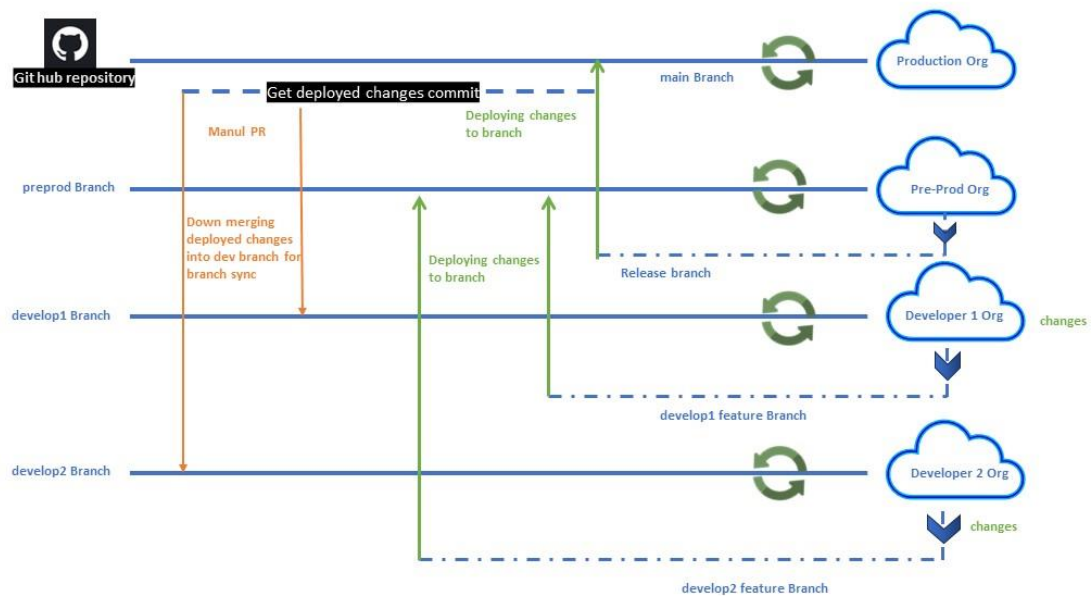# Deployment Procedure

Pre-requisites:

1. **Install SFDX cli** link
2. **Install git** link
3. **Install VS Code** link
4. **Install GitHub desktop** link
5. **Install Salesforce Extension Pack (Expanded) in VS code** link

Note: Ensure environment variables are set properly after installation of above apps, and update sfdx, git versions to latest.

