

Code commit process and guidelines

In this module I understood about the Code commit process and guidelines.

While committing code every developer should follow following practices:

Basic Rules:

1. **Commit related changes:**
Each commit should contain changes related to a specific task or issue.
2. **Commit often:**
Frequent commits help in tracking progress and make it easier to revert changes if necessary.
3. **Avoid half-dozen commits:**
Instead of making numerous small commits, aim for logical, cohesive commits that represent complete units of work.
4. **Test code before committing:**
Always test your code locally before committing to ensure it functions as expected.
5. **Write clear commit messages:**
Provide descriptive commit messages that explain the purpose and scope of the changes.

Branches:

1. **Master branch** (Production):
This branch reflects the code currently in production. It's important to never commit directly to this branch.
2. **Development branch** (Dev):
This branch holds the ongoing development code. Similar to the master branch, direct commits here are not allowed.
3. **Working/Feature/Issue branches**

These branches are typically named after a Jira or teamwork ticket ID, or the feature being worked on.

Branch Workflow:

1. **Commit and push in feature branch:**

Always commit and push your code in your feature branch before leaving the office. Ensure that your commit messages clearly describe the task you've accomplished.

2. **Raise pull request:**

Submit a pull request in GitHub/Bitbucket/GitLab against the development branch (Dev).

3. **Peer review:**

Your code will be reviewed by team members or leads.

4. **Pull latest changes:**

Before starting work each day, pull all the latest changes from the development branch (Dev) into your working branch. This ensures you're working with the most up-to-date codebase.

5. **Resolve merge conflicts:**

If faced with merge conflicts during the merge process, carefully resolve them. Ensure that you don't overwrite code that you haven't touched.

Release branch: If following a sprint cycle, a release branch will be created at the end of each sprint to prepare for deployment.