S. Katheeskumar.
BAT/IT/2018//FT/030

Software Configuration Management (FT)
Continues Assessment – 02

1. What do you mean by Version Control System and list four (04) advantage of having VCS than manual system?

Version control allows you to manage changes to files over time. You can use version control to version code, binary files, and digital assets. This includes version control software, version control systems, or version control tools. Version control is a component of software configuration management. It's sometimes referred to as VCS programming.

Advantages

- It supports for multiple users
- Descriptions keeping Our changes to make it easier to find the version need
- Tools to compare and merge different versions
- Allows to the permanent changes from trial work using branches

2. Define the term “Configuration Item” and give four (04) example for that.

A configuration item can not only be at the most atomic level, but also can consist of a more complex assembly of other configuration items. In other words, a configuration item can be a primitive component or an aggregate of other configuration items. In fact, the level at which a configuration item is considered as primitive or aggregate is often decided by the system in which it is created, maintained and managed.

3. Compare and Contrast three classes of tools in SCM.

- Individual support tool :- Smaller companies or development groups
  Small number of releases
  No variants

- Project-related support tool :- Medium to larger companies
  Parallel development
  Need for handling variants

- Full, company-wide process-support tool:- Larger companies
  Parallel development
  Need for handling variants

4. List out all the main activities in SCM and explain each activities.

Software Configuration Management(SCM) is a process to systematically manage, organize, and control the changes in the documents, codes, and other entities during the Software Development Life Cycle. The primary goal is to increase productivity with minimal mistakes. SCM is part of cross-disciplinary field of configuration management and it can accurately determine who made which revision.

5. Compare and contrast “Local only, Centralized and De-centralized” System in Source Control.

- Work slow in the centralized, but faster work in the De-centralized
- Centralized Focuses on synchronizing, tracking and backing up files, but Centralized focuses on sharing changes.
- Centralized is best for a small sized organization, but the large sized organization should practice decentralize.
- Centralized no local repository, but local repository in the De-centralized source control.
- A failure in the central server terminates all the version in the Centralized, but A failure in the main server does not affect the development in the Centralized

6. Write GIT command for following activities.

a. Create a folder called “Project” in your present working directory and move inside the directory

> **git init**

b. Let’s say there is a repository in the GitHub with following address “https://github.com/jasontylor/simple-demo.git”. Checkout this repository into your “Project” folder and move inside the “simple-demo” folder

> **git remote add origin https://github.com/jasontylor/simple-demo.git**

c. There is a file inside the simple-demo called “README.md”, make some changes to this file and commit it

> **git add README.md**
> **git commit -m “Added Readme file”**

d. Push the committed file to the GitHub account.

> **git push origin master**

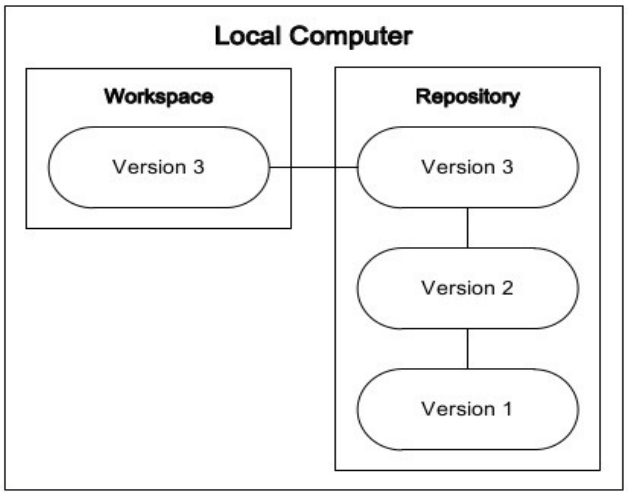
7. List down four (04) functions of Version Control System.

1.Clean up local work space.
2.Check out the project code base from the VCS into the local workspace.
3.Record changes to the project code base provided by VCS
4.Run the build to transform project code base to working software.
5.Record results and logs produced by the build for future reference

8. Briefly Explain the following terms,

a. Version

=> Version is a code or specific program updated level or bugs cleared level. it’s indicated by numbers. Ex: 1.1 , 1.2 ,1.3



b. Revision

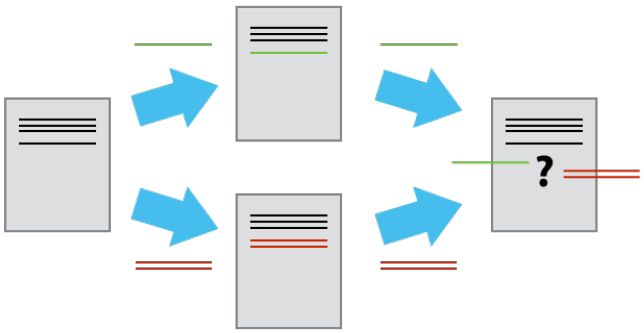
=> *Version control* is a *system* that records changes to a file or set of files over time had a simple database that kept all the changes to files under *revision control*.

c. Release

=> Throughout the software development process, your code will go through cycles of development, testing, and fixing. Depending on your team, technology, platform, and other variables, you'll likely have several environments that you use to manage your application. In order to effectively test and release software, it's imperative to have a simple and reliable release process for getting your code onto your various environments.

d. Conflict

=> conflict comes modified a file at the same time two or more developers.



e. Check in

=> Save changes back to the repository