➔ **Project Branching strategy:**

Branching strategy

Git hub repository

Get deployed changes commit

main Branch

Production Org

Manul PR

Deploying changes to branch

preprod Branch

Pre-Prod Org

Down merging deployed changes into dev branch for branch sync

Deploying changes to branch

Release branch

develop1 Branch

Developer 1 Org

changes

develop1 feature Branch

develop2 Branch

Developer 2 Org

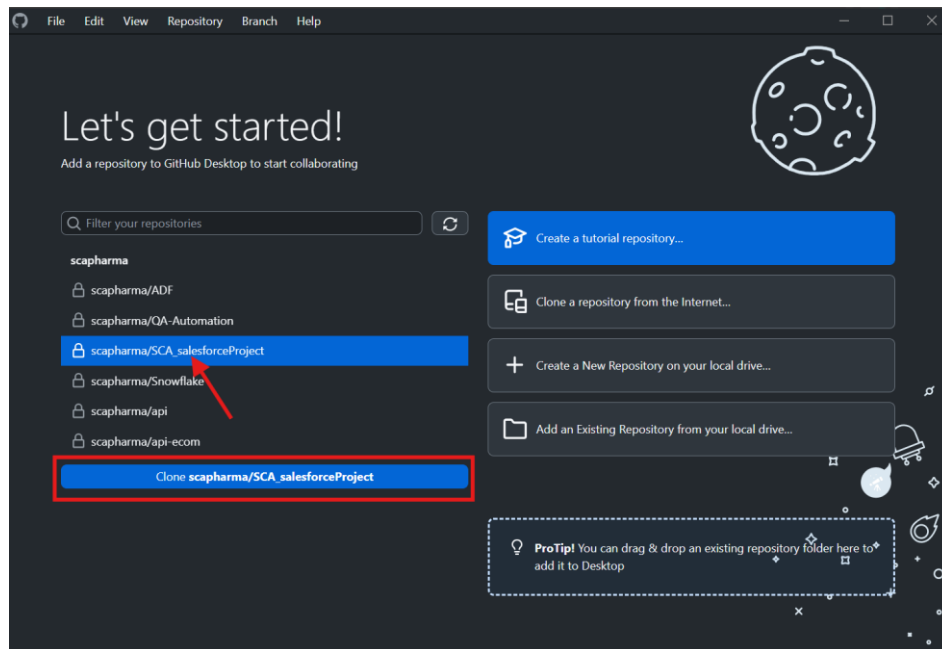changes

develop2 feature Branch

Author: Kiran Sreeram Prathi

➔ **Clone repository to local machine:**

1. After successful installation of above list software's open GitHub desktop and login with your credentials.
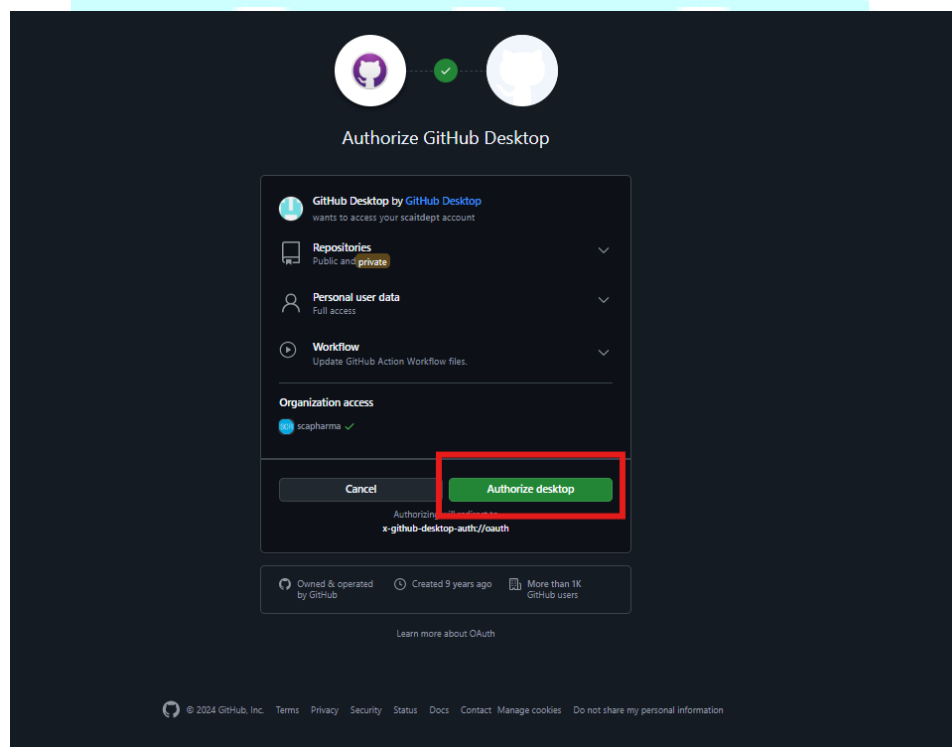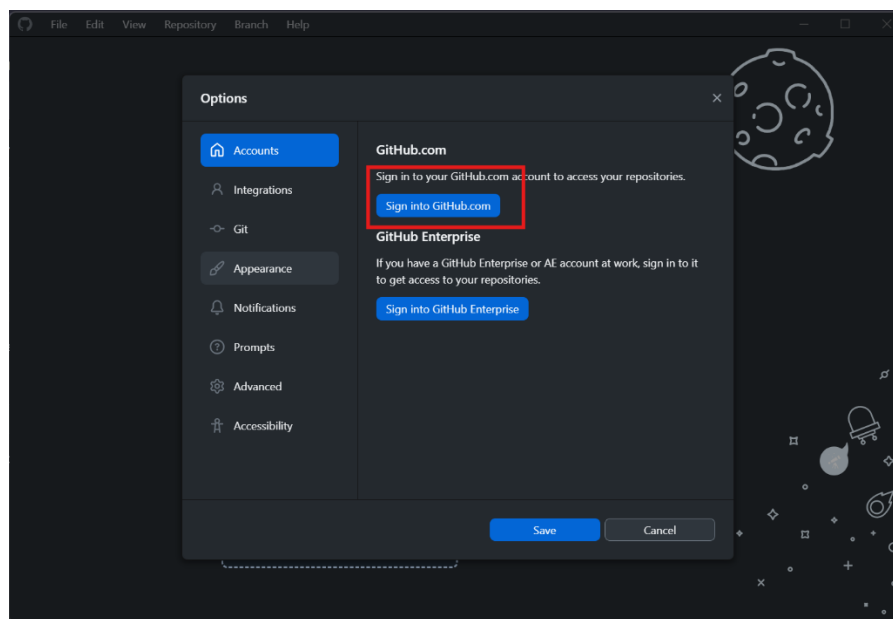
2. Click on clone repository and choose folder to store cloned repo.

