Introduction to Git

Scenario Without VCS (Version Control System)

- Once the code is saved, all the changes made in the file are permanent and cannot be reverted back
- No record of what was done by whom

Dave is working on a new feature

The client didn’t like the new feature

Dave delete the code and continue with the development

The client change his mind and want that feature back

Same Scenario With VCS (Version Control System)

Dave is working on a new feature

The client didn’t like the new feature

Dave delete the code and continue with the development and client change his mind and want that feature back

Dave is happy because of VCS

What Is VCS (Version Control System)?

- **VCS** is a system that documents changes made to a file or a set of files. It allows multiple users to manage multiple revisions of the same unit of information.
- A version control system records changes carried out to a file, or a set of files over time, and helps you to recall specific versions of the code later on when you need it. If you’re a developer or a web designer and want to keep track of each and every version of your code developed for a particular feature, a Version Control System “VCS” allows you to:
    - Revert the code files back to their previous state
    - Recall and revert the entire project back to its previous state
    - Compare code changes over specific durations of time
    - Find who last modified a piece of code that might be causing an issue or a problem
    - Who introduced a particular issue and when

What is Git?

Git is the most popular and widely used version control system today. Originally developed as an open-source project in 2005 by the creator of the Linux operating system, it is a matured and actively maintained environment. A huge number of developers rely upon Git version controlling to develop several types of commercial and non-commercial projects.

Git Workflow

Centralized VCS and Distributed VCS

Centralized vs Distributed Version Control: Which One Should We Choose

- Distributed VCS:
    - Git,
    - Bitbucket
    - etc.
- Centralized VCS:
    - CVS,
    - SVN etc.

Git Workflow with branches

Merge Conflicts

Dave and Sam start a new feature on **v1.2**

Dave is working on **v1.2** locally

Sam is also working on **v1.2** locally

Dave push the code into the remote repository and Sam is having a merge conflict with **Dave** code

Why do merge conflicts occur?

- Merge conflicts arise when two files with the same content are modified and merged.
- Merge conflicts can occur on merging branches

How to resolve merge conflicts?

- Merge conflicts are resolved manually by users
- We can use VS code also to compare and resolve merge conflicts