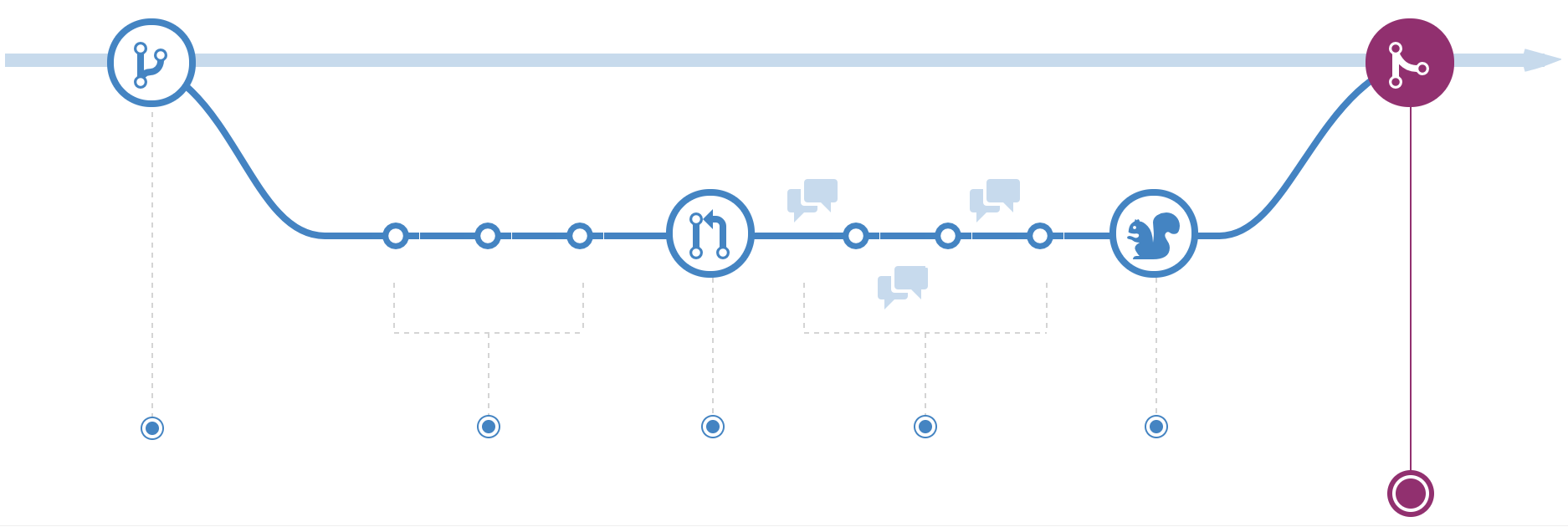
**GIT Considerations**

1. Use git to maintain below coding components only not configuration(we can aslo maintain configuration items in git but it’s very hard to merge when with Change Request’s changes)
   1. Apex Classes
   2. Aura Components
   3. Triggers
   4. Visualforce pages
   5. Visualforce Components
   6. Custom Labels
2. Use separate sandboxes for Phase 2 development and Change Requests
   1. TMDEV sandbox for Phase2
   2. DEV sandbox for CRs and Hotfixes
3. Branches :
   1. Master Branch – It’s long and stable branch.
   2. Feature Branch – short-term branch, it is for Phase 2 development and will be used in TMDEV sandbox, after merge into master branch, Can delete it later.
   3. Hotfix Branch – short-term branch, Developer create it when fix defects in PRODUCTION and will be tested in DEV sandbox. after merge into master branch, Can delete it later.
4. GitHub Flow:

Reference URL: <https://guides.github.com/introduction/flow/>

* 1. Create a Branch before starting new task.
  2. Add commits when complete task and tested it successfully in SF.
  3. Open a pull request.
  4. Discuss and review the code.
  5. Merge code into master branch.



