|  |  |  |
| --- | --- | --- |
| 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 NOSS** | IT-010-3:2016 PEMBANGUNAN APLIKASI | |
| **TAHAP** | 3 | |
| **KOD, NAMA CU DAN WA NOSS** | **CU03 / WA4 - COMMIT MODULE INTEGRATION CODE** | |
| **NAMA PROGRAM KV** | SISTEM PENGURUSAN PANGKALAN DATA DAN APLIKASI WEB | |
| **KOD DAN NAMA KURSUS KSKV** | KPD 2042 *BACKUP* STRATEGIES MANAGEMENT | |
| **NO.DAN TAJUK STANDARD KANDUNGAN KSKV** | K1 ANALYSE BACKUP REQUIREMENT | |
| **NO. KOD NOSS** | IT-010-4:2016-C03/P(14/19) | Muka Surat : 1  Drp : 8 |
| **NO. KOD KSKV** | KPD 2042 / P (1/7) |

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

**FUNGSI SANDARAN KOD SUMBER DAN SANDARAN PANGKALAN DATA**

**TUJUAN/** *PURPOSE:*

1. Terangkan sandaran kod sumber

* SCM
* Version Control

2. Terangkan sandaran pangkalan data.

* Files
* Filesystem
* Live data
* Metadata
* Log

1. ***BACKUP* (SANDARAN)**

Sandaran atau biasa disebut sebagai *Backup* merupakan suatu proses merujuk kepada menyalin data yang tersimpan pada storan komputer dari satu lokasi storan ke lokasi storan lain.Data atau kumpulan maklumat tersebut berbentuk seperti fail, dokumen, gambar, video, audio, sistem komputer, *driver* atau perisian tertentu. *Backup* biasanya dilakukan dengan tujuan tertentu seperti:

1. Melindungi keselamatan data daripada bencana dan kerosakan storan secara fizikal seperti banjir, kebakaran dan lain-lain
2. Melindungi kerosakan data akibat serangan ancaman digital seperti virus, *malware* atau *spyware*
3. Mengembalikan data dengan mudah sekiranya berlaku kerosakan atau kehilangan data di mana data asal boleh diperolehi daripada media sandaran tanpa perlu membuatnya semula.
4. Mendapatkan data yang lama semula sekiranya diperlukan

*Backup* perlu dilakukan apabila berhadapan dengan situasi berikut:

1. Apabila mempunyai aplikasi komputer baru yang ingin dimasukkan ke dalam server yang sudah mempunyai aplikasi lain di dalamnya
2. Menaiktaraf perisian atau menambah perkakasan ke atas server (*upgrade*)
3. Apabila menaiktaraf atau mengemaskini produk perisian itu sendiri (*update patches*). Proses sandaran perlu dilaksanakan dan di dokumentasikan supaya dapat mengelakkan kehilangan data terkini dan memudahkan kerja *Restore* jika diperlukan
4. Proses sandaran berkala secara berkala samada harian, mingguan atau bulanan bergantung kepada kategori data samada data kritikal atau tidak kritikal.
5. Sebelum menghantar komputer untuk dibaiki atau di format

Bagi produk perisian pula, terdapat 2 kategori *backup* yang perlu diberi perhatian oleh pembangun perisian iaitu ;

1. sandaran terhadap **kod sumber**
2. sandaran terhadap **pangkalan data**

***1.1 BACKUP* KOD SUMBER (*SOURCE CODE*)**

Kod sumber adalah komponen asas program komputer yang dicipta oleh seorang pengaturcara. Ia boleh dibaca dan mudah difahami oleh manusia. Apabila seorang pengatur cara menaip sebuah urutan teks dalam pernyataan bahasa HTML ke *Windows Notepad*, maka, teks di dalam fail itu dikatakan mengandungi kod sumber. Kod sumber akan menghasilkan sebuah produk perisian seperti sistem berasaskan laman web dan juga aplikasi mudah alih.

