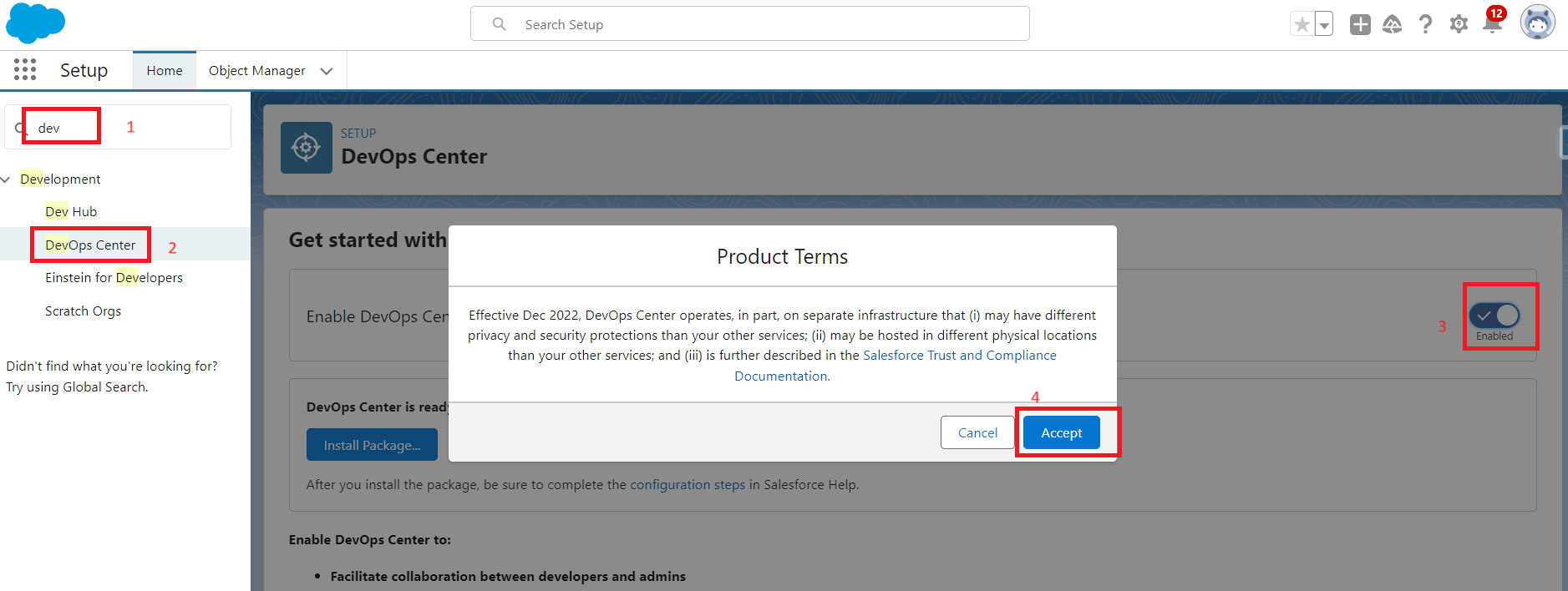
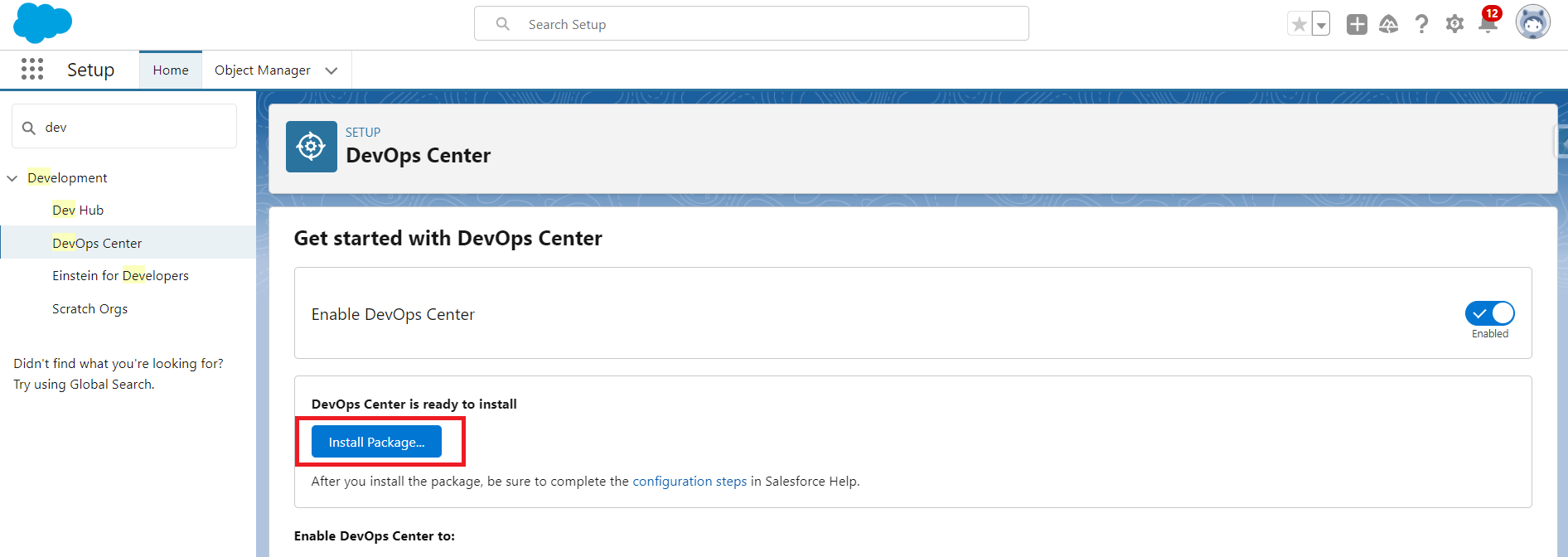
**Install and Configure DevOps Center**

