|  |  |  |
| --- | --- | --- |
| 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/WA2 – SETUP LOCAL ENVIRONMENT | |
| **KOD DAN NAMA PROGRAM KV** | TEKNOLOGI SISTEM PENGURUSAN PANGKALAN DATA DAN APLIKASI WEB | |
| **KOD DAN NAMA KURSUS KSKV** | KPD 2023 WEB PROGRAMMING | |
| **NO.DAN TAJUK STANDARD KANDUNGAN KSKV** | K2 SETUP LOCAL ENVIRONMENT | |
| **NO. KOD NOSS** | IT-010-3:2016-C02/ P(4/11) | Muka Surat : 1 Drp : 8 |
| **NO. KOD KSKV** | KPD 2023 / P(4/11) |

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

**SETUP LOCAL ENVIRONMENT**

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

Kertas penerangan ini adalah bertujuan menerangkan mengenai :

* Definisi persekitaran pembangunan (*development environment*)
* Pengenalan *local server*
* Pengenalan kepada procedur pemasangan *server*
* Pengenalan *Source Code Management (SCM)*

**2.1 PENGENALAN KEPADA PERSEKITARAN PEMBANGUNAN (*DEVELOPMENT ENVIRONMENT)***

Persekitaran pembangunan (*development environment*) adalah satu set perisian yang membolehkan anda untuk menulis program bagi bahasa atau platform tertentu. Perisian ini adalah seperti *editor teks*, shell, dan pengkompil / jurubahasa. *Editor teks* merupakan satu program di mana membolehkan anda menaip kod anda. Contoh *editor teks* termasuk *Sublime Text, Notepad ++,* dan *Eclipse, NetBeans , Aptana*. Pada sistem pengoperasian *Windows, default shell* yang digunakan adalah *Command Prompt.* Pada pengoperasian *OS X, shell* yang digunakan adalah *Terminal.* Perkara yang perlu dititikberatkan dan penting dalam persekitaran pembangunan (*development environment*) adalah pengkompil atau pentafsir. Pada asasnya, pengkompil / jurubahasa adalah perisian yang akan mengambil kod yang anda menulis dan menukarkannya ke dalam bentuk Bahasa yang komputer boleh memahami.

**2.2 PENGENALAN KEPADA *LOCAL SERVER***

*"Localhost"* merujuk kepada *local computer* bahawa program sedang dijalankan. Sebagai contoh, jika anda menjalankan pelayar web (*web browser*) pada komputer anda, komputer anda dianggap sebagai *"localhost"*. Walaupun ini tidak perlu ditentukan semasa menggunakan komputer, *localhost* perlu ditakrifkan apabila menjalankan program daripada pelbagai komputer. Sebagai contoh, pentadbir rangkaian mungkin menggunakan *local machine* untuk memulakan pelayan Web pada satu sistem dan menggunakan program akses jauh. Program-program ini akan dijalankan dari komputer selain daripada*localhost*.

Dalam contoh di atas, kedua-dua *non-local computers* mesti ditakrifkan oleh alamat IP mereka. *Local machine* ditakrifkan sebagai *"localhost"*, yang memberikannya alamat *IP 127.0.0.1*. Ini dianggap sebagai *"loopback"* alamat kerana maklumat yang dihantar kepadanya dikembalikan ke *local machine*. *Localhost* sering digunakan dalam bahasa skrip web seperti PHP dan ASP ketika menentukan apa yang pelayan harus menjalankan kod dari atau di mana pangkalan data berada.

Pembangunan web, *localhost* adalah persekitaran pelayan tempatan di mana kita boleh menguji dan menggunakan skrip pelayan pada komputer anda. Anda boleh *set-up* persekitaran *localhost* dengan aplikasi pelayan Apache, MySQL dan PHP.

**2.2.1 SENARAI *PHP LOCALHOST SERVERS (LOCAL SERVERS)***

|  |  |
| --- | --- |
| **AMPPS Stack** |  |
| **XAMPP** |  |
| **WinNMP** |  |
| **VertrigoServ** |  |
| **EasyPHP DevServer** |  |
| **WampServer** |  |
| **fenixwerserver** |  |
| **Laragon** |  |
| **Bitnami** |  |
| **Devilbox** |  |
| **MAMP** |  |

**2.3 PENGENALAN KEPADA PROCEDUR PEMASANGAN *SERVER***

Dokumen ini menerangkan prosedur langkah demi langkah sekiranya ingin melakukan sesuatu pemasangan Perisian *Server* pada perkakasan pelanggan. Dokumen itu juga menggambarkan keadaan yang mesti dipenuhi oleh pelanggan sebelum boleh melakukan pemasangan.

Pra syarat yang perlu dipatuhi oleh pelanggan.

1. Keperluan perkakasan mesti memenuhi keperluan yang minima.
2. Keperluan perisian
3. Keperluan infrastruktur
4. Maklumat yang diperlukan
5. Rangkaian Internet

**2.4 PENGENALAN KEPADA *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 *programmer* untuk berkomunikasi dengan komputer menggunakan beberapa perintah (*command*). Terdapat beberapa elemen dalam *Source Code Management (SCM)* antaranya ialah *Source code repository*, *Branching*, *Revision*, *Access control* dan *Distribution*.

**2.3.1 *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.”*

**2.3.2 *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.”*

**2.3.3 *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.”*

**2.3.4 *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.”*

**2.3.5 *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.”*

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

1. Jelaskan definisi persekitaran pembangunan.
2. Jelaskan definisi *local server.*
3. Jelaskan definisi *Source Code Management* (SCM).

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

1. https://redmonk.com/sogrady/2007/06/26/dscm/ - *Distributed Source Code Management*
2. https://www.youtube.com/watch?v=tpyDK-Y44eI – pautan untuk *How to setup a development environment (youtube)*
3. https://www.youtube.com/watch?v=sWqaN1s14jk - pautan untuk *How to setup a development environment 2 (youtube)*