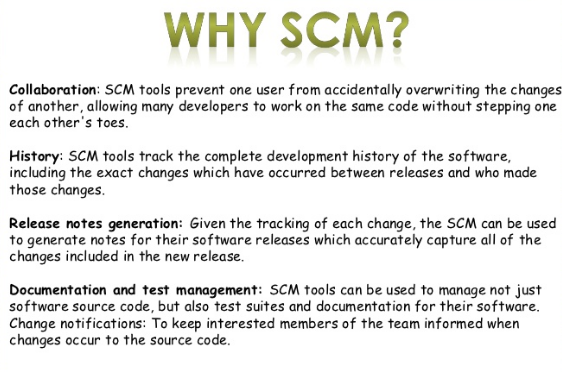
Semasa kod sumber dibangunkan, penyimpanan failnya perlu diurus dengan baik. Pembangun perlu peka dengan setiap perubahan terhadap kod sumber. Apalagi jika kod sumber yang melibatkan lebih daripada seorang pembangun. Kod sumber yang telah dilepaskan kepada pengguna jarang melalui proses *backup* oleh pengguna selain salinan yang disimpan oleh pembangun.

***Source Code Management* (SCM)**

* Juga dikenali sebagai ***source code maintenance*** / ***revision control*** / ***version control***.
* Merupakan pengurusan terhadap perubahan yang terjadi pada *source code* dan dokumen-dokumen lain semasa proses pembangunan perisian.
* Dalam proses pembangunan sistem, kumpulan pembangun sistem biasanya terdiri daripada beberapa orang pengaturcara. Oleh yang demikian, *source code* mungkin diubah oleh lebih daripada seorang pengaturcara.
* *Revision control* merupakan proses mengurus, merekod dan menjejaki perubahan yang telah dilakukan ke atas sistem / perisian berkenaan.

**Fungsi utama pengurusan kod sumber adalah untuk menyediakan pengurusan fail dan *version control* supaya ahli pasukan sentiasa peka dengan perubahan yang berlaku, dan hanya versi fail terbaru dikenal pasti untuk digunakan dalam ruang kerja**.

* Proses *revision control* dapat dipermudahkan dengan menggunakan perisian khas yang dikategorikan di bawah ***Version Control System***.
* Contoh perisian version control:
  + - Git
    - Mercurial
    - Subversion
    - Source Code Control System (SCCS)
    - Vesta
    - AutodeskVault
    - Visual Studio Team Services (VSTS)
    - BitKeeper
* Terminologi dalam SCM:
  + - *Source code repository*
      * Merupakan struktur data yang mengandungi metadata bagi fail-fail yang disimpan dalam komputer.
      * *Repository* mengandungi metadata seperti:
        + Sejarah dan rekod perubahan data
        + Objek yang dihantar
        + Rujukan kepada objek yang dihantar
      * Merupakan pangkalan data / *server* yang menyimpan semua kod sumber.
* *Working copy* 
  + - * Fail setempat (*local files*) di mana pengaturcara membuat perubahan pada kod sumber.
      * Terminologi ***check out*** juga digunakan untuk menunjukkan pengaturcara membuat salinan fail daripada *repository* untuk dijadikan *working* *copy*.
* *Branching*
  + - * Merupakan penduaan objek supaya sebarang perubahan yang dibuat akan berlaku kepada semua objek pada masa yang sama.



***Version Control***

* Apakah version control, dan mengapa anda perlu mengambil berat? Version Control adalah sistem yang merekod sebarang perubahan pada fail atau set fail supaya pada bila-bila masa anda boleh kembali ke satu versi fail. Sebagai contoh, anda akan menggunakan kod sumber perisian sebagai fail kawalan versi, walaupun pada hakikatnya anda boleh melakukan ini pada hampir semua jenis fail pada komputer.
* Jika anda seorang pereka grafik atau pereka web dan anda ingin menyimpan setiap versi imej atau susun atur yang anda buat, Version Control adalah penyelesaian yang bijak untuk digunakan. Sistem ini membolehkan anda memulihkan fail anda ke keadaan sebelumnya, memulihkan keseluruhan projek dalam keadaan sebelumnya, dan banyak lagi. Menggunakan VCS bermakna jika anda melakukan kerosakkan pada fail atau kehilangannya, anda dengan mudah boleh mendapatkannya semula.
* Kebanyakan Version Control melibatkan konsep berikut:-

