**VERSION CONTROLS SYSTEMS**

**PROJECT**

What is Version Control System?

A version control system is also known as revision control, source control, or source code management is a class of systems responsible for managing changes to computer programs, documents, large websites, or other collections of information.

Project Topic: Write in detail about a Version Control system of your choice ensure to include its functionalities, advantages, and disadvantages of the software

Kindly attach below the version control systems you will be discussing

1. Edwin VCS GitHub

2. Obafemi VCS

3. Ali VCS Mercurial.

4. Abiodun VCS

**Name: Ali Abdul**

**Topic: Mercurial Version Control System**

**YDEV ACADEMY: Cloud and Devops.**

**Definition:**

**Mercurial is a free, distributed source control management tool. It offers you the power to efficiently handle projects of any size while using an intuitive interface. It is easy to use and hard to break, making it ideal for anyone working with versioned files.**

**Distributed architecture:**

Traditional version control systems such as Subversion are typical client-server architectures with a central server to store the revisions of a project. In contrast, Mercurial is truly distributed, giving each developer a local copy of the entire development history. This way it works independent of network access or a central server. Committing, branching and merging are fast and cheap.

## Fast:

Mercurial's implementation and data structures are designed to be fast. You can generate diffs between revisions, or jump back in time within seconds. Therefore Mercurial is perfectly suitable for large projects such as nginx ([hg](http://hg.nginx.org/nginx/)) or Net Beans ([hg](http://hg.netbeans.org/)).

## Platform independent:

Mercurial was written with platform independence in mind. Therefore most of Mercurial is written in Python, with a small part in portable C for performance reasons. As a result, binary releases are available on all major platforms.

## Extensible:

The functionality of Mercurial can be increased with extensions, either by activating the official ones which are shipped with Mercurial or downloading some [from the wiki](https://www.mercurial-scm.org/wiki/UsingExtensions) or by [writing your own](https://www.mercurial-scm.org/wiki/WritingExtensions). Extensions are written in Python and can change the workings of the basic commands, add new commands and access all the core functions of Mercurial.

## Easy to use:

Mercurial sports a consistent command set in which most subversion users feel right at home. Potentially dangerous actions are available via extensions you need to enable, so the basic interface is easy to use, easy to learn and hard to break. The [Quick Start](https://www.mercurial-scm.org/quickstart) should get you going in a just few minutes.