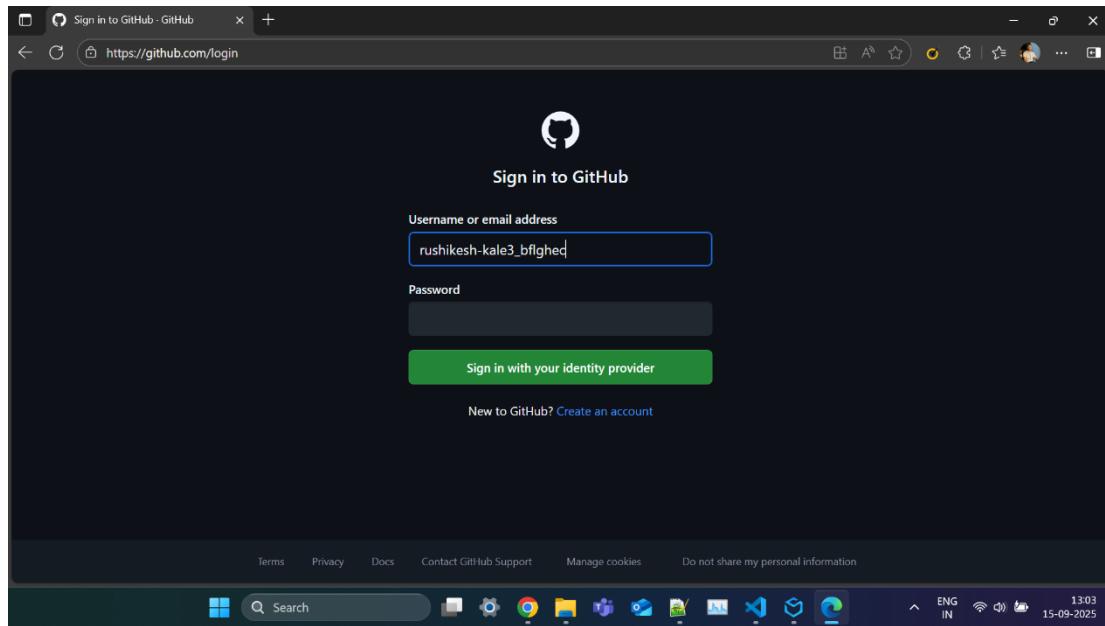


# # GitHub PR, Development And Deployment :

## 1. Login in the GitHub:

- <https://github.com> => GitHub\_User\_Name => Login with SOSL



## 2. Navigate to Repos :

- Profile => Organization -> Select Project => Select Repos

The image consists of three vertically stacked screenshots:

- Screenshot 1: User Profile (Dark Mode)**  
Shows the user's profile with a pink icon, name "rushikesh-kale3...", and email "Rushikesh.Kale4". Below the profile are links for "Set status", "Single-sign on", "Profile", "Repositories", "Stars", and "Organizations". The "Organizations" link is highlighted in green.
- Screenshot 2: Organizations List**  
Shows a list of organizations the user is part of:
  - Copilot-Buss (Member)
  - SFDC-LSF-PROJECT (Owner)
  - SFDC-POS-PROJECT (Owner)** (highlighted in green)
  - SFDC-TWF-PROJECT (Owner)
  - Two-Wheeler-Project-Git (Owner)
- Screenshot 3: SFDC-POS-PROJECT Repository List**  
Shows the repository list for the SFDC-POS-PROJECT organization:
  - Overview
  - Repositories** (highlighted in orange)
  - Projects
  - Packages
  - Teams
  - People
  - Security
  - Insights
  - Settings

