

A4 Slide — Git Workflow Practice Assignment (4 Members)

Git Workflow Practice — Team Assignment (4 Members)

Goal: Practice real Git collaboration using branches, PRs, and sequential merging.

Team Roles & Files:

- **Member 1:** header.txt (Global Header)
- **Member 2:** navbar.txt (Navigation Bar)
- **Member 3:** video_player.txt (Video Player)
- **Member 4:** config.txt (Configuration)

Workflow Summary:

1. **Lead creates repo** → Adds README → Pushes → Invites collaborators.
2. **Each member:**
 3. Clone → Create feature branch → Add assigned file → Commit → Push.
4. **Sequential PR merging:**
 5. Member 1 PR → Merge.
 6. Member 2-4: Pull latest main → Merge main → Create PR → Merge.

Final Output: Main branch contains:

```
README.md, header.txt, navbar.txt, video_player.txt, config.txt
```

Assignment Document — Full Instructions

Simplified Git Workflow Practice Assignment (4 Members)

Objective:

To simulate professional Git collaboration using feature branching, merging, and PR workflows using simple text files.

PART 1: Repository Setup (Team Lead)

1. Create a new repository (e.g., `git-workflow-practice`).
2. Clone it locally.
3. Add a `README.md` containing project title and description.
4. Commit and push it:

```
git add README.md  
git commit -m "Initial setup with README"  
git push origin main
```

5. Add all 3 teammates as **collaborators**.

PART 2: Feature Creation (Each Member)

Each member creates a dedicated file:

Member	Component	File	Branch
1	Header Component	header.txt	feat/create-header
2	Navigation Bar	navbar.txt	feat/create-navbar
3	Video Player	video_player.txt	feat/create-video
4	Config Settings	config.txt	feat/create-config

Steps for all members:

1. Clone the repo:

```
git clone <repo-url>
```

2. Create & switch to your feature branch:

```
git checkout -b <your-branch-name>
```

3. Create your file with assigned text.

4. Commit your changes:

```
git add <file>.txt  
git commit -m "feat: added <file> component file"
```

5. Push the branch:

```
git push origin <branch>
```

PART 3: Pull Requests & Merging Workflow

Member 1:

- Create PR → merge to main.

Members 2, 3, 4:

Before creating PR:

```
git checkout main  
git pull origin main  
  
git checkout <your-branch>  
git merge main
```

Then: - Create PR → merge.

EXPECTED FINAL RESULT

Main branch must contain: - `README.md` - `header.txt` - `navbar.txt` - `video_player.txt` - `config.txt`

This confirms successful Git branching, collaboration, syncing, and merging workflow.
