|  |  |  |
| --- | --- | --- |
| 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** | **CU04/WA1 - DEPLOY SOURCE CODE TO DEVELOPMENT SERVER** | |
| **KOD DAN NAMA PROGRAM KV** | SISTEM PENGURUSAN PANGKALAN DATA DAN APLIKASI WEB | |
| **KOD DAN NAMA KURSUS KSKV** | K  KPD 3033 DEVELOPMENT ENVIRONMENT DEPLOYMENT | |
| **NO.DAN TAJUK KOMPETENSI** | K K1 DEPLOY SOURCE CODE TO DEVELOPMENT SERVER | |
| **NO. KOD NOSS** | IT-010-3:2016 - C04/P(1/12) | Muka Surat : 1 Drp : 5 |
| **NO. KOD KSKV** | KPD 3033 / P(1/12) |

**TAJUK:**

**PENYELENGGARAAN KOD SUMBER**

**TUJUAN:**

Kertas penerangan ini adalah bertujuan menerangkan mengenai :

1. Jelaskan penyelenggaraan Kod Sumber menggunakan:

* SCM
* Version Control

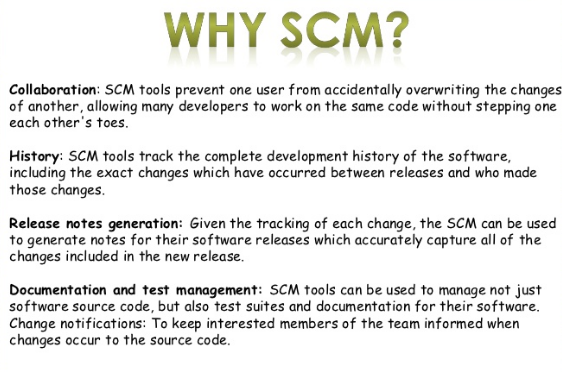
**PENERANGAN/***INFORMATION* **:**

**i. *MAINTENANCE OF SOURCE CODE***

1. ***Source Code Management* (SCM)**
   1. Juga dikenali sebagai ***source code maintenance*** / ***revision control*** / ***version control***.
   2. Merupakan pengurusan terhadap perubahan yang terjadi pada *source code* dan dokumen-dokumen lain semasa proses pembangunan perisian.
   3. 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.
   4. *Revision control* merupakan proses mengurus, merekod dan menjejaki perubahan yang telah dilakukan ke atas sistem / perisian berkenaan.
   5. **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**.
   6. Proses *revision control* dapat dipermudahkan dengan menggunakan perisian khas yang dikategorikan di bawah ***Version Control System***.
   7. Contoh perisian version control:
      * Git
      * Mercurial
      * Subversion
      * Source Code Control System (SCCS)
      * Vesta
      * AutodeskVault
      * Visual Studio Team Services (VSTS)
      * BitKeeper
   8. Terminologi dalam SCM:

|  |  |
| --- | --- |
| **Terminologi** | **Fungsi** |
| Source code repository | Merupakan pangkalan data / *server* yang menyimpan semua kod sumber. |
| Branching | Merupakan penduaan objek supaya sebarang perubahan yang dibuat akan berlaku kepada semua objek pada masa yang sama. |
| Revision | Merupakan proses mengurus, merekod dan menjejaki perubahan yang telah dilakukan ke atas sistem / perisian berkenaan.  Proses *revision control* dapat dipermudahkan dengan menggunakan perisian khas yang dikategorikan di bawah ***Version Control System***. |
| Access control | Pendekatan yang digunakan untuk keselamatan *source code.* |
| Distribution | Membolehkan banyak pembangun perisian untuk mengerjakan sesuatu projek yang sama tanpa memerlukan mereka berkongsi rangkaian yang sama |

* 1. Apakah itu *Source code* dan *Repository*?
* *Source code* 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
  1. Working Copy dalam SCM
* Fail setempat (*local files*) di mana pengaturcara membuat perubahan pada kod sumber.
* Perubahan ini tidak akan berubah sehingga proses *commit* dilaksanakan.



Rajah 1: Kebaikan kegunaan SCM

b) **Version Control**

1. 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.

Situasi1 1:

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.

Situasi 2:

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.

1. 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.
2. 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. Tindakan Asas dalam *versioning*

|  |  |
| --- | --- |
| **Tindakan** | **Penerangan** |
| **Add** | Masukkan fail ke repo untuk kali pertama, iaitu mengesannya dengan Version Control |
| **Revision** | Apa versi fail pada (v1, v2, v3, dll). |
| **Head** | Semakan terkini dalam repo. |
| **Check Out** | Muat turun fail dari repo. |
| **Check In** | Muat naik fail ke repositori (jika ia telah berubah). Fail mendapat nombor semakan baru, dan orang boleh "menyemak" yang terkini. |
| **Checkin Message** | Mesej ringkas yang menerangkan apa yang telah diubah |
| **Changelog / History** | Senarai perubahan yang dibuat pada fail sejak ia dicipta |
| **Update** | Segerakkan fail anda dengan yang terbaru dari repositori. Ini membolehkan anda merebut semakan terkini semua fail. |
| **Revert** | Buang “local changes” anda dan muat semula versi terkini dari repositori. |
| **Commit** | Dalam sistem version control, commit menambah perubahan terbaru pada kod sumber ke repositori, membuat perubahan ini menukar sebahagian daripada head revision repositori. Tidak seperti commit dalam pengurusan data, version control disimpan dalam repositori selama-lamanya. Oleh itu, apabila pengguna lain melakukan kemas kini atau checkout dari repositori, mereka akan menerima versi commit terbaru, kecuali mereka menyatakan mereka ingin mendapatkan versi sebelumnya. Version Control membolehkan kita beralih kembali ke versi sebelumnya dengan mudah. Dalam konteks ini, commit dalam version control dilindungi kerana ia mudah dilancarkan. |