1. Git Overview



Workspace：Working Directory

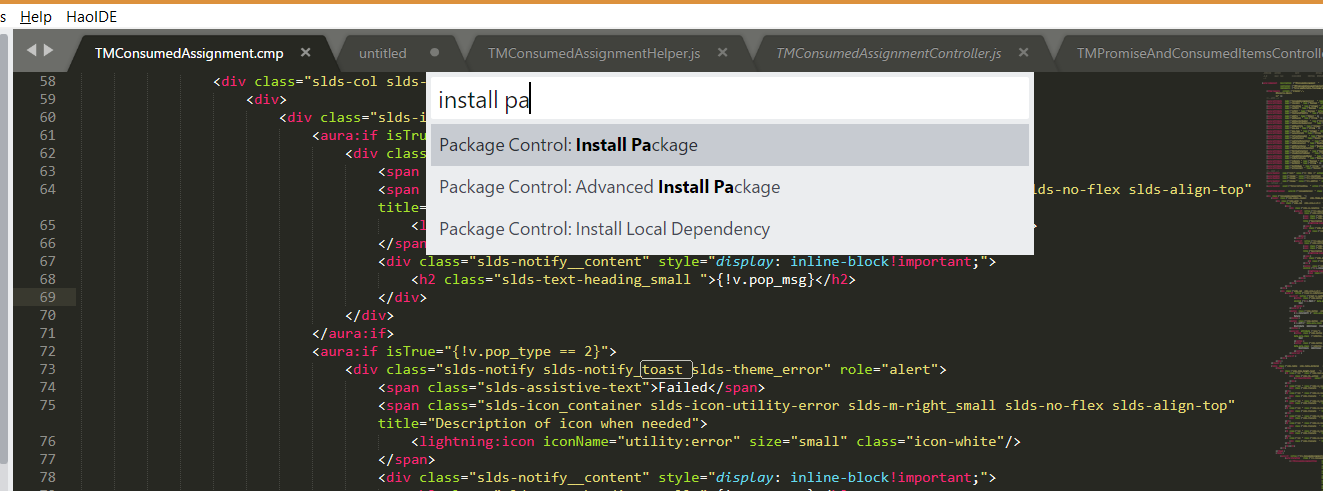
Index / Stage：Stage area

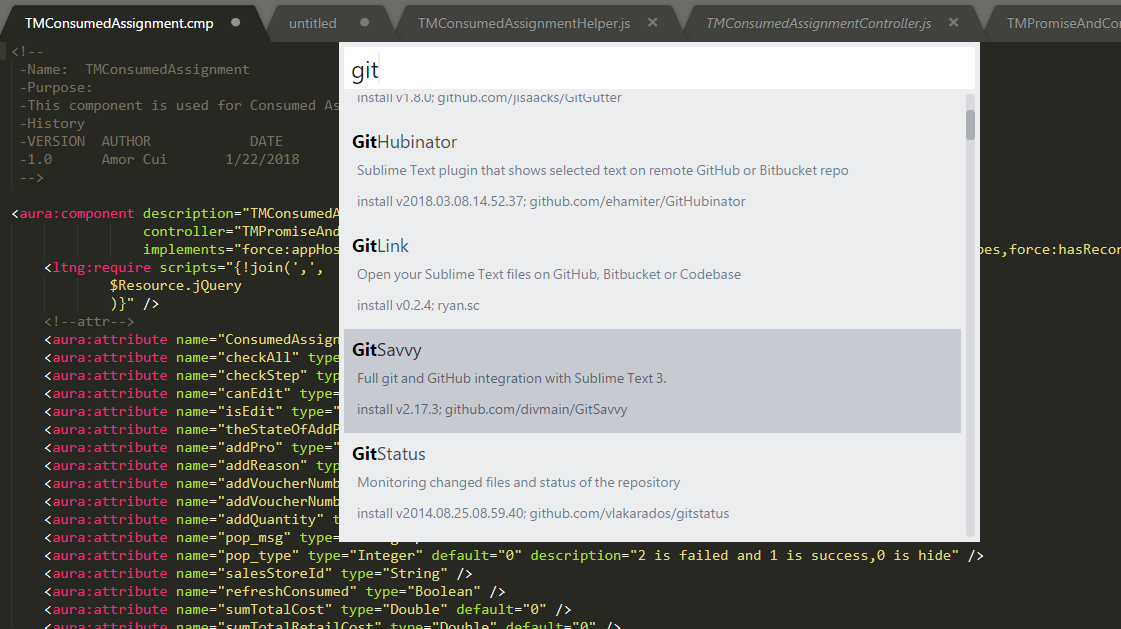
Repository：Local repository

Remote：Remote repository

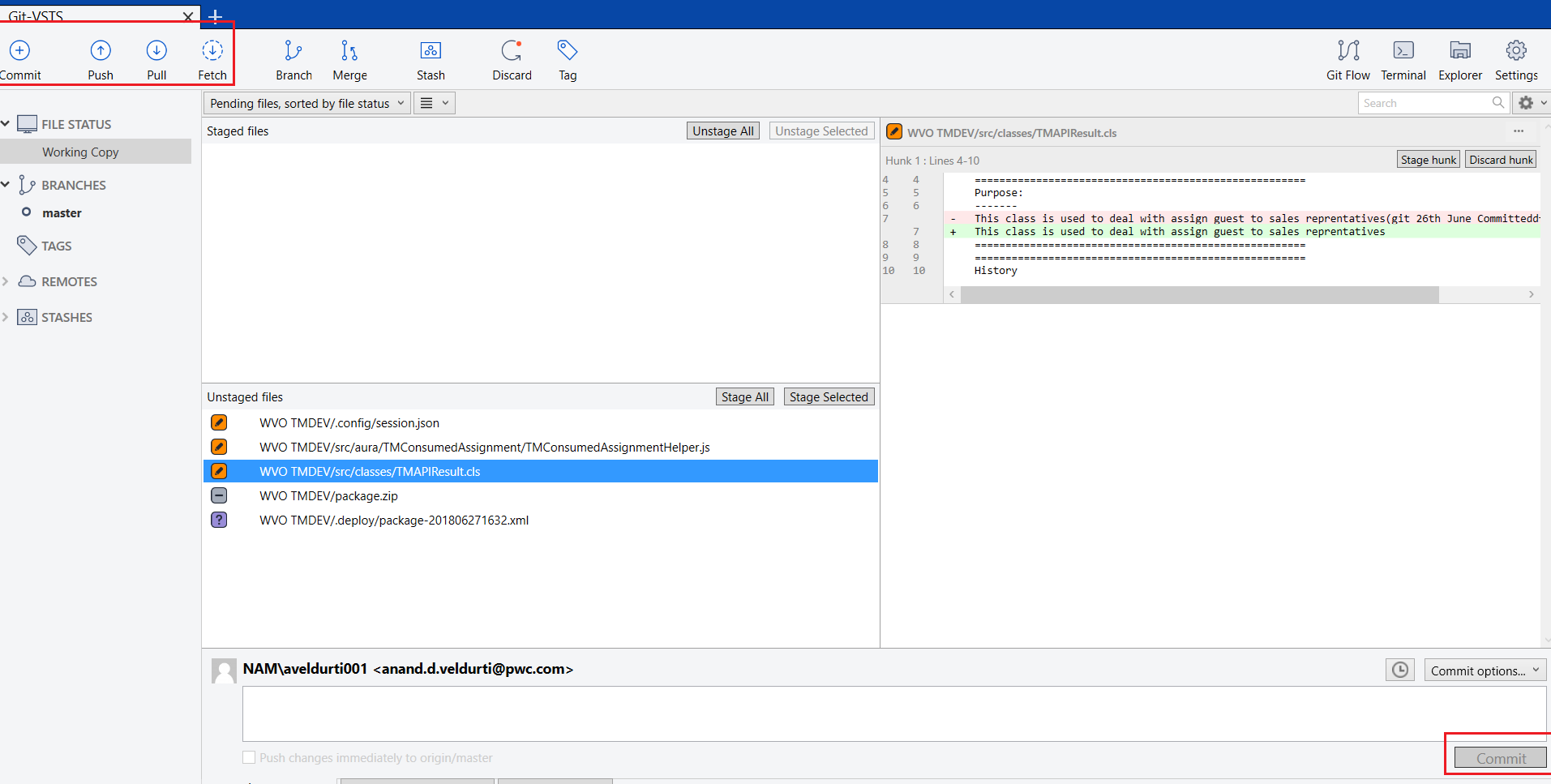
Git Configuration

1. Create repository in git and get the URL
2. Setup git in IDE (using git plugin in Ecplise, using git package control Sublime)
3. Navigate to package manager and type “install package” (cntl+alt+p)





1. Install the git client (Sourcetree, Tortoise, git batch) to perform push/pull/merge operations to git repository.
2. Below you can see the git client “**SourceTree**” with recently changed components.



1. If multiple users perform push on same class then git client will give a error message like below.