➔ **Creation of feature branch:**
   1. In this branch strategy feature branch is used to deploy changes from **developer sandbox** to **Pre-Production.**
   2. Open GitHub desktop => click on current branch "main"
   3. Select "**preprod**" branch.
   4. After checking out to "**preprod**" branch click on "**Fetch origin**" this will pull changes from remote and refreshes the local cloned repo.
   5. Then again click "Current branch" => New Branch => enter branch name like below pattern.
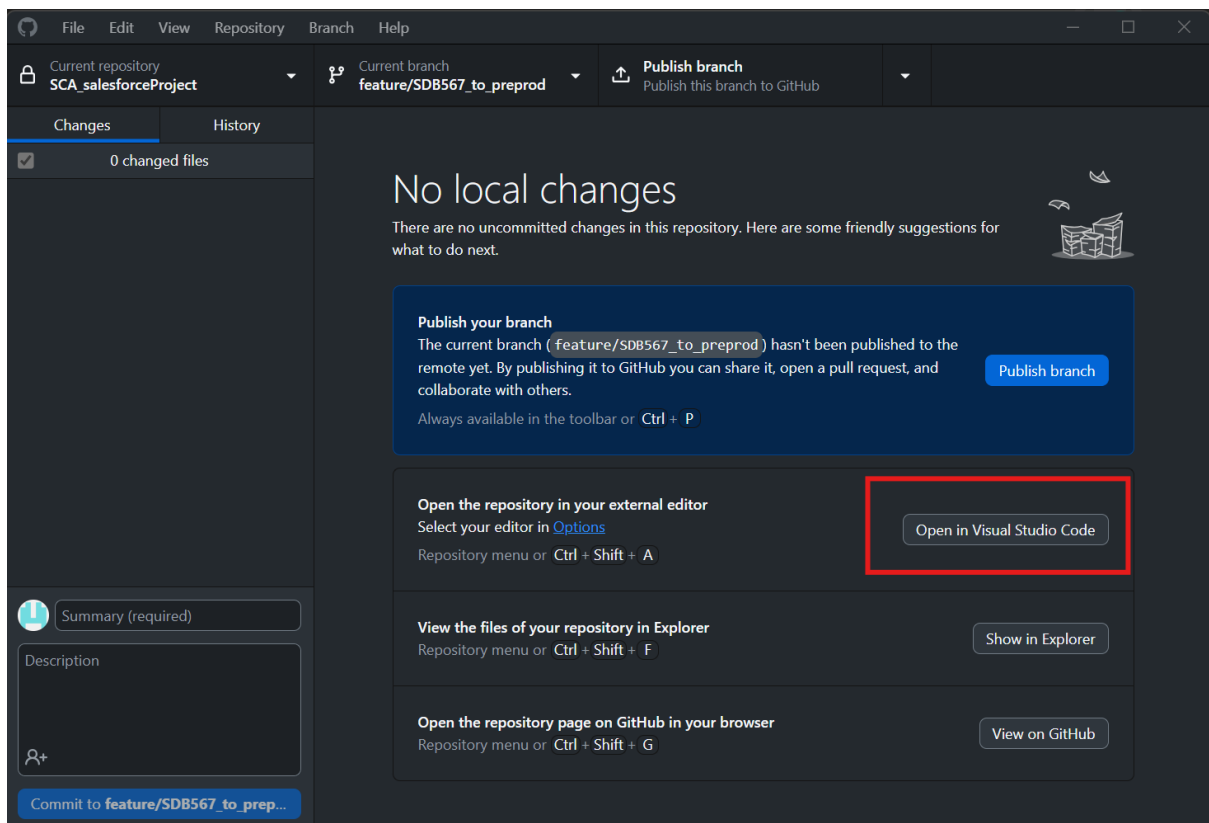
   Pattern: feature/jiraTicketName_to_targetBranch

