**Git and GitHub**

**1. Version Control System (VCS)**

* **Definition**: A system that tracks changes to files (e.g., source code, documents, images) over time.
* **Purpose**:
  + Recover older versions of files if mistakes are made.
  + Facilitate collaboration among team members.
* **Example**: Managing a shared shopping list with roommates. Version control ensures everyone’s contributions are tracked and merged seamlessly.

**2. What is Git?**

* **Definition**: A **distributed version control system (DVCS)** that allows users to track changes to files and collaborate on projects.
* **Key Features**:
  + **Free and Open Source**: Distributed under the GNU General Public License.
  + **Distributed**: Every user has a full copy of the project on their computer.
  + **Branching Strategies**: Supports workflows like **feature branching** to organize development.
* **Use Cases**:
  + Tracking changes in code, documents, images, and other file types.
  + Enabling collaboration among developers and data scientists.

**3. What is GitHub?**

* **Definition**: A **web-hosted service** for Git repositories, providing a user-friendly interface for collaboration.
* **Key Features**:
  + Hosts Git repositories remotely.
  + Provides tools for collaboration, such as pull requests and issue tracking.
* **Alternatives**: GitLab, Bitbucket, Beanstalk.

**4. Basic Terms in Git and GitHub**

* **SSH (Secure Shell)**: A protocol for secure remote login between computers.
* **Repository (Repo)**: A folder or directory that contains your project files and is set up for version control.
* **Fork**: A copy of a repository, allowing you to experiment without affecting the original project.
* **Pull Request**: A request to merge changes from one branch into another, often used for code reviews.
* **Working Directory**: The local files and subdirectories on your computer associated with a Git repository.
* **Commit**: A snapshot of the project’s state at a specific point in time, along with a description of changes.
* **Branch**: A separate line of development, allowing you to work on features or fixes independently.
* **Merge**: Combining changes from one branch into another (e.g., merging a feature branch into the main branch).
* **Clone**: Creating a local copy of a remote Git repository on your computer.

**5. Key Takeaways**

* **Git** is a distributed version control system used to track changes and collaborate on projects.
* **GitHub** is a popular web-hosted service for Git repositories, offering tools for collaboration.
* **Version Control Systems** like Git help you:
  + Revert to previous versions of files.
  + Review the history of changes.
  + Collaborate efficiently with others.
* **Branching Strategies** (e.g., feature branching) help organize and manage development.
* **Basic Terms** include repository, fork, pull request, commit, branch, merge, and clone.