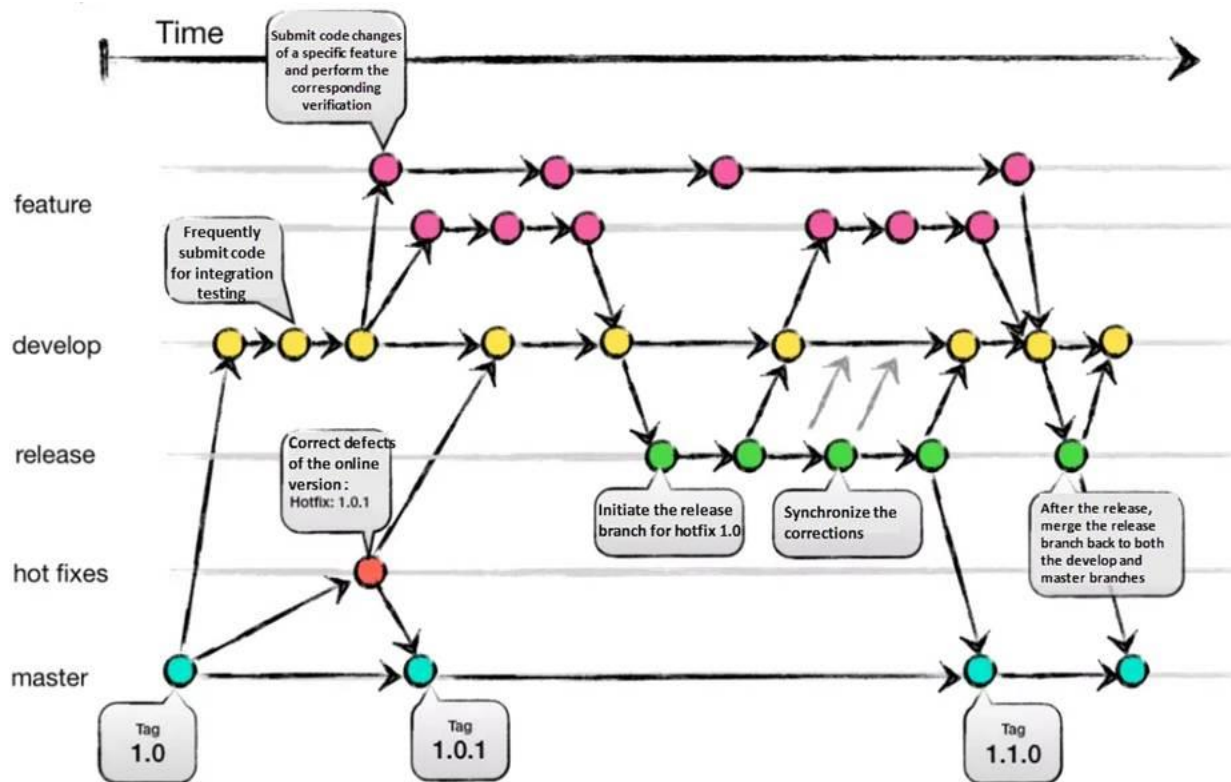


Git Branching Strategy



1. Main Branch (master/main): Only merge branches that have been thoroughly tested and accepted.
2. Develop Branch: Feature branches and hotfix branches are merged into this branch.
3. Feature Branches: Develop new features. Once completed, merge into the develop branch
 - Branch name:
 - Implement: features/Implement_UserStoryName
 - Create or update design: features/Design_UserStoryName
4. Hotfix Branches: Fix critical bugs in the code. Once completed, merge into the main and develop branches.
5. Release Branches: Prepare the code for release. Whenever have a release branch must have a release note. Can refer: [April 2025 \(version 1.100\)](#)

Pull Request Format:

- Name pull request needs to be clear.
- Format:
 - [Implement] User Story Name
 - [Design] User Story Name
- Description needs to have a DoD (Definition of Done) to make sure the user story is done.
 - ☐ Code has been fully written and is correct.
 - ☐ Unit tests have been performed.
 - ☐ Code has undergone self-review.
 - ☐ Adheres to best practices.
 - ☐ Design documentation is created for new features or updated for changes.
 - ☐ Approved by the reviewer.
 - ☐ The code implemented must match acceptance criteria

Commit:

- Should commit clearly and can readable
- Do not use: Update, Create,
- Example: For implement feature Login, the commit should be:
 - *Create API login.*
 - *Create login service.*