Step:1 From Setup, enter DevOps Center in the Quick Find box, then select DevOps Center.

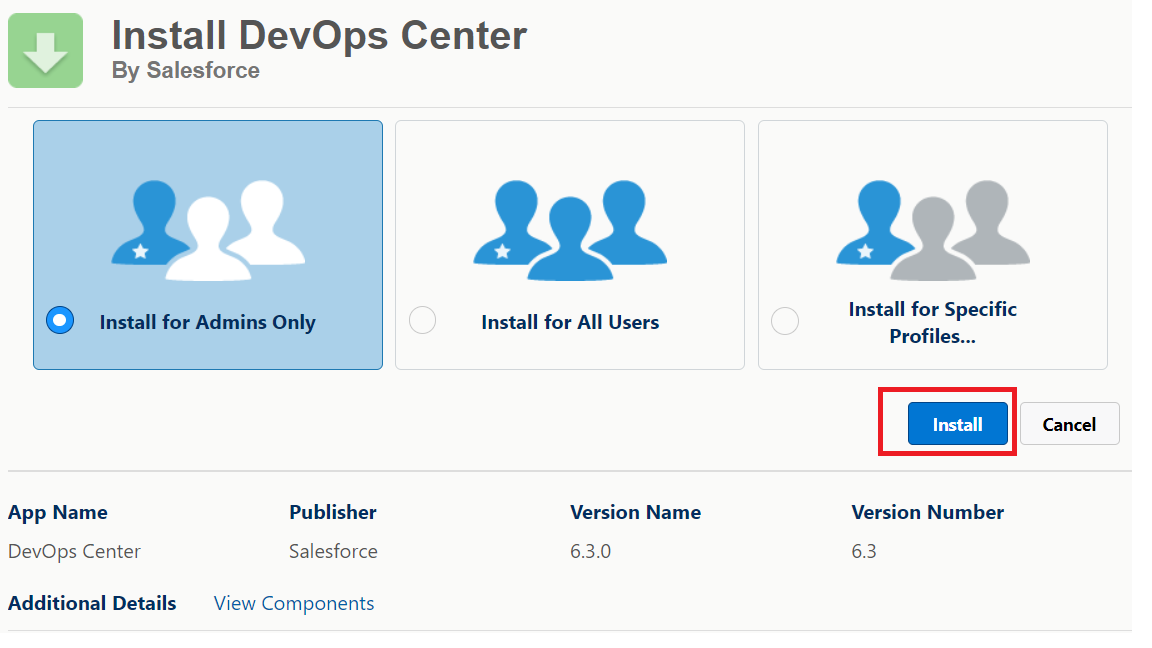
Step:2 Click the toggle to enable DevOps Center, and then review and accept the license agreement.



