  |
| --- | --- | --- |
| **DeMarco & Yourdan** | **MAKSUD** | **Gane & Sarson** |
|  | Proses  Storan data  Sumber  Aliran data |  |

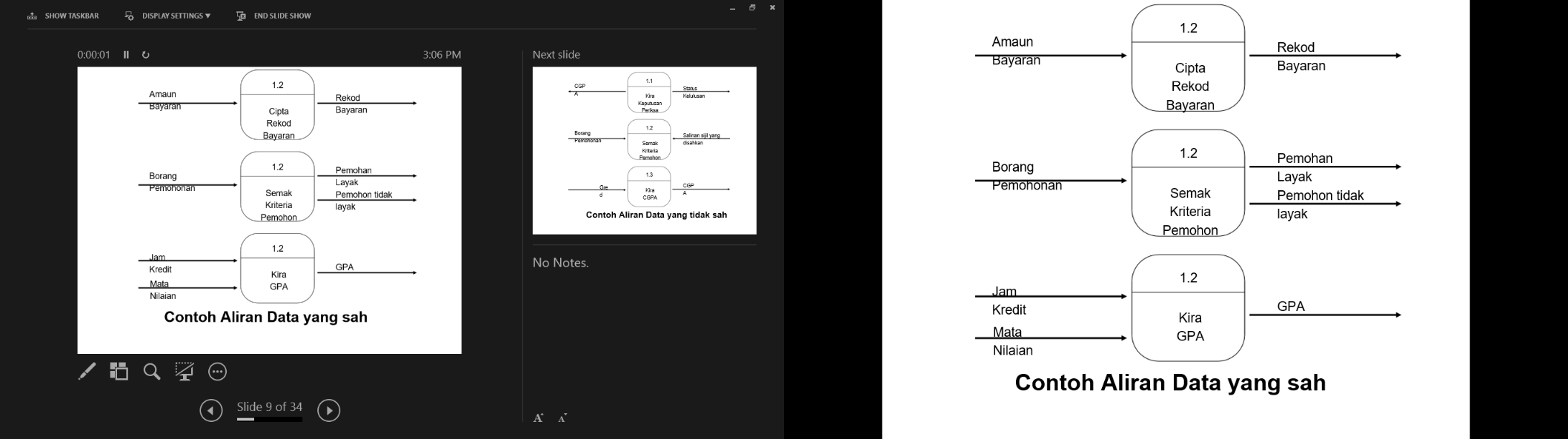
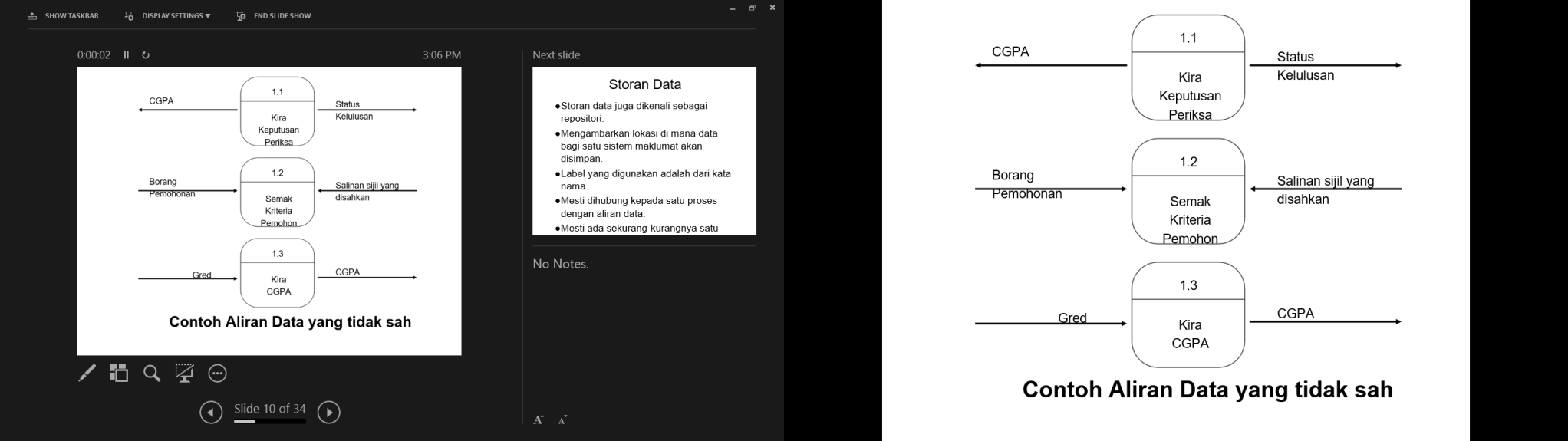
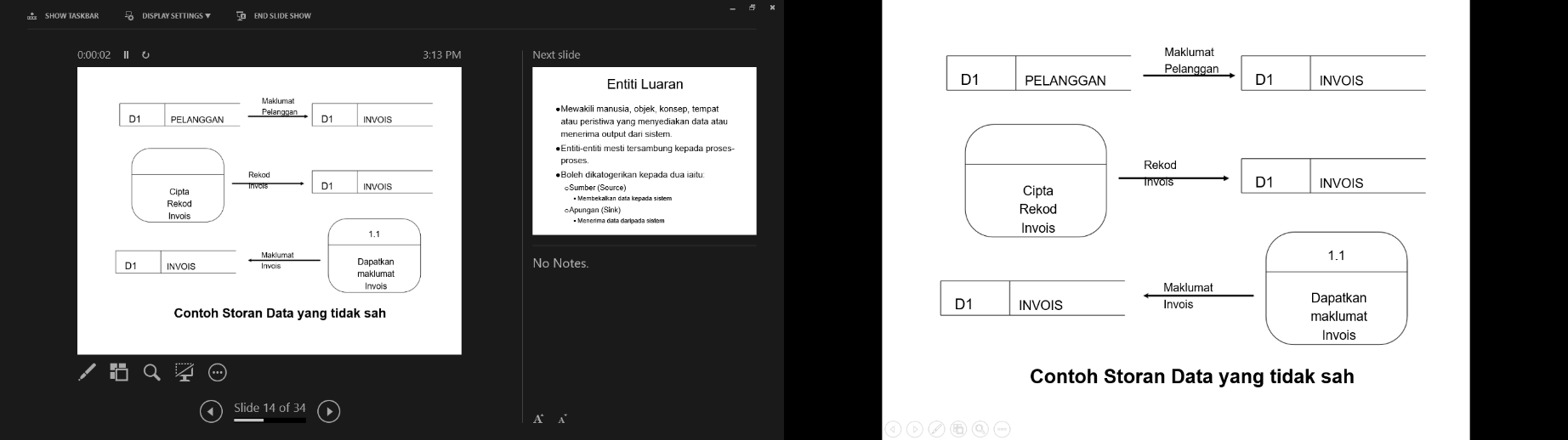
Rajah 1: Simbol dalam *Data Flow Diagram*

