

Department	Semester	Roll no	Name of Student	Assignment No
INFT	6	22101B0037	Aneesh Angane	2

Git Workflow for TechCo's Team Project

Branching Strategy

- **Main branches**
 - main: The stable production-ready branch.
 - develop: The integration branch where all features are merged before release.
- **Feature branches** (for new development)
 - feature/user-authentication → Adds login and signup functionality
 - feature/dashboard-ui → Implements the main dashboard interface
 - feature/payment-gateway → Integrates Stripe for payments
 - feature/notifications → Adds email and in-app notifications
 - feature/chat-system → Implements real-time chat between users
- **Bugfix branches** (for fixing issues)
 - bugfix/fix-login-issue → Resolves incorrect password handling
 - bugfix/fix-dashboard-styling → Fixes alignment issues in the dashboard
 - bugfix/fix-payment-error → Solves an issue where transactions fail intermittently

Workflow Steps

1. **Project Initialization**
 - git init (if starting fresh)
 - git clone <https://github.com/VITEDUIN/Assignment2---Git-Collaboration.git>
 - git checkout -b develop (create and switch to develop branch)
2. **Setting Up the Team**
 - Assign roles (e.g., one person manages releases, others work on features/bugs)
 - Developers clone the repository and set up their branches
3. **Developing Features**
 - Each developer creates a feature branch from develop
 - Example:


```
git checkout -b feature/user-authentication develop
```
 - After coding, commit changes:


```
git add .
git commit -m "Added login & signup functionality"
git push origin feature/user-authentication
```

4. Code Review & Pull Requests

- Open a pull request (PR) to merge into develop
- At least one team member must review and approve before merging

5. Bug Fixing

- Developers working on bug fixes follow a similar process but use `bugfix/` branches
- Example:

```
git checkout -b bugfix/fix-login-issue develop
```

6. Merging & Conflict Resolution

- Before merging, ensure the latest changes are pulled:

```
git checkout develop
git pull origin develop
git merge feature/user-authentication
```

- If conflicts arise, resolve them manually and commit

7. Release Process

- When ready for deployment, create a release branch:

```
git checkout -b release/v1.0 develop
```

- Perform final testing, then merge into main

8. Deployment & Hotfixes

- Once deployed, hotfixes go into `hotfix/` branches (e.g., `hotfix/fix-critical-bug`)
- Merge hotfixes into both `main` and `develop` to keep versions aligned