|  |  |  |
| --- | --- | --- |
| 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 NOSS** | 3 | |
| **KOD, NAMA CU DAN WA NOSS** | **CU03 / WA1 - INTERPRET MODULE INTEGRATION REQUIREMENT** | |
| **NAMA PROGRAM KV** | SISTEM PENGURUSAN PANGKALAN DATA DAN APLIKASI WEB | |
| **KOD DAN NAMA KURSUS KSKV** | KPD 3024 APPLICATION MODULE INTEGRATION | |
| **NO.DAN TAJUK KOMPETENSI** | K1 INTERPRET MODULE INTEGRATION REQUIREMENT | |
| **NO. KOD NOSS** | IT-010-3:2016 – C03 /P(2/19) | Muka Surat : 1 Drp : 5 |
| **NO. KOD KSKV** | KPD 3024 / P(2/12) |

**TAJUK:**

**KEPERLUAN INTEGRASI MODUL**

**TUJUAN:**

Kertas penerangan ini adalah bertujuan menerangkan mengenai :

1. Senaraikan keperluan integrasi modul
2. Senaraikan aras integrasi
3. Terangkan maksud titik integrasi

**PENERANGAN/***INFORMATION* **:**

**K1 INTERPRET MODULE INTEGRATION REQUIREMENT**

P3 Integration Requirement

1. Module to be integrated

Pengumpulan modul atau sub sistem sebelum di integrasikan perlu menitikberatkan ciri-ciri yang berikut:

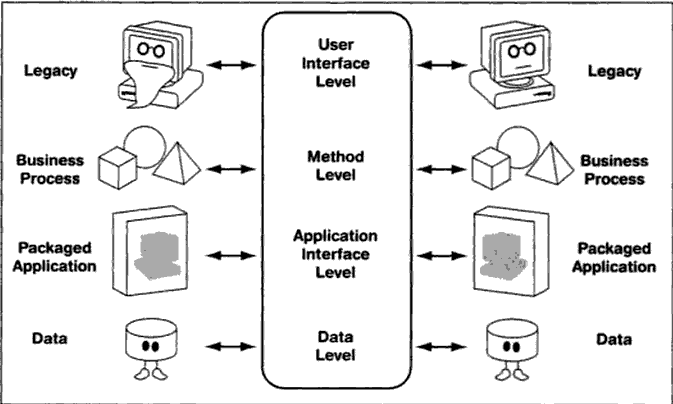
* + Nama fail yang ringkas
  + Versi
  + Bahasa pengaturcaraan

Jenis-jenis modul yang terlibat antaranya adalah:

* Email
* Domain
* Pangkalan data
* Fail
* Dan sebagainya

1. Level of integration

Pembangunan sesuatu sistem melalui beberapa proses. Pembangun sistem perlu memahami proses pengintegrasian yang terdiri daripada *data level, application interface level, method level* dan *user interface level.*



Rajah 3: Proses Pengintegrasian

* **Data Level** adalah proses pemindahan data antara data simpanan. Ia dijelaskan sebagai pengambilan maklumat daripada satu pangkalan data, memproses dan mengemaskini maklumat yang sama dalam pangkalan data yang lain. Kebaikan data level adalah kos yang rendah kerana tidak akan berlaku perubahan kod dan tidak memerlukan pengujian pada sistem.
* **User interface level** adalah kaedah yang lebih primitive. Pembangun sistem boleh mengumpulkan aplikasi menggunakan *user interface* sebagai titik integrasi yang sama. Sebagai contoh, sistem yang tidak dapat akses ke pangkalan data, boleh masuk melalui *user interface* bagi sistem tersebut.
* **Method level** adalah perkongsian *business logic.* sebagai contoh*,* pengemaskinian rekod pelanggan boleh di akses dari pelbagai aplikasi, aplikasi lain boleh mengakses tanpa perlu menulis kembali *method* untuk membuat capaian maklumat.
* **Application Interface level** digunakan oleh pembangun untuk mengakses proses dan data perniagaan. Tujuan utama adalah untuk menghubungkan aplikasi dan bertukar maklumat antara kedua-dua perniagaan.

1. Integration point

Titik integrasi adalah suatu antara muka yang digunakan apabila perisian menerima input daripada pengguna. Ia berlaku apabila sistem mengendalikan interaksi dengan sistem luar. Sebagai contoh halaman *log in* diklasifikasikan sebagai *integration point* apabila sistem menerima *user id* dan *password*.

**Metode**  
Metode dalam melakukan integrasi aplikasi dapat menggunakan empat metode, antara lain data level, application interface level, method level, dan user interface level.

1. **Data Level**  
   Metode ini merupakan suatu proses, teknik dan teknologi yang memungkinkan perpindahan data melalui data store/ database system. Metode ini melakukan ekstraksi data dari satu sistem database, lalu memproses data tersebut jika diperlukan, dan melakukan updating pada sistem database yang lain. Data Level EAI digunakan ketika aplikasi tidak menyediakan APIs atau client interfaces. Dalam melakukan metode data level EAI, dibutuhkan pemahaman yang baik terkatir operasional bisnis yang mungkin dapat mempengaruhi application’s data model. Kelebihan dalam metode ini adalah dapat meminimalisir biaya dan tidak mengubah code aplikasi. Infrasturktur yang dapat digunakan dalam metode ini antara lain Distributed DBMS dan ETL Tools dan contoh proyeknya akan dipublish di postingan selanjutnya.
2. **Application Interface Level**  
   Dalam menerapkan metode ini, seorang developers dapat menggabungkan (bundle) banyak application secara bersama dan  melakukan sharing logika bisnis dan informasi. Beberapa packaged application yang sering digunakan antara lain, SAP, PeopleSoft, Baan, dan lain-lain). Metode ini menggunakan interface untuk mengakses proses dan data, mengekstraksi informasinya, dan meletakkannya ke dalam suatu format yang dapat dimengerti oleh target aplikasi dan mengirimkan informasinya. Infrastruktur yang dapat digunakan pada metode ini adalah API (Application Programming Interface), dan contoh proyek akan dipublish di postingan selanjutnya.
3. **Method Level**  
   Pada umumnya, metode ini digunakan ketika masing-masing aplikasi yang akan diintegrasikan memiliki kesamaaan functional methods. Misalkan suatu metode unuk melakukan update customer record dapat diakses oleh berbagai macam aplikasi dan aplikasi tersebut dapat mengakses metode tersebut tanpa melakukan penulisan ulang (rewrite) masing-masing metode tersebut. Contoh penerapannya adalah distributed component atau CORBA. Infrastruktur yang dapat digunakan pada metode ini adalah distributed object/ component framework.
4. **User Interface Level**Metode ini menggabungkan aplikasi dengan menggunakan user interface (UI) mereka sebagai common point dalam integrasi atau biasa disebut screen scraping. Metode ini digunakan ketika suatu sistem tidak memungkinkan untuk dilakukan integrasi melalui data level, aplikasi level mahupun method level. Selain itu, metode ini juga dapat digunakan ketika business logic tertanam dalam user interfacenya. Infratruktur yang dapat digunakan dalam metode ini adalah screen scraper, dan contoh proyeknya akan dipublish di postingan selanjutnya.

**SOALAN** / *QUESTION*:

1. Apakah keperluan integrasi modul?
2. Apakah aras dalam integrasi modul?

**RUJUKAN** / *REFERENCE*:

1. Paulheim, H. (2011). Application Integration on the User Interface Level. *Ontology-Based Application Integration*, 9–25. doi: 10.1007/978-1-4614-1430-8\_2