Assignment 1

GIT & GITHUB

Sasanka Sekhar Kundu

Date of Submission: 11th November 2025

Roll No: 150096725118

Git and GitHub Assignment:

Q1. What is a Version Control System (VCS)?

Answer : A **Version Control System (VCS)** is a software tool that helps manage changes to source code over time, allowing multiple developers to work together, track history, revert to earlier versions, and collaborate efficiently.

Q2. Why is VCS needed in software development?

Answer : VCS needed in software development because of the following reason :

- To maintain a history of changes and track who made them.
- To enable collaboration between multiple developers without overwriting each other's work.
- To roll back to previous stable versions when bugs occur.
- To manage different features or bug fixes using branches.

Q3. List any two examples each of Centralized and Distributed VCS .

Answer : The two example of each -

- **Centralized VCS (CVCS):** Subversion (SVN), Concurrent Versions System (CVS).
- **Distributed VCS (DVCS):** Git, Mercurial.

Q4. Explain the major difference between Centralized and Distributed systems.

Answer:

Centralized VCS (CVCS)	Distributed VCS (DVCS)
A single **central server** holds the entire codebase.	Every developer has a **full copy** of the repository (including history).
Developers must be **online** and connected to the central server to commit or update.	Developers can **work offline**, commit locally, and sync later.
If the **central server fails**, collaboration stops until it is restored.	No single point of failure; any developer's copy can restore the repository.
Examples: **SVN, CVS**	Examples: **Git, Mercurial**

Q5. Write two advantages and disadvantages of a Distributed VCS.

Answer:

Advantages :

- Developers can work offline since they have the full history locally.
- No single point of failure; if the server goes down, any developer's copy can restore it.

Disadvantages :

- Slightly complex to learn and manage compared to CVCS.
- Requires more disk space since each developer has a full copy of the repository.

Q6. What happens if the central server fails in a CVCS?

Answer : If the central server fails, developers cannot commit changes, collaborate, or access the latest code until the server is restored, as the server is the single source of truth.

Q7. Which type of VCS do you prefer and why?

Answer: I prefer a **Distributed VCS (e.g., Git)** because it is faster, supports offline development, offers better branching/merging, and ensures project continuity even if the central server fails.