Course Schedule for Value Added Certificate Course on End-to-End DevOps Automation Using Git, Docker, and Jenkins CI/CD Pipelines

Module 1 – Version Control System and Basic Git Commands (8 sessions)

Sess.	Date	Module	Topics to Cover
1	10-11-2025	M1	**Introduction to Version Control**: Why VCS is needed; problems without VCS; basic terminology (repository, commit, revision). **Centralized vs Distributed VCS**: Concepts, examples (SVN vs Git); advantages of distributed VCS. **Overview of Git**: History, where Git is used, basic workflow at a high level.
2	11-11-2025	M1	**Installing Git** (Windows/Linux): Download, install, verify with git --version. **Git life cycle overview**: untracked → staged → committed. **Local repository concept**: working directory vs repository. Demo simple project folder.
3	12-11-2025	M1	**Configuring Git identity**: git config --global user.name, user.email; checking config. **Initializing repository**: git init, role of .git folder. **Untracked vs tracked files**: using git status to see them; first-time tracking.
4	13-11-2025	M1	**Staging area**: meaning of index; git add variations (git add file, git add .). **First commit**: git commit -m "message"; what a commit stores. **File status transitions**: unmodified/modified/staged; practice with edits and multiple commits.
5	14-11-2025	M1	**Viewing commits**: git log, git log --oneline, commit hash, HEAD. **Git folder structure**: basic tour of .git (HEAD, refs, objects at a conceptual level). **Checking differences**: intro to git diff (working tree vs staged vs committed).
6	15-11-2025	M1	**Restoring deleted/modified files (simple)**: git restore <file>, git restore --staged <file>. **Undo last changes before commit**: using git checkout -- <file> (legacy), git restore. Hands-on mini-exercise of intentionally deleting and restoring.
7	17-11-2025	M1	**Git reset options (intro)**: concept of moving HEAD. Difference between --soft, --mixed, --hard at a conceptual level (no risky demos yet). Practice git reset --soft HEAD~1, git reset --mixed.
8	18-11-2025	M1	**Cloning repositories**: git clone <url>; local vs remote repo. **Remote basics**: origin, default branch. **Pull and push operations**: git pull, git push; basic sequence of clone → modify → commit → push. Quick recap of Module 1.

Module 2 – Advanced Git Commands and Collaboration (8 sessions)

Sess.	Date	Module	Topics to Cover
9	19-11-2025	M2	**Reference logs (reflog)**: what reflog stores; difference between git log and git reflog; recovering lost commits scenario. Hands-on: create a few commits and view git reflog.
10	20-11-2025	M2	**Tagging (lightweight vs annotated)**: git tag v1.0, git tag -a v1.0 -m. Listing tags, showing tagged commit. **Using tags for releases**: pushing tags with git push origin --tags, deleting tags locally and remotely.
11	21-11-2025	M2	**Branching concepts**: why branches are needed; feature branch workflow. Commands: git branch, git switch, git checkout -b. Understanding HEAD and active branch. Naming conventions for branches.
12	22-11-2025	M2	**Merging branches**: Fast-forward vs merge commit; git merge. Conflict scenarios: what a conflict looks like; simple conflict resolution using editor. **Merge commit messages** and verifying history with git log --graph.
13	24-11-2025	M2	**Reverting merges and commits**: git revert <commit>, special case of reverting merge commit with -m option (conceptual). **Deleting branches**: git branch -d vs -D, when each is appropriate. Safety tips.
14	25-11-2025	M2	**Stash operations**: use cases for git stash. Commands: git stash, git stash list, git stash show, git stash apply vs pop, dropping stash. Scenario: switching branches without committing by using stash.
15	26-11-2025	M2	**Archiving repositories**: git archive concept; creating zip/tar archives for distribution. **Hosting repositories on GitHub**: creating GitHub account, creating new repo, connecting local repo (git remote add origin). Brief on README, LICENSE.
16	27-11-2025	M2	**Managing remote repositories**: git remote -v, adding multiple remotes, changing URLs. **Synchronizing local and remote**: pull before push, git fetch vs git pull. **Access control for collaborators**: adding collaborators on GitHub, roles, basic workflow of fork → clone → PR.

Module 3 – Introduction to Docker (10 sessions)

Sess.	Date	Module	Topics to Cover
17	28-11-2025	M3	**Containerization concepts**: difference between VM and container; isolation and images; benefits of containerization in development and deployment. **Use cases** in real projects.
18	29-11-2025	M3	**Docker architecture**: Docker daemon, client, registry, images, containers. **Key terminologies**: image, container, registry, Docker Hub, Dockerfile, volume, network. Simple architecture diagram explanation.
19	01-12-2025	M3	**Installing Docker** (Desktop / Engine overview): prerequisites and basic configuration. **Verifying installation**: docker version, docker info. Understanding root vs non-root usage (conceptual).
20	02-12-2025	M3	**Basic Docker commands (containers)**: docker run, docker ps, docker ps -a, docker stop, docker start, docker restart, docker rm. Running simple container (hello-world, nginx, or alpine).
21	03-12-2025	M3	**Managing images**: docker images, docker pull, docker rmi. Image naming convention (repository:tag). **Docker Hub basics**: searching images (docker search), official vs community images.
22	04-12-2025	M3	**Networking and ports**: container networking basics; -p host:container port mapping. Example: running a web server container and accessing it via browser (localhost:port). Concept of exposing ports.
23	05-12-2025	M3	**Docker volumes and storage**: why we need volumes; bind mounts vs named volumes. Commands: docker volume create, docker volume ls, docker run -v. Demonstrate data persistence when container is removed.
24	06-12-2025	M3	**Linking containers / container communication** (bridge network). **Building Dockerfiles**: structure of a Dockerfile (FROM, WORKDIR, COPY, RUN, CMD, EXPOSE). Build an image using docker build -t name . and run it. Intro to optimizing images (using smaller base images).
25	08-12-2025	M3	**Deploying simple web server**: create a small web app (e.g., static HTML or simple PHP/Node app), write Dockerfile, build and run with correct port mapping. Test in browser. Discussion of environment variables (-e).
26	09-12-2025	M3	**Docker Compose (intro)**: why Compose is needed (multi-container apps). Structure of docker-compose.yml (services, image/build, ports, volumes). Run docker compose up/down on a simple 2-service example (web + database).

Module 4 – Jenkins Pipeline and CI/CD (4 sessions)

Sess.	Date	Module	Topics to Cover
27	10-12-2025	M4	**Overview of Jenkins and CI/CD**: what CI/CD is, why it matters. Jenkins architecture at high level, use cases. **Installing Jenkins** (conceptual + screenshots/steps). Accessing Jenkins web UI for the first time.
28	11-12-2025	M4	**Configuring Jenkins with Git**: global tools configuration (Git, JDK). Creating first freestyle job pulling from Git repository. Build triggers: manual vs poll SCM. Running first build and viewing console output.
29	12-12-2025	M4	**Jenkins distributed architecture**: master / controller and agents. Types of agents (SSH, JNLP, Docker-based). Adding a simple agent (conceptual or demo on localhost). When and why to scale using agents.
30	13-12-2025	M4	**Jenkins pipelines (intro)**: scripted vs declarative pipeline. Creating a simple declarative Jenkinsfile (checkout, build, test, archive). Storing Jenkinsfile in Git, configuring pipeline job to use it. **Basic notifications** (e.g., email/post-build actions conceptually).