Step:3 Click on the “Install Package” button to proceed.



Step:4 Select Install for Admins Only, and then click Install.

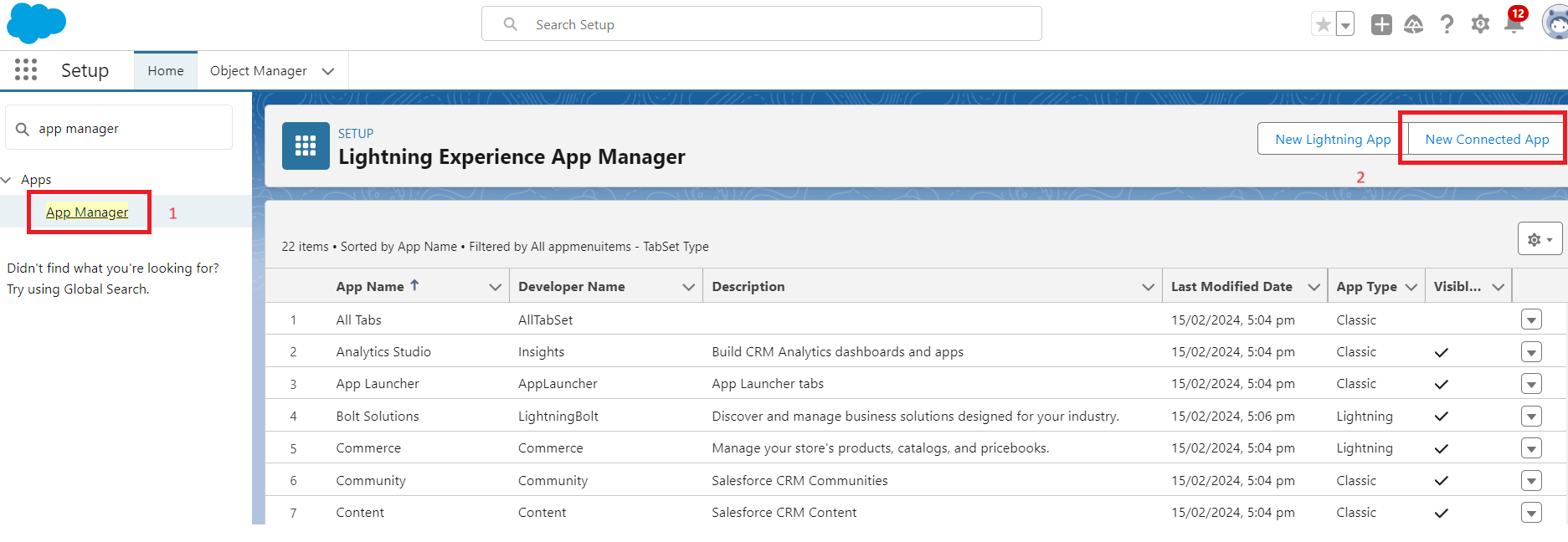


successfully installed the devOps center package. Now let’s create a connected App

**Create a connected app**

Step:1 From Setup, enter App manager in the Quick Find box, then select App manager.

