|  |  |  |
| --- | --- | --- |
| **BAHAGIAN PENDIDIKAN DAN LATIHAN TEKNIK VOKASIONAL**  **KEMENTERIAN PENDIDIKAN MALAYSIA**  **ARAS 5 & 6, BLOK E14, KOMPLEKS E,**  **PUSAT PENTADBIRAN KERAJAAN PERSEKUTUAN**  **KERTAS PENERANGAN**  ***(INFORMATION SHEET)***  C:\Users\norsu\Desktop\jata negara.png | | |
| **KOD DAN NAMA PROGRAM NOSS** | IT-010-3:2016 PEMBANGUNAN APLIKASI | |
| **TAHAP NOSS** | 3 | |
| **KOD, NAMA CU DAN WA NOSS** | CU01/WA2 – SETUP LOCAL ENVIRONMENT | |
| **NAMA PROGRAM KV** | SISTEM PENGURUSAN PANGKALAN DATA DAN APLIKASI WEB | |
| **KOD DAN NAMA KURSUS KSKV** | KPD 1013 PRINCIPLES AND PRACTICES IN DATABASE | |
| **NO DAN TAJUK STANDARD KANDUNGAN KSKV** | K3 CREATE AND MANAGE TABLE USING DATA  DEFINITION LANGUAGE | |
| **NO. KOD NOSS** | IT-010-3:2016-C01 / P(11/26) | Muka Surat : 1 Drp : 9 |
| **NO. KOD KSKV** | KPD1043/KP(11/26) |

**TAJUK/TITLE :**

PROSEDUR PEMASANGAN PEMBANGUNAN PERISIAN

**TUJUAN/PURPOSE :**

Diakhir sesi pembelajaran, pelatih – pelatih mesti boleh:

1. Jelaskan mengenai pengurusan kod sumber(SCM)
2. Jelaskan mengenai Prosedur pembangunan pemasangan *kit*(*Development Kit Installation)*

**PENERANGAN**/*INFORMATION :*

**1. Pengenalan Source Code Management (SCM)**

* *Source code* (atau disebut juga *source*) adalah kumpulan pernyataan atau bahasa pemprograman komputer yang ditulis dan dapat di baca oleh manusia. *Source code* membolehkan pengaturcara untuk berkomunikasi dengan komputer menggunakan beberapa perintah (*command*).
* SCM juga dikenali sebagai ***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* sangat penting bagi 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***.

1.1 Contoh perisian SCM/version control:

* Source Code Control System (SCCS)
* Vesta
* AutodeskVault
* Visual Studio Team Services (VSTS)
* BitKeeper
* Git

1.2 Fungsi SCM

* Alat pengurusan kod sumber fungsi menggunakan prinsip asas memisahkan pengubahsuaian tempatan yang dibuat oleh pengguna, dalam direktori setempatnya, dan apa yang disebut repositori. Repositori mengandungi secara tersusun sejarah semua modifikasi yang dibuat oleh semua penyumbang projek.
* Perkembangan sesuatu projek perisian menggunakan alat pengurusan kod sumber diubahsuai berbanding pembangunan projek tempatan.
* Daripada menyimpan pengubahsuaian anda ke pemacu cakera setempat, anda sebagai penyumbang harus *submit* (*commit*) pengubahsuaian anda ke repositori untuk menjadikannya tersedia kepada pemaju lain.
* Sebaliknya, pemaju perlu memastikan mereka menggunakan versi terkini fail dengan mengambilnya dari repositori (*checkout*, *update*) sebelum membuat modifikasi mereka sendiri.
* Jika dua pengaturcara memodifikasi fail kod sumber yang sama pada masa yang sama, alat pengurusan kod sumber boleh melaporkan konflik semasa penyerahan dua modifikasi saingan atau secara automatik menyelesaikan konflik

1.3 Terdapat beberapa fungsi yang terdapat dalam perisian SCM antaranya:

* + *Source code Repository*
  + *Branching*
  + *Revision*
  + *Access control*
  + *Distribution*

a. *Source code repository*:

Repositori kod sumber adalah arkib fail dan kemudahan hosting web di mana sejumlah besar kod sumber, untuk perisian atau halaman web, disimpan, sama ada secara terbuka atau secara peribadi. Mereka sering digunakan oleh projek perisian sumber terbuka dan projek berbilang pemaju lain untuk mengendalikan pelbagai versi.

*“A source-code repository is a file archive and web hosting facility where a large amount of source code, for software or for web pages, is kept, either publicly or privately. They are often used by open-source software projects and other multi-developer projects to handle various versions.”*

*b. Branching*

Cabang (*branching*) adalah amalan mencipta salinan program atau objek dalam pembangunan untuk berfungsi dalam versi selari, mengekalkan asal dan bekerja di cawangan atau membuat perubahan yang berbeza untuk setiap. Setiap salinan dianggap sebagai cawangan; program asal yang mana cawangan diambil disebut sebagai batang, garis dasar(*baseline*), garis utama(*mainline*) atau *the master*.