The left sidebar shows filter options: All, Public, Internal, Private, Sources, Forks, Archived, and Templates. The main area shows "All" with a search bar and "3 repositories": SFDC-POS-REPO, POS-Bajaj, and B2B-POS-Bajaj.
- Screenshot 4: SFDC-POS-REPO Repository Details**  
Shows the detailed view for the SFDC-POS-REPO repository:
  - Code, Issues, Pull requests, Actions, Projects, Models, Wiki, Security, Insights, Settings
  - Code tab selected
  - Branch: master, 9797 Branches, 0 Tags
  - File list: .github/workflows, .sf, .vscode, config, force-app/main/default, manifest, scripts, .eslintignore, .forceignore, .gitignore
  - Activity: 6,641 Commits, 4 hours ago
  - About: No description, website, or topics provided.
  - Readme, Activity, Custom properties, 0 stars, 0 watching, 0 forks
  - Releases: No releases published, Create a new release
  - Packages: No packages published, Publish your first package

### 3. Create Personal Access Token :

- Profile => Setting => [Developer Setting](#) => Personal Access Token
- Generate New Classic Personal Access token => auth a git repo

The screenshot shows the GitHub developer settings interface. On the left, there's a sidebar with various options like Repositories, CodeSpaces, Models, Packages, Copilot, Pages, Saved replies, Security, Code security, Integrations, Applications, Scheduled reminders, Archives, Security log, Sponsorship log, and Developer settings (which is highlighted). The main area is titled "Personal access tokens (classic)". It displays a single token named "GH\_PAT" with scopes: admin:org, admin:ssh\_signing\_key, repo, workflow, write:packages. A note says "This token has no expiration date." Below this, a section titled "Expiration" shows "No expiration". Under "Select scopes", there are two columns of checkboxes. The first column includes repo, workflow, write:packages, delete:packages, admin:org, and admin:repo\_content. The second column describes each scope: repo gives full control of private repositories, workflow updates GitHub Action workflows, write:packages uploads packages to the GitHub Package Registry, delete:packages deletes packages from the GitHub Package Registry, admin:org provides full control of orgs and teams, and admin:repo\_content manages org runners and runner groups.

- Authorize the Repos for Token :

This screenshot shows the "Personal access tokens (classic)" page again, but with a modal window open titled "Single sign-on organizations". The modal explains that certain organizations require tokens to be authorized for access and provides a search bar and a list of available organizations to authorize. Two organizations are listed: "Two-Wheeler-Project-Git" and "Copilot-Buss", each with an "Authorize" button.

#### 4. Cloning Repos :

- Open terminal => git clone GITHUB\_URL

```
PS C:\Users\rushikeshkale3\Desktop>New folder (2)> git clone https://github.com/SFDC-POS-PROJECT/SFDC-POS-REPO.git
```

- Login in Terminal :