Step:2 Click on “New Connected App” to create a connected app



Step:3 Fill the Basic information:

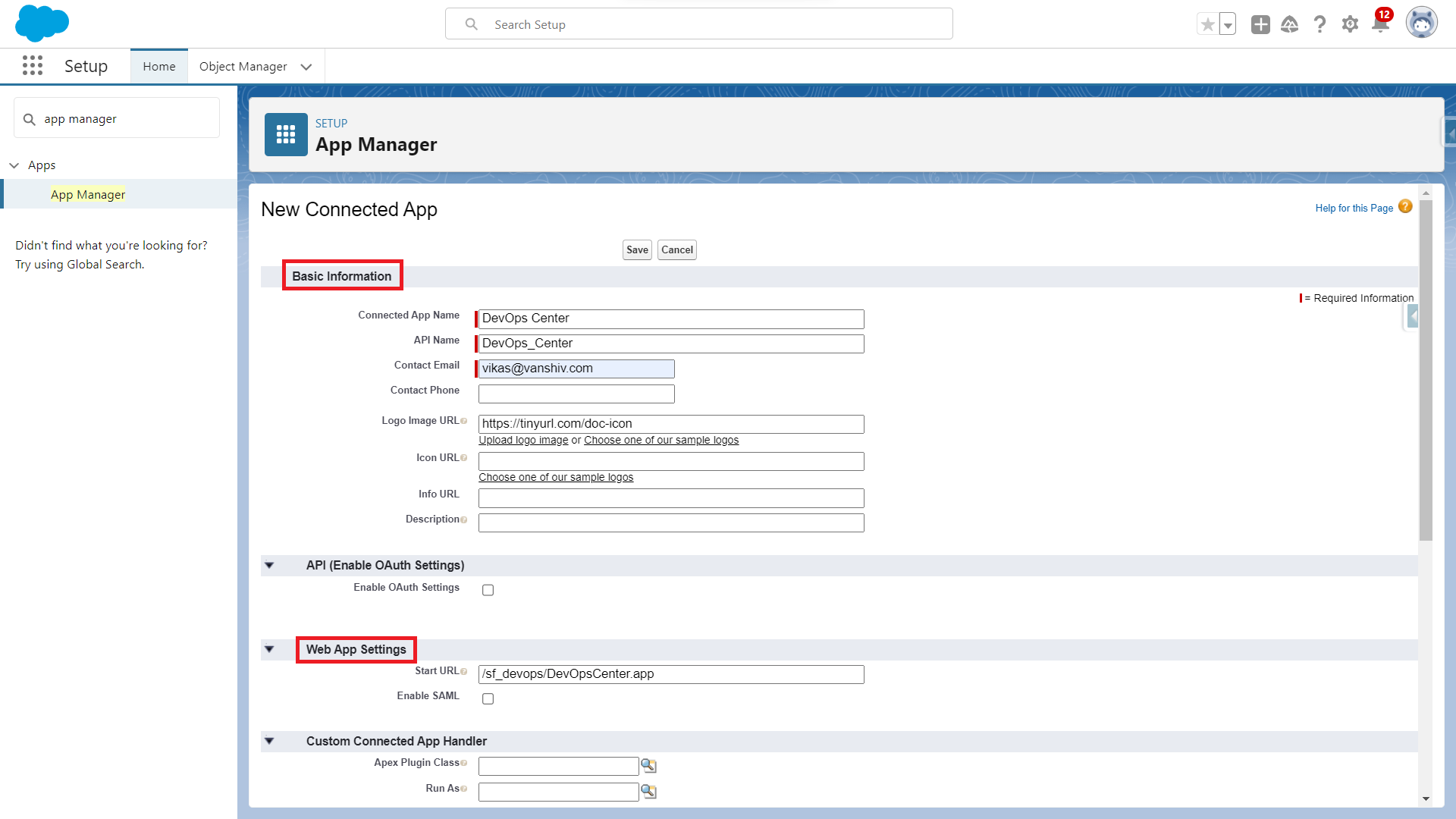
Connected App Name: DevOps Center

API Name: DevOps\_Center