   **(Ex:** feature/SDB567_to_preprod)

   6. And select "preprod" for create branch based on. the select create branch.
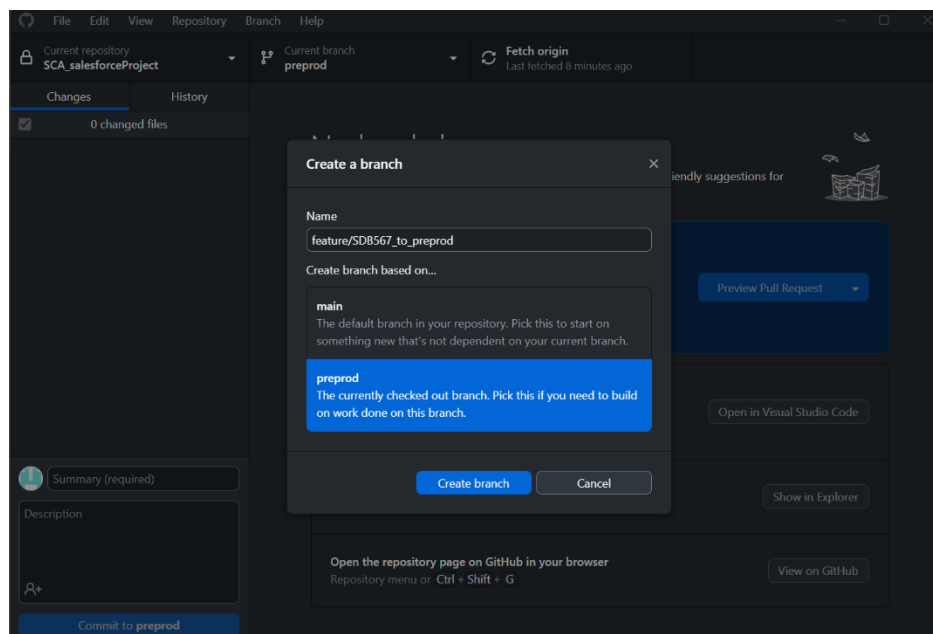
7. After branch creation click on "**Open in visual studio code**"