- o window =>

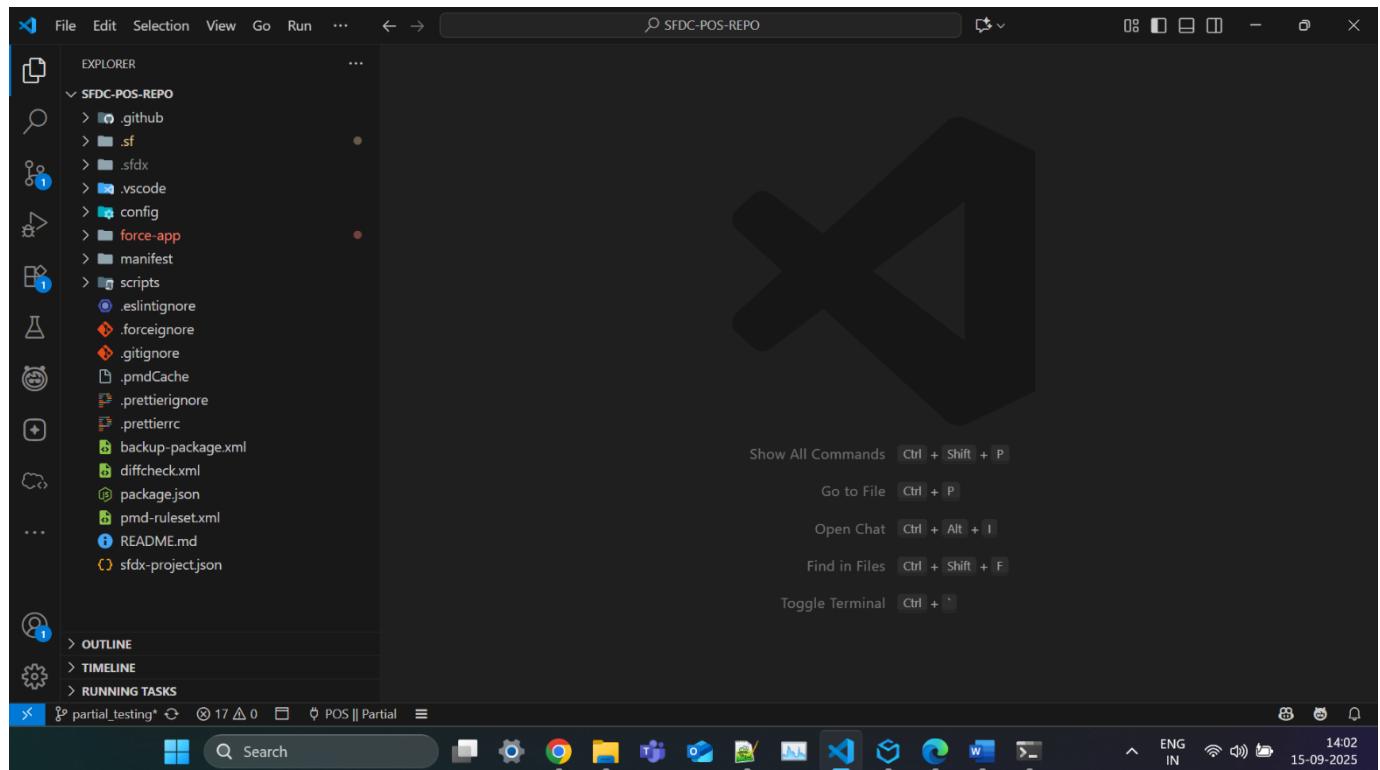
- Auto Logging => Browser Based Login

- o mac =>

- username : GitHub\_User\_Name

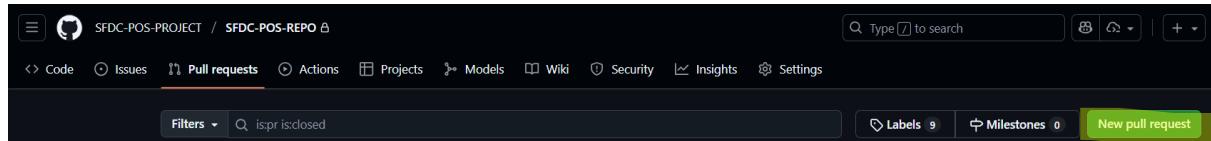
- Pass : Personal Access Token generated in the Previous step

- Open Vs Code in the Cloned Folder :

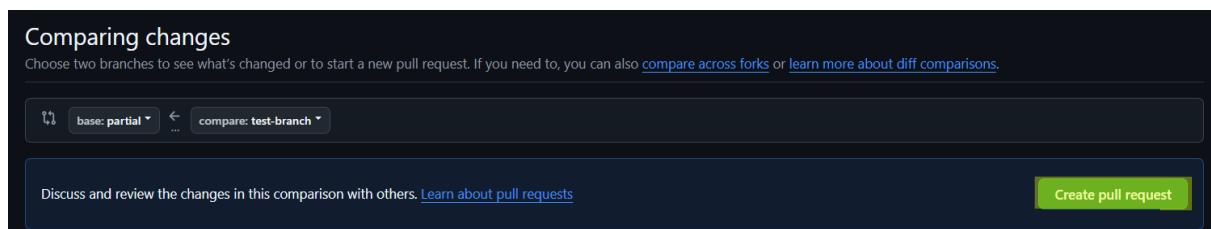


## 5. Raised the PR And Error Checking :

- Create Branch From Base Branch in VS Code[dev/partial/preprod/master]
- In your New Branch => Do Changes in the respective Classes and Code
- Update Package.xml file
- Commit the changes and Publish You Branch
- Go to GitHub in PR Section