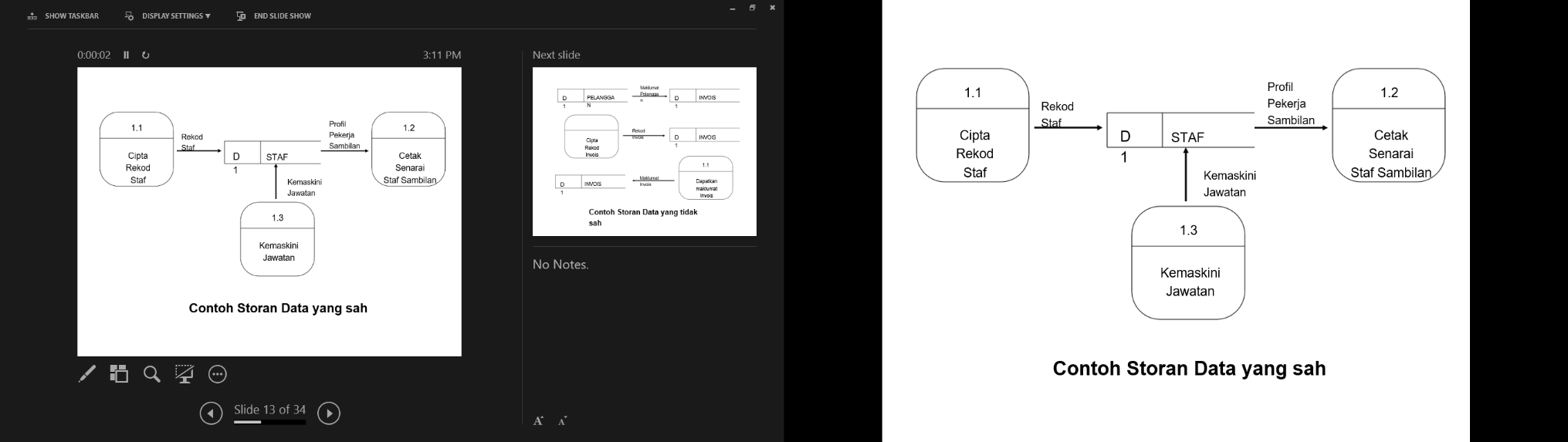
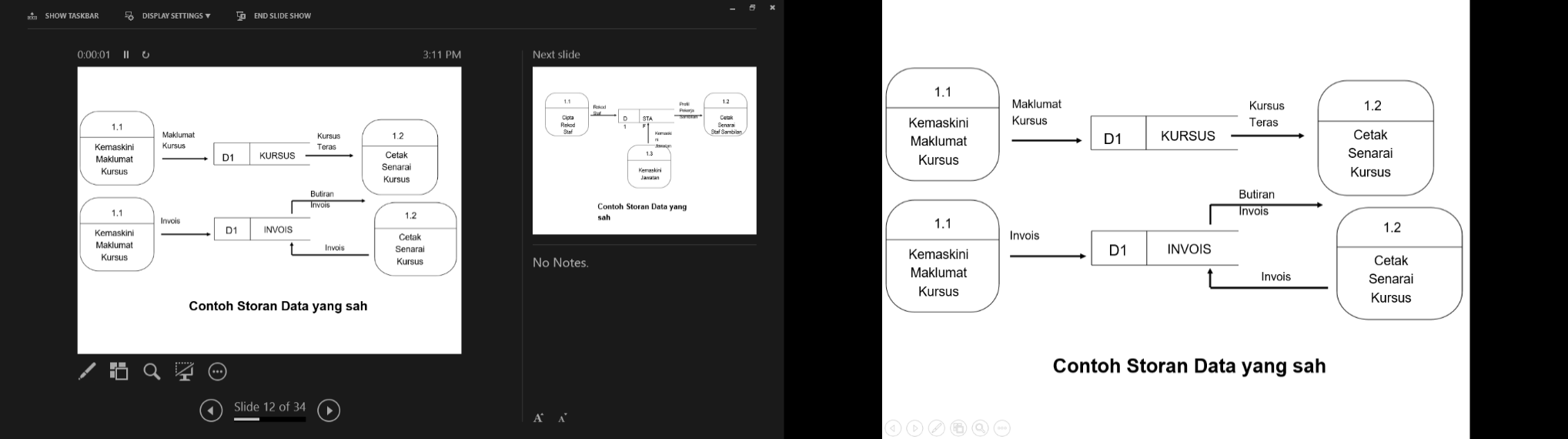
1. Proses
2. Proses akan menerima input dan menghasilkan output.
3. Output tidak boleh sama dengan input dari segi bentuk atau kandungan atau kedua-duanya.
4. Didokumentasikan dengan deskripsi proses.
5. Butiran terperinci ditunjukkan dalam proses deskripsi.
6. Nama untuk melabel proses merupakan kata kerja (*verb*).
7. Setiap proses diberikan nombor mengikut paras DFD.