8. After opening visual studio code, you can see checkout branch, and set default org and create package.xml to retrieve modified files/components from sand box.
9. After creation of package.xml, right click on package.xml and select retrieve source in manifest from Org.
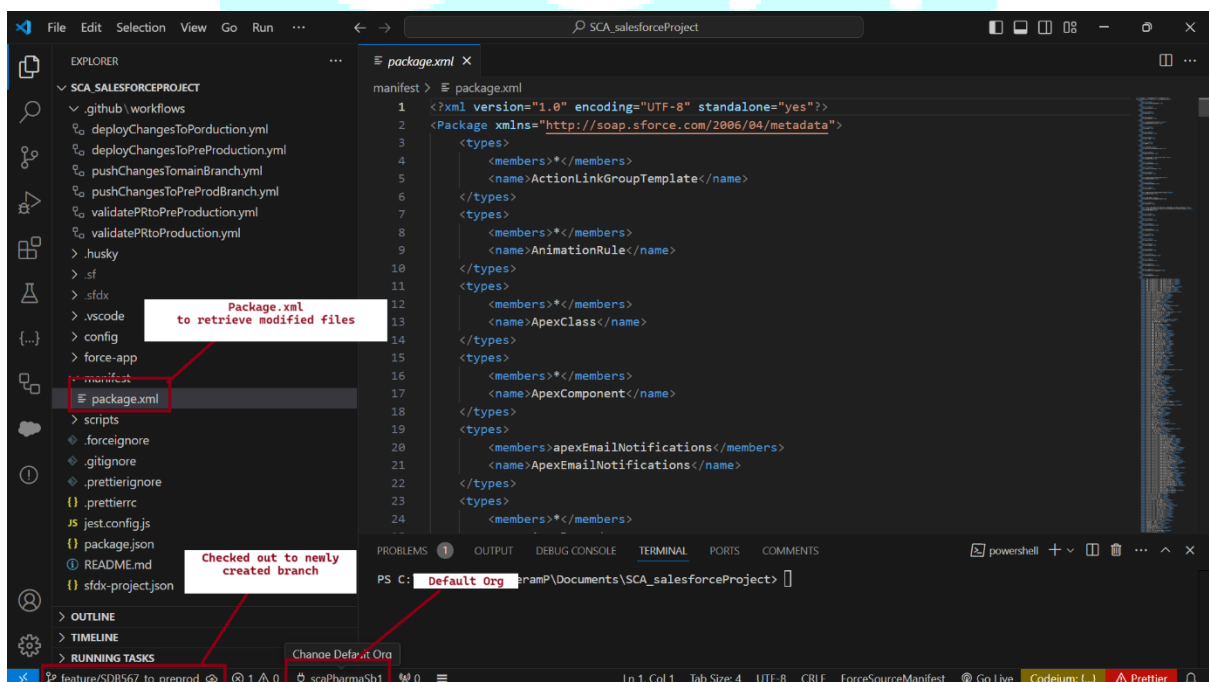
10. After successful retrieval of meta-data from org you can see files changed in source control (VS code).
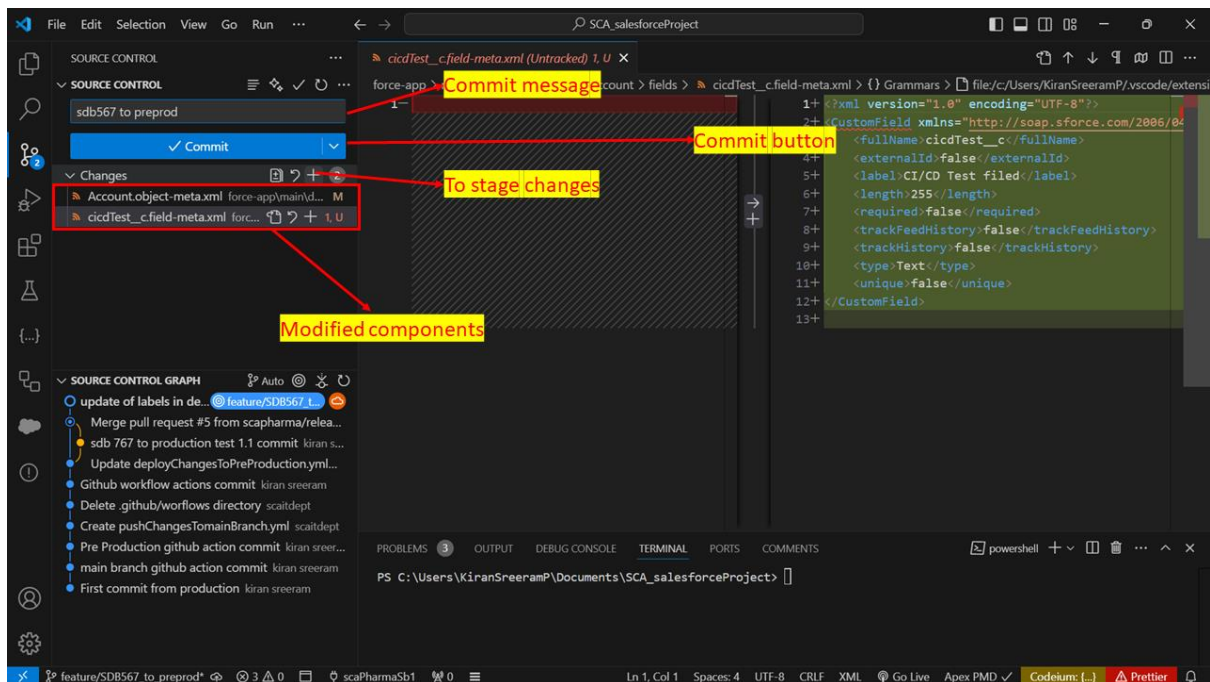
Click on "**+**" icon to stage all changes. (Note: while staging do not stage package.xml and. yml type files)