- Select you branch and base branch [like : test-branch into => partial ]
- And Create Pull Request



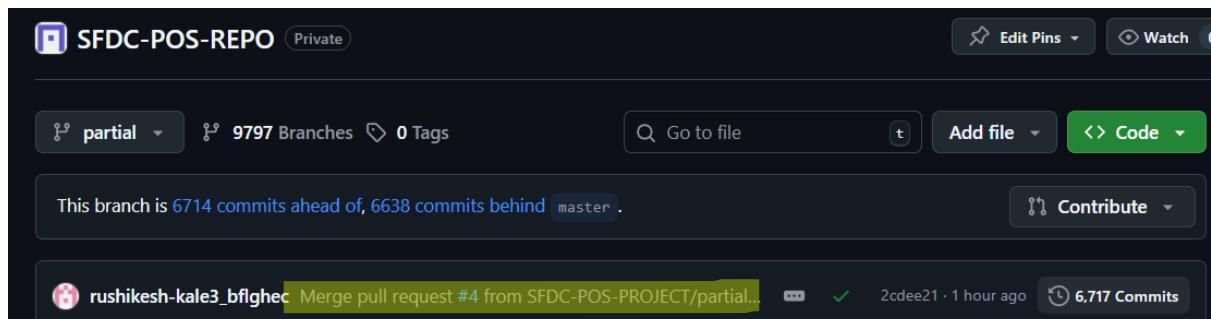
Comparing changes

Choose two branches to see what's changed or to start a new pull request. If you need to, you can also [compare across forks](#) or [learn more about diff comparisons](#).

base: partial compare: test-branch

Create pull request

- After approval of the reviewer, You can Merge PR .
- Checking PR Commit and Release
- Code => select Base Branch => commits => your PR



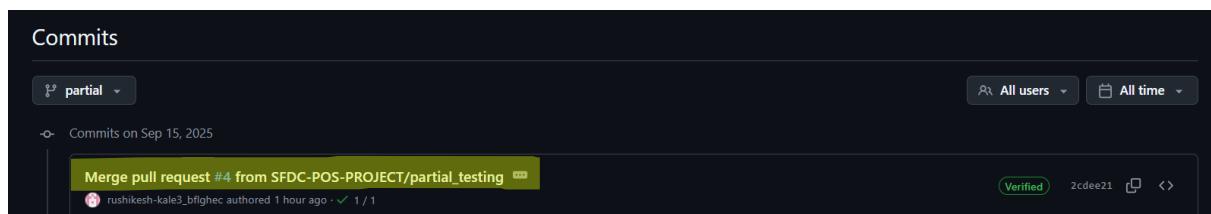
SFDC-POS-REPO Private

partial 9797 Branches 0 Tags

This branch is 6714 commits ahead of, 6638 commits behind master.

rushikesh-kale3\_bfighec Merge pull request #4 from SFDC-POS-PROJECT/partial... 2cdee21 · 1 hour ago 6,717 Commits

- Check Merge Commit



Commits

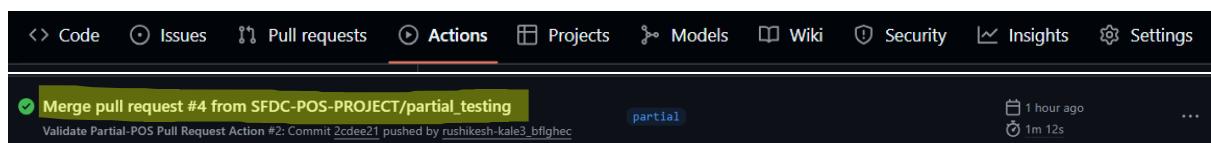
partial

Merge pull request #4 from SFDC-POS-PROJECT/partial\_testing

rushikesh-kale3\_bfighec authored 1 hour ago 1 / 1

Verified 2cdee21

- Check Status of the Merging in the respective org :