Contoh rajah:

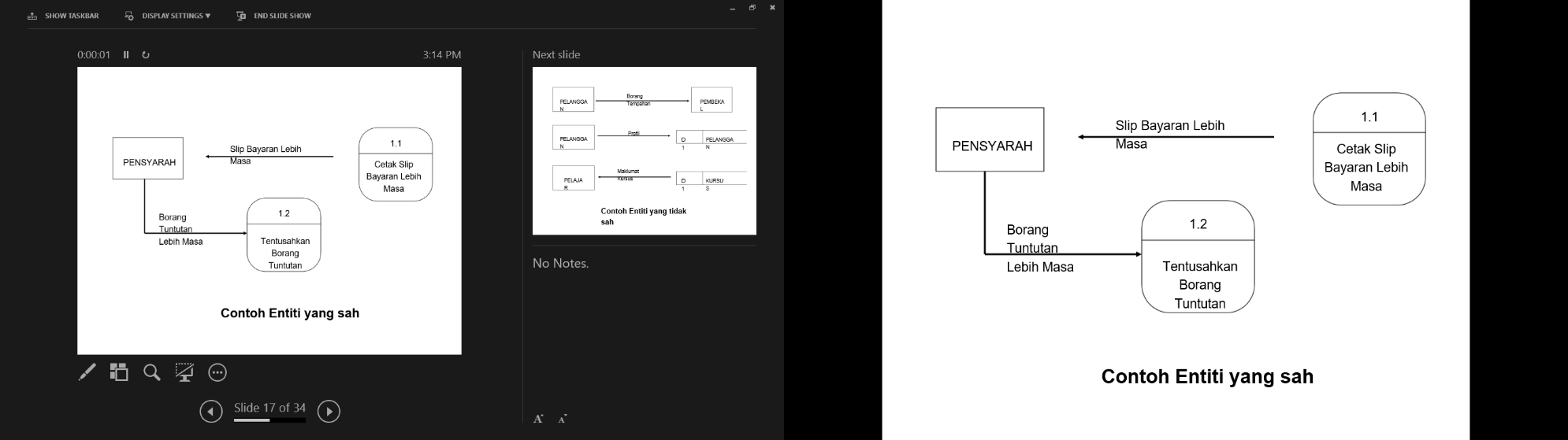
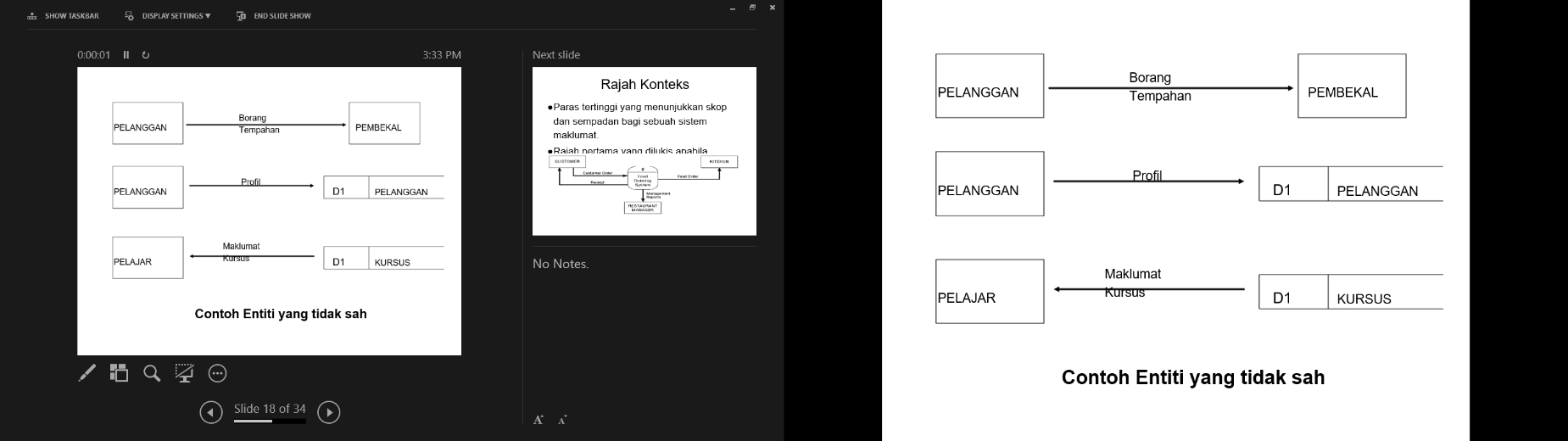


