SOURCE CODE MANAGEMENT

TASK (13/11/2019)

**Content:**

* + What is Source Code Management ……………....2

# Features of Version Control ….……….…………..2

# Types of Version Control …………………………....2

## Centralised Version Control…………………...3

## Distributed Version Control…………………….3

## Popular Tools used in Version Control ………....... 3

What is Source Code Management?

* Source code management (SCM) is used to track modifications to a source code repository. SCM tracks a running history of changes to a code base and helps resolve conflicts when merging updates from multiple contributors.
* We can call this source code management as version control or revision control.

Features of Version Control:

* Authenticated access for commits
* Revision history on files
* Atomic commits of multiple files
* Versioning/Tagging

Types of Version Control:

1. Centralised Version Control
2. Distributed Version Control

1.Centralised Version Control:

* Centralized Version Control works in a client and server relationship. The repository is located in one place and allows access to multiple clients.
* It’s very similar to FTP where you have FTP clients which connect to FTP server. Here all the user changes and commits must pass through the central server.

The benefits of centralized version control are:

* It’s easy to understand.
* There are more GUI and IDE clients.
* You have more control over the users and access.

The drawbacks of centralized version control are:

* It is dependent on the access to the server.
* It can be slower because every command from the client has to pass the server.
* Branching and merging strategies are difficult to use.

2.Distributed Version Control:

* In Distributed Version Control, each user has their own copy of the entire repository as well as the files and history.
* By using that copy we can change data and again recommit to remote repository.

The benefits of distributed version control are:

* More powerful and easy change tracking.
* No need of a centralized server. Most of the functionalities work in offline mode also apart from sharing the repositories.
* Branching and Merging strategies are easier and more reliable.
* It’s faster than the other one.

The drawbacks of distributed version control are:

* It is harder to understand.

Popular Tools used in Version Control:

* Git
* VSTS (Visual Studio Team Services)
* Perforce Helix Core
* Mercurial
* Monotone
* Bazaar
* CVS
* SVN