Actions

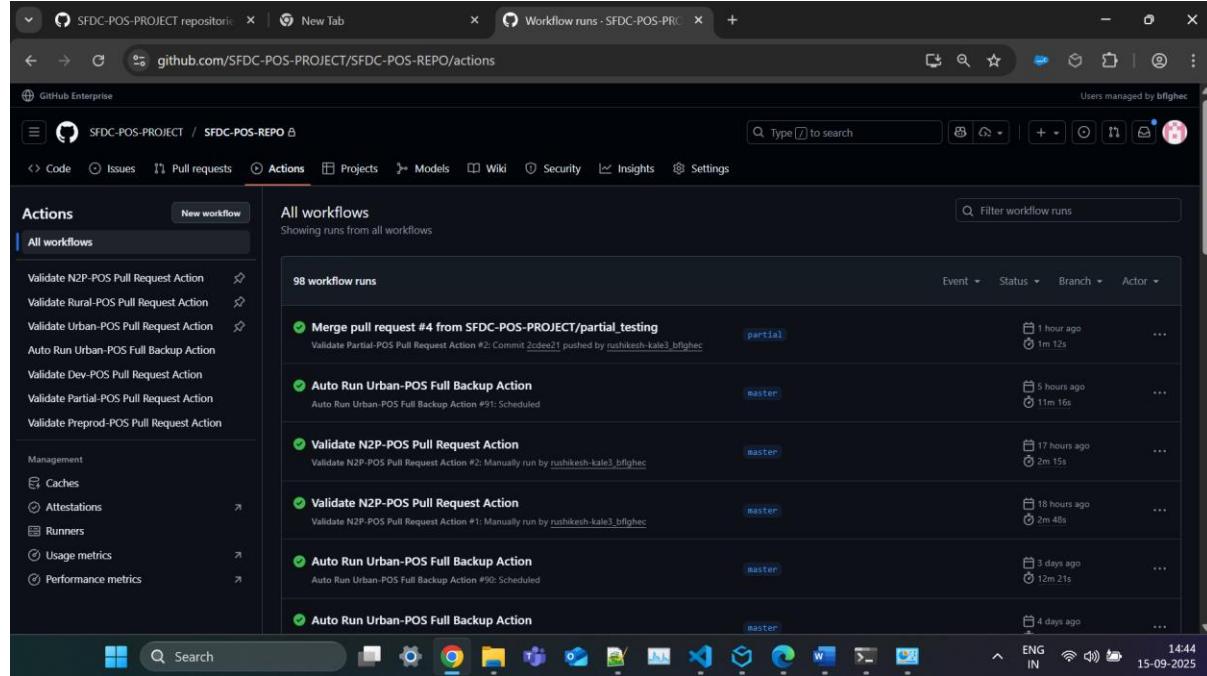
Merge pull request #4 from SFDC-POS-PROJECT/partial\_testing

Validate Partial-POS Pull Request Action #2: Commit 2cdee21 pushed by rushikesh-kale3\_bfighec