11. After staging click "**commit**" and open GitHub desktop and click on "**Publish**".
12. After publishing branch click on "**Preview Pull request**", it will open pop-up and then verify your changes and click "**Create Pull request**" where it will redirect to browser.
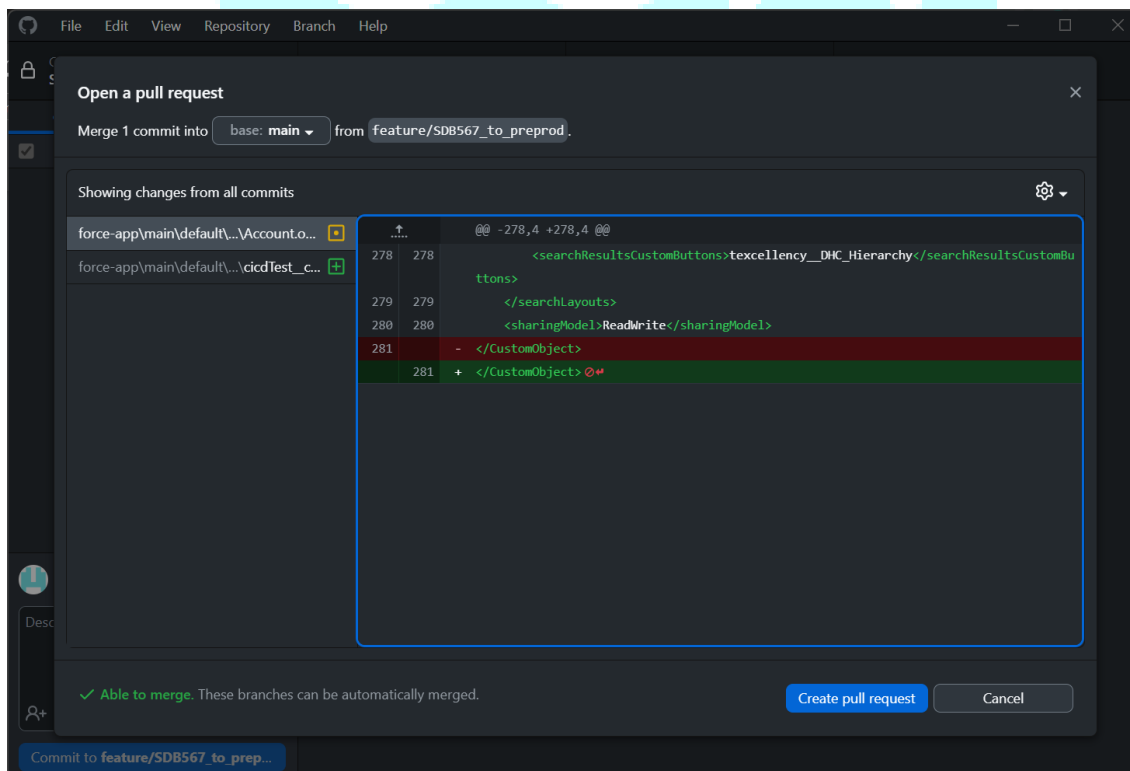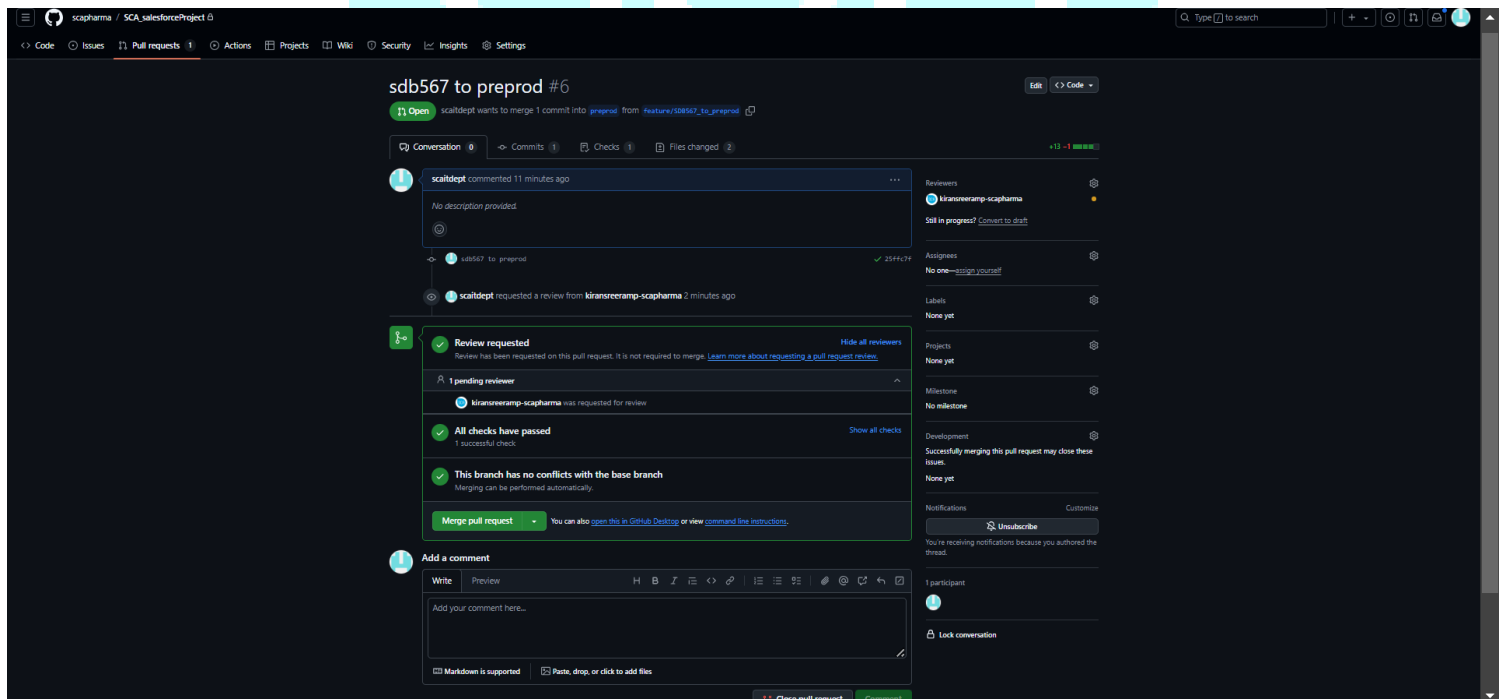