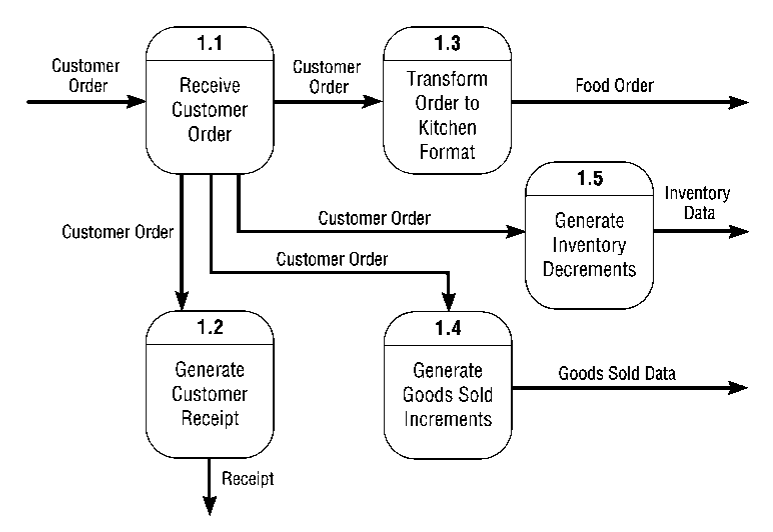
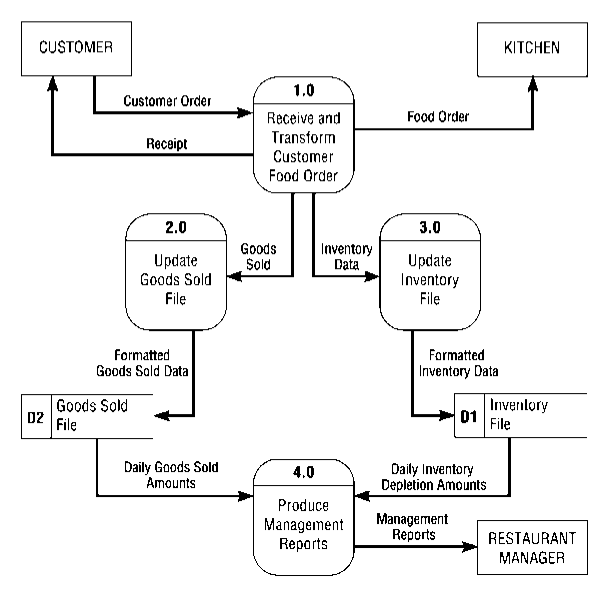
Bukan kata kerja

1. Aliran data
2. Laluan bagi pergerakan data dari satu lokasi kepada satu lokasi yang berlainan dalam sistem maklumat.
3. Mewakili satu atau beberapa data.
4. Label yang digunakan dari kata nama
5. Sekurang-kurangnya satu aliran data mesti masuk dan keluar daripada setiap proses.
6. Kesilapan kombinasi proses dan aliran data akan menyebabkan:-
   1. Proses ajaib – aliran keluar tanpa aliran masuk
   2. Lubang Hitam (*Black hole*) – aliran masuk tanpa keluar
   3. Proses Kelabu – aliran input yang minima menghasilkan output dari proses
7. Storan data
8. Storan data juga dikenali sebagai repositori.
9. Mengambarkan lokasi di mana data bagi satu sistem maklumat akan disimpan.
10. Label yang digunakan adalah dari kata nama.
11. Mesti dihubung kepada satu proses dengan aliran data.
12. Mesti ada sekurang-kurangnya satu data masuk dan data keluar



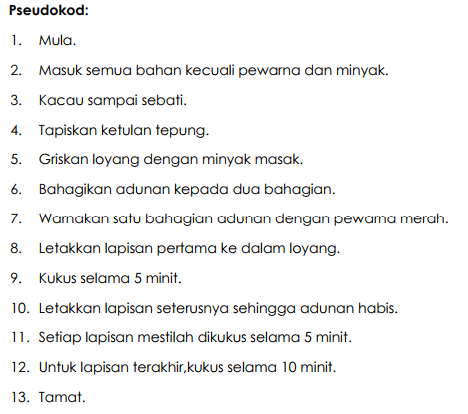
1. Entiti luaran
2. Mewakili manusia, objek, konsep, tempat atau peristiwa yang menyediakan data atau menerima output dari sistem.
3. Entiti-entiti mesti tersambung kepada proses-proses.
4. Boleh dikatogerikan kepada dua iaitu:
   1. Sumber (*Source*)
      * **Membekalkan data** kepada sistem
   2. Apungan (*Sink*)