**Persediaan Asas-**

* **Repositori (repo)**: Pangkalan data menyimpan fail.
* **Server**: Komputer yang menyimpan repo.
* **Client**: Komputer yang menyambungkan ke repo.
* **Set Kerja / Salinan Kerja**: Direktori fail setempat anda, di mana anda membuat perubahan.
* **Trunk / Main**: Lokasi utama untuk kod di repo.

***1.2 BACKUP* PANGKALAN DATA**

*Backup* terhadap pangkalan data perlu dilakukan dalam beberapa keadaan iaitu

1. Semasa proses pembangunan perisian, pangkalan data yang kosong (tanpa data) perlu disalin supaya dapat dipasang semasa dilepaskan kepada pengguna dengan keadaan yang belum dimasukkan sebarang data. Biasanya pembangun akan memasukkan data *dummy* untuk menguji sistem yang dibangunkan.
2. Pangkalan data yang siap dipasang untuk digunakan oleh pengguna akhir tentunya mengandungi data yang sebenar. Data-data ini penting untuk dilaksanakan proses *backup pada* waktu tertentu supaya dapat dikembalikan semula sekiranya berlaku sebarang kerosakan pada data.
3. Data yang terlalu lama yang tidak perlu dipamerkan lagi kepada pengguna juga perlu diurus dengan baik. Data ini perlu disalin dan disimpan dengan selamat supaya dapat disediakan atau dikembalikan jika diperlukan pada bila-bila masa.

Terdapat beberapa kaedah untuk melaksanakan *backup* pangkalan data antaranya ialah :

1. *Full backup : Backup* keseluruhan data yang terlibat. Kaedah ini sesuai untuk *backup* keseluruhan server, pangkalan data dan maklumat serta dokumen yang bukan sulit.
2. *Incremental backup :* Bagi fail-fail yang kerap kali perlu dipindah. Teknik ini tidak memerlukan pengguna membuka fail yang banyak dan sukar dikenalpasti. Jika ada perubahan data dalam fail tersebut, pengguna hanya perlu overwrite data tersebut dalam fail yang sama tanpa perlu buka fail name baru berulang kali
3. *Differential backup : Backup* hanya data yang berubah sahaja. Kaedah ini sesuai untuk *backup* seperti contohnya fasa pembangunan sistem. Apabila selesai satu modul pembangun sistem hanya perlu *backup* modul yang terlibat sahaja.
4. *Hot backup : Backup* dilakukan ke atas data ketika akses ke atas sistem/aplikasi masih aktif. Praktis ini membenarkan sistem/Aplikasi diakses oleh pengguna tetapi tidak digalakkan kerana integriti data tidak dapat dijamin.
5. *Cold backup :* Sesuai untuk dilaksanakan ke atas pangkalan data sahaja. *Backup* pangkalan data dilaksanakan semasa pangkalan data tidak aktif dan tiada akses oleh pengguna. Ini adalah praktis terbaik untuk memastikan integriti data adalah konsisten dan betul

**SOALAN/***QUESTION*:

1. Apakah maksud *backup*?
2. Apakah tujuan melaksanakan *backup*?
3. Terangkan kepentingan *backup* terhadap kod sumber.
4. Terangkan kepentingan *backup* terhadap pangkalan data.

**RUJUKAN /** *REFERENCES :*

1. Atlassian. (n.d.). What is version control: Atlassian Git Tutorial. Retrieved from <https://www.atlassian.com/git/tutorials/what-is-version-control>.
2. Back-up Activities. (n.d.). Retrieved from <http://ec-easycomputer.blogspot.com/2011/09/back-up-activities.html>.
3. (n.d.). Retrieved from [http://itinfo.uthm.edu.my/polisi/garis\_panduan\_*backup*.htm](http://itinfo.uthm.edu.my/polisi/garis_panduan_backup.htm).