13. After redirection to browser click on "**Create Pull request**" button, so the PR will be created, and validation will start.
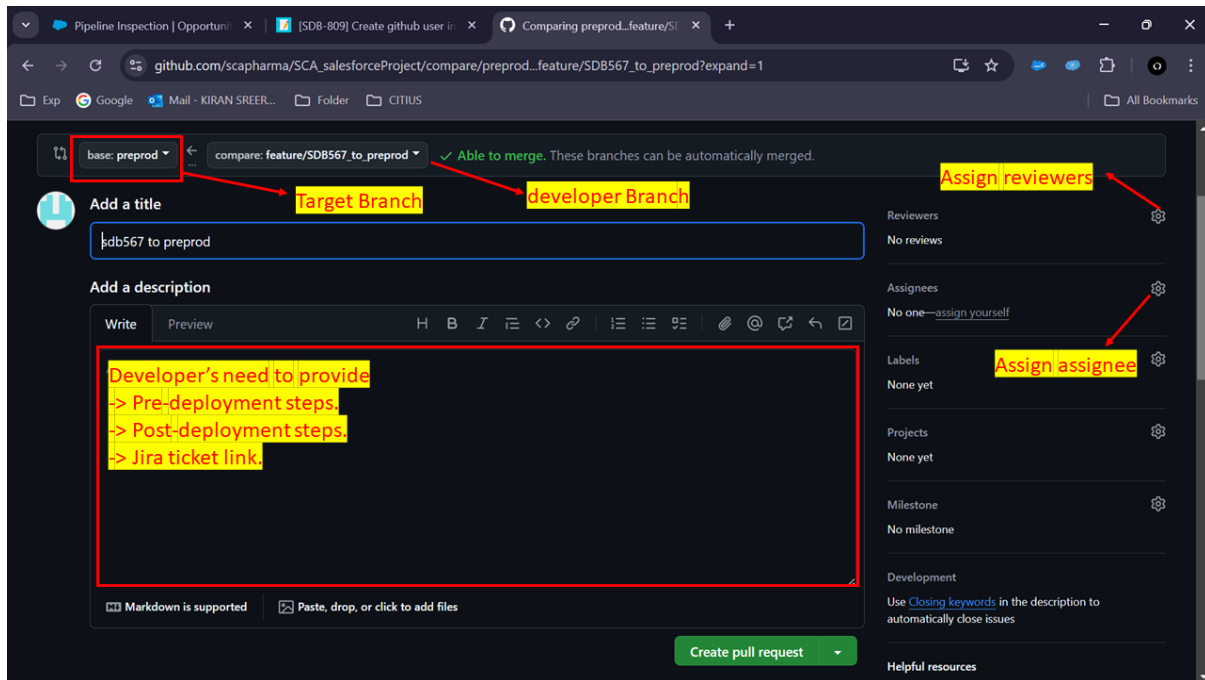
➔ **For code reviewers and Deployment Team**
14. Once validation was successful and post checking the changes and reviewer approved changes deployment team can merge pull request by clicking on Merge Pull request and will click on confirm merge.