1 hour ago

## 6. Production Deployment Details :

- Action Tab

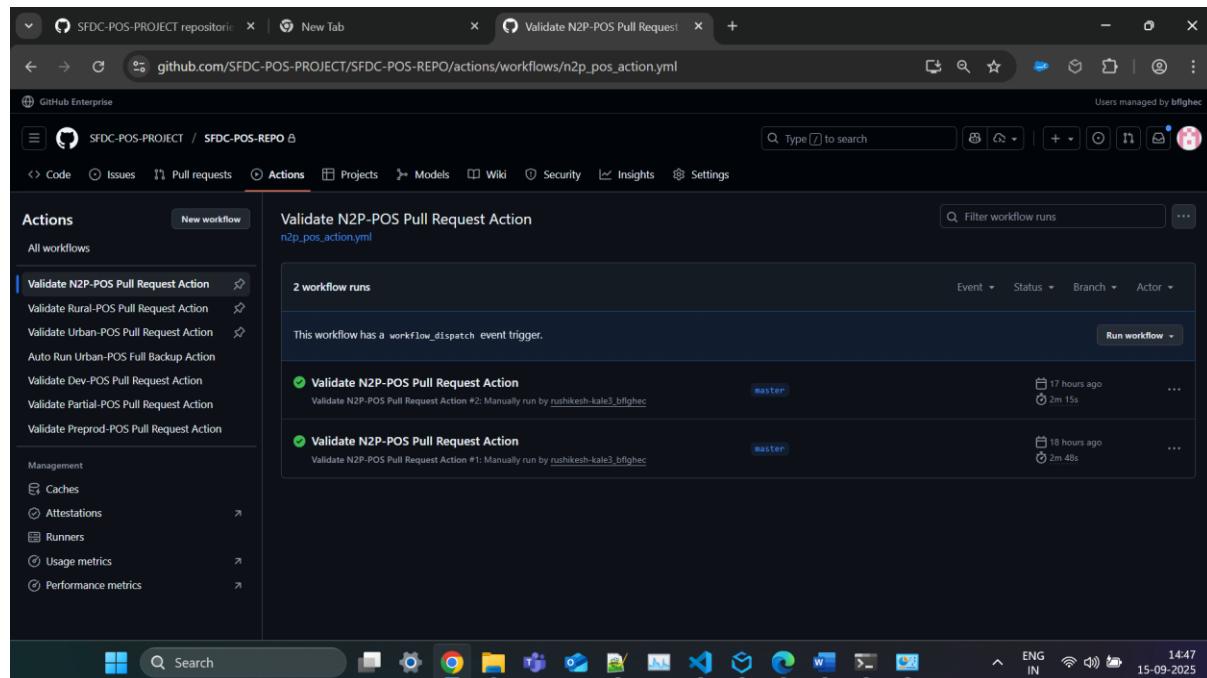


The screenshot shows the GitHub Actions page for the SFDC-POS-PROJECT repository. The left sidebar has 'Actions' selected. The main area shows 'All workflows' with 98 workflow runs. The runs are listed in descending order of age, with the most recent at the top. Each run includes a status icon (green checkmark), the workflow name, the event type (partial or scheduled), the branch (master), and the duration.

Workflow Run	Status	Event	Branch	Duration
Merge pull request #4 from SFDC-POS-PROJECT/partial_testing	partial	1 hour ago	master	1m 12s
Auto Run Urban-POS Full Backup Action	master	5 hours ago	master	11m 16s
Validate N2P-POS Pull Request Action	master	17 hours ago	master	2m 15s
Validate N2P-POS Pull Request Action	master	18 hours ago	master	2m 48s
Auto Run Urban-POS Full Backup Action	master	3 days ago	master	12m 21s
Auto Run Urban-POS Full Backup Action	master	4 days ago	master	...

- Select the Work Flow which you have to Run :