Cabang digunakan dalam kawalan versi dan pengurusan perisian untuk mengekalkan kestabilan sementara perubahan terpencil dibuat untuk kod. Cabang memudahcarakan pembangunan pembetulan pepijat, penambahan keupayaan baru dan integrasi versi baru selepas mereka telah diuji secara berasingan.

*“Branching is the practice of creating copies of programs or objects in development to work in parallel versions, retaining the original and working on the branch or making different changes to each. Each copy is considered a branch; the original program from which the branch is taken is referred to as the trunk, the baseline, the mainline or the master.*

*Branching is used in version control and software management to maintain stability while isolated changes are made to code. Branching facilitates the development of bug fixes, the addition of new capabilities and the integration of new versions after they have been tested in isolation.”*

C. *Revision*

Dalam proses pembangunan perisian, kawalan semakan (*revision*), juga dikenali sebagai kawalan versi atau kawalan sumber, adalah pengurusan perubahan yang dibuat dari masa ke masa. Perubahan ini boleh digunakan untuk kod sumber, aset projek, atau sebarang maklumat lain yang masuk ke dalam produk siap. Ia membenarkan ramai orang untuk bekerja di bahagian yang sama dalam projek tanpa bimbang bahawa perubahan mereka akan menimpa kerja orang lain. Koleksi semakan dan metadata mereka dipanggil repositori atau repo. Repositori merupakan rekod kronologi langkah-demi-langkah setiap perubahan yang dibuat untuk membantu pengurus projek mengembalikan semua atau sebahagian projek ke keadaan sebelumnya jika perlu.

*“In the software development process, revision control, also known as version control or source control, is the management of changes made over time. These changes can be to source code, project assets, or any other information that goes into the finished product. It permits many people to work on the same parts of a project without worrying that their changes will overwrite the work of anyone else. The collection of revisions and their metadata is called a repository or repo. The repository represents a step-by-step chronological record of every change made to help project managers revert all or part of the project to a previous state if necessary.”*

*d. Access Control*

Kawalan akses adalah cara untuk mengehadkan akses kepada sistem atau sumber fizikal atau maya. Dalam pengkomputeran, kawalan akses adalah proses di mana pengguna diberi akses dan keistimewaan tertentu kepada sistem, sumber atau maklumat.

Dalam sistem kawalan akses, pengguna mesti mengemukakan kelayakan sebelum mereka boleh diberikan akses. Dalam sistem fizikal, kelayakan ini mungkin datang dalam pelbagai bentuk, tetapi kelayakan yang tidak dapat dipindahkan memberikan keselamatan yang paling.

*“Access control is a way of limiting access to a system or to physical or virtual resources. In computing, access control is a process by which users are granted access and certain privileges to systems, resources or information.*

*In access control systems, users must present credentials before they can be granted access. In physical systems, these credentials may come in many forms, but credentials that can't be transferred provide the most security.”*

*e. Distribution*

Dalam pembangunan perisian, kawalan versi diedarkan (*distributed version control*) adalah satu bentuk kawalan versi di mana kod lengkap - termasuk sejarah penuh - dicerminkan pada setiap komputer pemaju. Ini membolehkan cawangan dan penggabungan untuk diuruskan secara automatik, meningkatkan kelajuan kebanyakan operasi (kecuali untuk menolak dan menarik), meningkatkan keupayaan untuk bekerja di luar talian, dan tidak bergantung pada satu lokasi untuk backup.

*“In software development, distributed version control (also known as distributed revision control) is a form of version control where the complete codebase - including its full history - is mirrored on every developer's computer. This allows branching and merging to be managed automatically, increases speeds of most operations (except for pushing and pulling), improves the ability to work offline, and does not rely on a single location for backups.”*

**2. Prosedur pembangunan pemasangan *Kit* (*Development Kit Installation)***

2.1 Java Development Kit (JDK)

Kit Pembangunan Java (JDK) adalah persekitaran pembangunan perisian yang digunakan untuk membangunkan aplikasi java dan *applet*. Ia termasuk Persekitaran Runtime Jave (JRE), penterjemah/ *loader* (java), pengkompil (javac), sebuah archiver (jar), dokumentasi generator (javadoc) dan alat – alat lain yang diperlukan dalam pembangunan java.

a. Prosedur Instalasi JDK

i. Download JDK 8 (atau bisa disesuaikan dengan versi terbaru) dari lama oracle dan klik [link download](http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html) pilih sesuai bit OS yang digunakan.