➔ **For to Deploy changes into production.**

1. Release team need to create release branch based on main branch in below format.

   Format: release/dateOfRelease_to_prod.

   (Example: release/10Oct2024_to_prod)

2. Commit all successful PRs into created release branch which are raised to pre-prod from developer sandbox.
3. Resolve if any merge conflicts arise.
4. Raise pull request as same above mentioned and assign approvers, once approved and validation check successful merge changes.
5. Once you click merge changes the changes will merge into branch and at a same time the changes will get deployed into target org.
6. And verify by checking the deployed components.
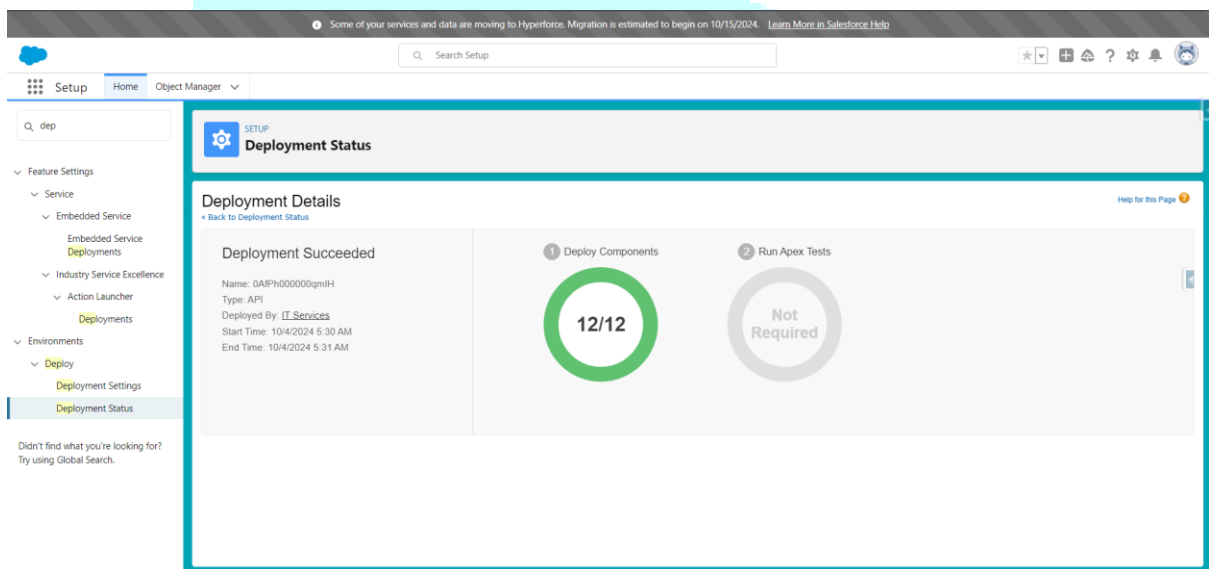
Note: Pre-deployment steps need to be done before merging the changes in GitHub browser otherwise deployment will get failed.

Thank you.

For queries:

Mail: kiransreeram.prathi@citiustech.com