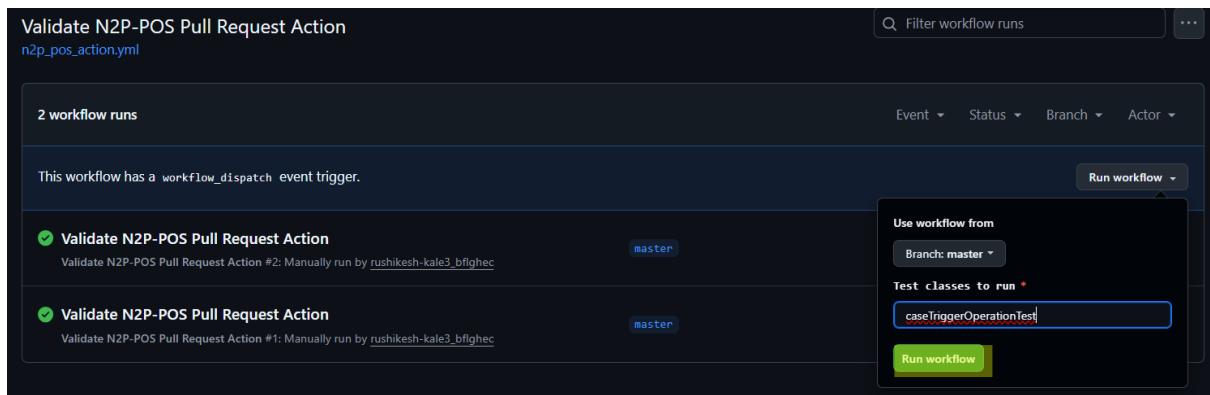
- Validate N2P-POS Pull Request Action – Rural N2P Package Validation
- Validate Rural-POS Pull Request Action – Rural Package Validation
- Validate Urban-POS Pull Request Action – Urban Package Validation
- Auto Run Urban-POS Full Backup Action – Urban Production Full Backup



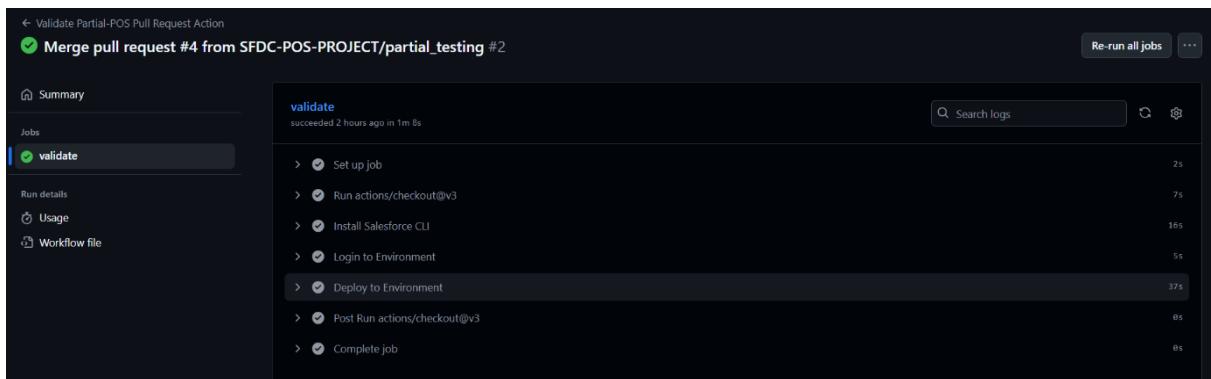
The screenshot shows the GitHub Actions page for the 'Validate N2P-POS Pull Request' workflow. The left sidebar has 'Actions' selected. The main area shows the workflow details and its runs. The workflow has a 'workFlow\_dispatch' event trigger. There are two workflow runs listed, both of which were manually triggered by 'rushikesh-kale3.bfghec'. The runs show the status, event type (partial), branch (master), and duration.

Workflow Run	Status	Event	Branch	Duration
Validate N2P-POS Pull Request Action	partial	17 hours ago	master	2m 15s
Validate N2P-POS Pull Request Action	partial	18 hours ago	master	2m 48s

- Run Workflow [ If test String Required Put it space separated]:

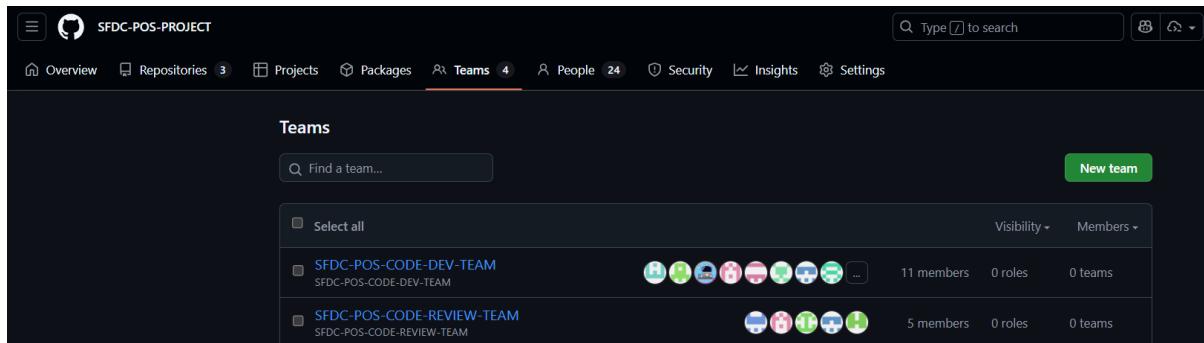


- After Workflow Run, we can check the status by clicking on the particular action



## 7. Admin Part :

- Add user According to Roles :
  - o Owner -> People -> Add Member -> Owner
  - o Code Review -> Teams -> **SFDC-POS-CODE-REVIEW-TEAM**
  - o Developer -> Teams -> **SFDC-POS-CODE-DEV-TEAM**
- Developer and Code Review :

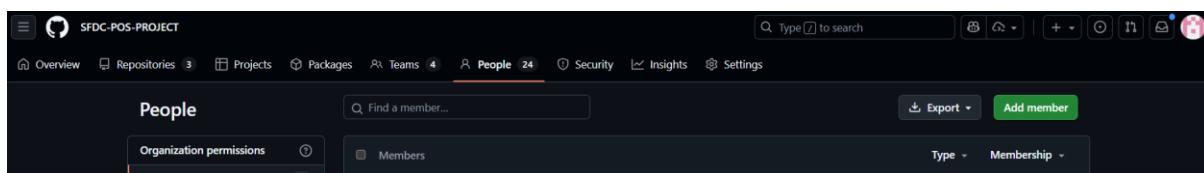


The screenshot shows the GitHub Project interface for 'SFDC-POS-PROJECT'. The 'Teams' tab is selected. There are two teams listed:

- SFDC-POS-CODE-DEV-TEAM**: 11 members, 0 roles, 0 teams. It has 11 profile icons.
- SFDC-POS-CODE-REVIEW-TEAM**: 5 members, 0 roles, 0 teams. It has 5 profile icons.

A 'New team' button is located in the top right corner of the 'Teams' section.

- Owner :



The screenshot shows the GitHub Project interface for 'SFDC-POS-PROJECT'. The 'People' tab is selected. It displays organization permissions and membership details:

- Organization permissions**: Shows a dropdown menu.
- Members**: Shows a table with columns for Type and Membership.
- Add member** button: Located in the top right corner.

- Permission For Roles :
  - o Owner -> Can Add New Member
  - o Code Review -> Can Approve PR
  - o Developer -> Raised the PR

## 8. Note :

### 1. Git Command :

- a. git clone {GITHUB\_URL} => cloning into local machine
- b. git fetch -all => update old local repos
- c. git remote -v => check added remote URL
- d. git remote set-url origin NEW\_URL\_WITHTOKEN

### 2. It Is Good Practice to Set Remote with Personal Access Token :

- a. git remote -v => will get URL

```
PS C:\Users\rushikeshkale3\Desktop\New folder (2)\SFDC-POS-REPO> git remote -v
origin  https://github.com/SFDC-POS-PROJECT/SFDC-POS-REPO.git (fetch)
origin  https://github.com/SFDC-POS-PROJECT/SFDC-POS-REPO.git (push)
```

- b. NEW\_URL\_WITHTOKEN = https:// 'access\_token@' existing\_url[from github]
- c. git remote set-url origin NEW\_URL\_WITHTOKEN

### 3. Error : [ repository not found ] :

- a. Navigate to : Control Panel => User Accounts => Credential Manager => Remove For GitHub