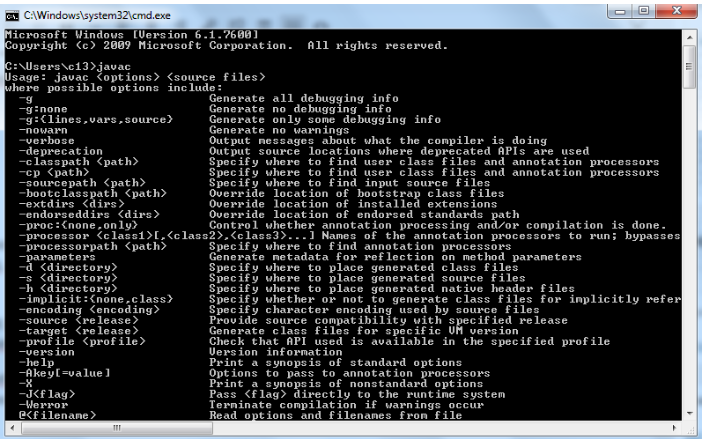
ii. Selesai proses instalasi, salin alamat tempat folder penginstallan jdk dengan cara :  **Computer > local Disk (c) > program files > java > folder jdk > bin > javac**

iii. Klik kanan pada **Computer > Properties > Advanced System Settings**

**iv.** Pilih **Environment Variable**

**v.** Maka akan terbuka window baru yaitu **Edit System Variable.** Arahkan kursor pada **box** bagian **Variable Value**dan arahkan kursor kebagian paling kanan. Tambahkan titik koma (;) dan **Paste** alamat lokasi folder JDK yang telah kita **Copy** sebelumnya dan klik **OK.**

**vi. Fungsi setting path environment ini adalah untuk membenarkan akses fail yang berada pada direktori yang telah ditetapkan menggunakan comman line.**

**vii. Untuk memastikan ketidak berhasilan path environment yang dilakukan, tulis perintah “javac” pada command line.**

Rajah 1: Hasil instalasi JDK

2.2 Software Development Kit (SDK)

i) *Software Development Kit (SDK)* adalah merujuk kepada koleksi perisian yang digunakan untuk membangun aplikasi bagi peranti atau sistem operasi tertentu. Contoh SDK termasuk Windows 10 SDK, Mac OS X SDK, dan SDK iPhone.

ii) SDK biasanya merangkumi persekitaran pembangunan bersepadu (IDE), yang berfungsi sebagai antara muka pusat pengaturcaraan. IDE merangkumi tetingkap pengaturcaraan untuk menulis kod sumber, *debugger* untuk menetapkan kesilapan program, dan editor visual, yang membolehkan pemaju membuat dan mengedit antara muka pengguna grafik (GUI) program. IDE juga termasuklah merangkumi pengkompil, yang digunakan untuk membuat aplikasi dari fail kod sumber.

iii) Kebanyakan SDK mengandungi kod sampel, yang menyediakan pemaju dengan program contoh dan perpustakaan (*library*). Sampel ini membantu pemaju belajar bagaimana membina program asas dengan SDK, yang membolehkan mereka akhirnya membuat aplikasi yang lebih kompleks. SDK juga menawarkan dokumentasi teknikal, termasuk tutorial dan Soalan Lazim. Sesetengah SDK juga mungkin termasuk grafik sampel, seperti butang dan ikon, yang boleh dimasukkan ke dalam aplikasi.

iv) Memandangkan kebanyakan syarikat ingin menggalakkan pemaju untuk membuat aplikasi untuk platform mereka, SDK biasanya disediakan secara percuma. Pemaju hanya boleh memuat turun SDK dari laman web syarikat dan mula pengaturcaraan dengan segera. Walau bagaimanapun, memandangkan setiap kit pembangunan perisian berbeza, boleh mengambil masa beberapa lama untuk pemaju untuk mempelajari cara menggunakan SDK baharu. Oleh itu, kebanyakan SDK moden termasuk dokumentasi yang luas dan mempunyai antara muka pengaturcaraan intuitif, yang membantu memberi insentif kepada pembangunan program.

**SOALAN**

1. Jelaskan definisi Source Code Management (SCM).
2. Jelaskan fungsi kit pembangunan perisian (Software Development Kit).

RUJUKAN/REFERENCE :

1. Abdullah Embong, 2000, sistem pangkalan data konsep asas, rekabentuk dan pelaksanaan, tradisi ilmu sdb.bhd
2. Cosmopoint Institut of information technology , 2001, data analysis & database design, Penerbitan cosmopoint
3. Halina Mohamed Dahlan, 2011, Sistem Pangkalan Data, Penerbitan UTM Press Sdn. Bhd, (ms 35 - 48)
4. Modi Lakulu, Aslina Saad, 2007, Sistem Pangkalan Data, Universiti Pendidikan Sultan Idris
5. Thomas Connolly, Database Systems 2005, Pearson Education Limited Sdn. Bhd, (ms 34)
6. <https://www.tutorialspoint.com/sql/sql-transactions.htm>
7. <http://ampps.com/mamp>
8. <http://www.wampserver.com/en/>
9. <https://www.techopedia.com/definition/5594/java-development-kit-jdk>