1. Rajah Paras Rendah
2. Perincian yang lebih mendalam
3. Rajah yang dihasilkan dari pemecahan Rajah Sifar dinamakan Rajah Paras Satu
4. Pemecahan proses perlu mengambil kira:
   1. Pemarasan (*Leveling*)
      * Proses melukis DFD secara berperingkat-peringkat iaitu bermula dari paras yang lebih tinggi kepada paras yang lebih rendah.
      * Aktiviti pemecahan proses-proses
      * Satu **proses induk mesti menghasilkan sekurang-kurang dua proses anak**.
      * Proses anak akan dinomborkan mengikut proses induk
      * Contoh
        1. Induk 1.0
        2. Anak 1.1, 1.2, 1.3
   2. Penyeimbangan (*Balancing*)
      * **Pemeliharaan dan pengekalan aliran data input dan output** bagi proses paras tinggi ke dalam proses-proses paras rendah.

**3.4 ALGORITMA (*ALGORITHM*)**

1. Algoritma merupakan satu set arahan / tatacara langkah demi langkah untuk menyelesaikan masalah. Langkah-langkah yang ditulis adalah terperinci, senang difahami dan mesti mempunyai kesudahan / penamatan.
2. Contoh algoritma:



**Rajah 9**

1. Terdapat beberapa cara untuk menterjemahkan algoritma, antaranya adalah:

* Carta alir (*flowchart*)
* Pseudo kod (*pseudocode*) - \* rujuk bahagian 2.3.1

**3.5 CARTA ALIR (*FLOWCHART*)**

1. Menggunakan simbol grafik untuk mewakili arahan-arahan penyelesaian masalah.
2. Terdiri daripada nod dan garis penghubung.
3. Setiap elemen dalam carta alir mempunyai simbol standard dan fungsi tertentu.
4. Berikut merupakan simbol standard yang lazim digunakan:

|  |  |  |
| --- | --- | --- |
| **Simbol** | **Nama Nod** | **Fungsi** |
|  | Terminal  *Terminator* | Permulaan / penamatan carta alir |
|  | Input / Output | Membaca data yang dimasukkan (input) atau memaparkan data / hasil (output) |
|  | Proses  *Process* | Apa-apa jenis operasi pemprosesan data seperti operasi matematik, operasi logik, perpindahan data, penghapusan data dan lain-lain |
|  | Syarat / Pengujian  *Decision* | Menguji syarat yang terkandung dalam simbol berkenaan. Aliran akan ditentukan melalui keputusan syarat sama ada Benar (*True*) atau Palsu (*False*) |
|  | Penghubung  *Connector* | Titik sambungan yang menghubungkan carta alir yang terpisah dalam satu muka surat yang sama. |
|  | Penghubung  *Off-page Connector* | Titik sambungan yang menghubungkan carta alir yang terpisah ke muka surat yang berlainan. |
|  | Aliran Aktiviti  *Flow Line* | Menghubungkan nod-nod untuk menunjukkan aliran proses. |

Jadual 1

1. Contoh carta alir bagi sebuah sistem untuk mengira luas bulatan apabila pengguna memasukkan nilai jejari bulatan:

Mula

Tamat

OUTPUT

luas\_bulatan

INPUT

jejari\_bulatan

luas\_bulatan = 3.142 \* jejari\_bulatan \* jejari\_bulatan

Carta alir : Luas bulatan

**SOALAN/***QUESTION***:**

1. Senaraikan simbol-simbol yang digunakan dalam carta alir.

**RUJUKAN/***REFERENCE* **:**

1. http://www.slideshare.net/DeclanChellar/activity-diagram-tutorial ,

Activity Diagram, 20 September, 09:15

2. Chua siwe Geck et.al (2016) .Buku Sains Komputer Tingkatan 4,KPM:

Oxford Fajar Sdn.Bhd. Kuala Lumpur. ISBN NO.978-983-47-2013-1.

Mukasurat:23 hingga 61