Contact Email: Your Email Address

Logo Image URL: <https://tinyurl.com/doc-icon>

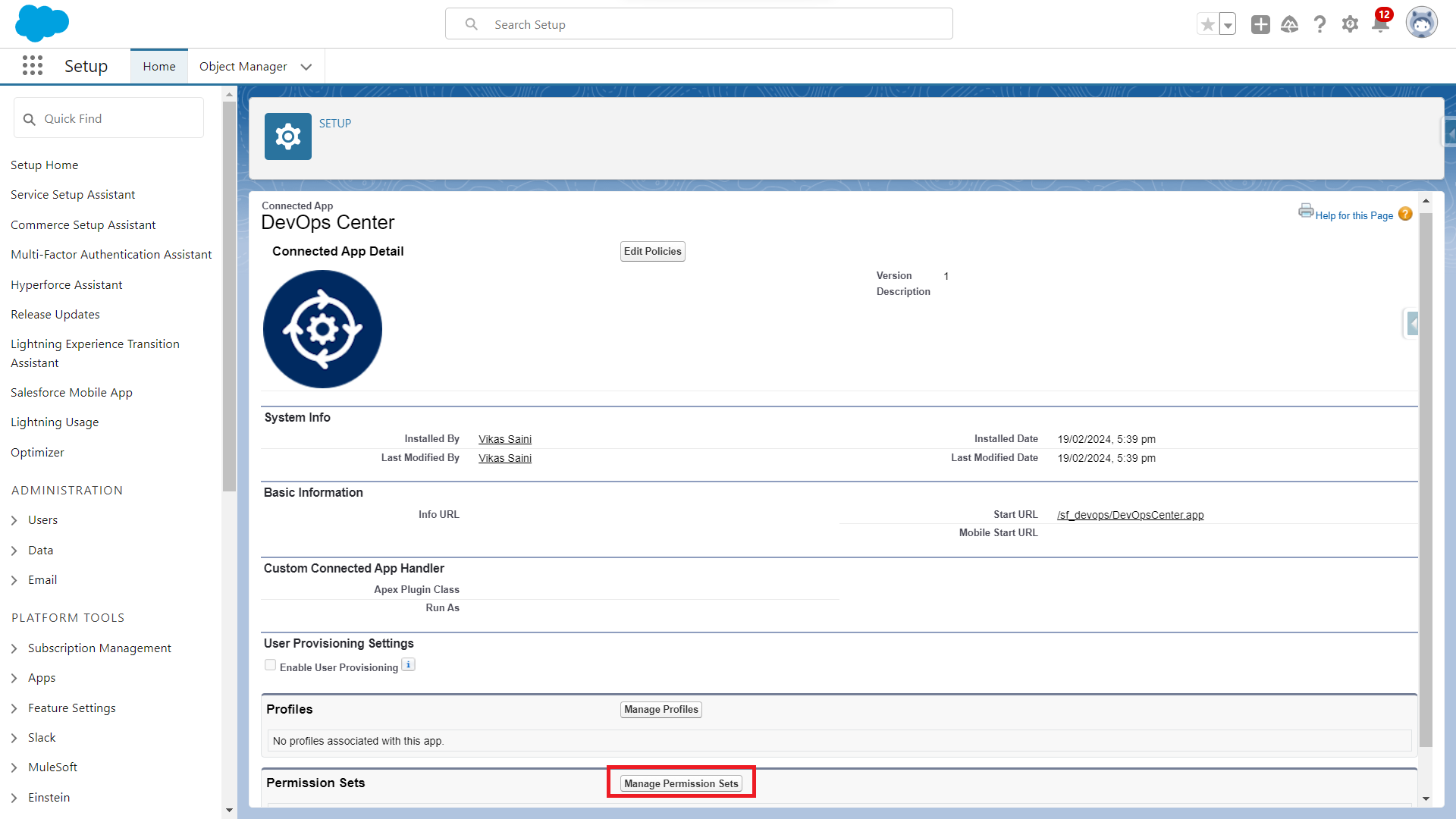
In Web App Settings, enter the Start URL: /sf\_devops/DevOpsCenter.app



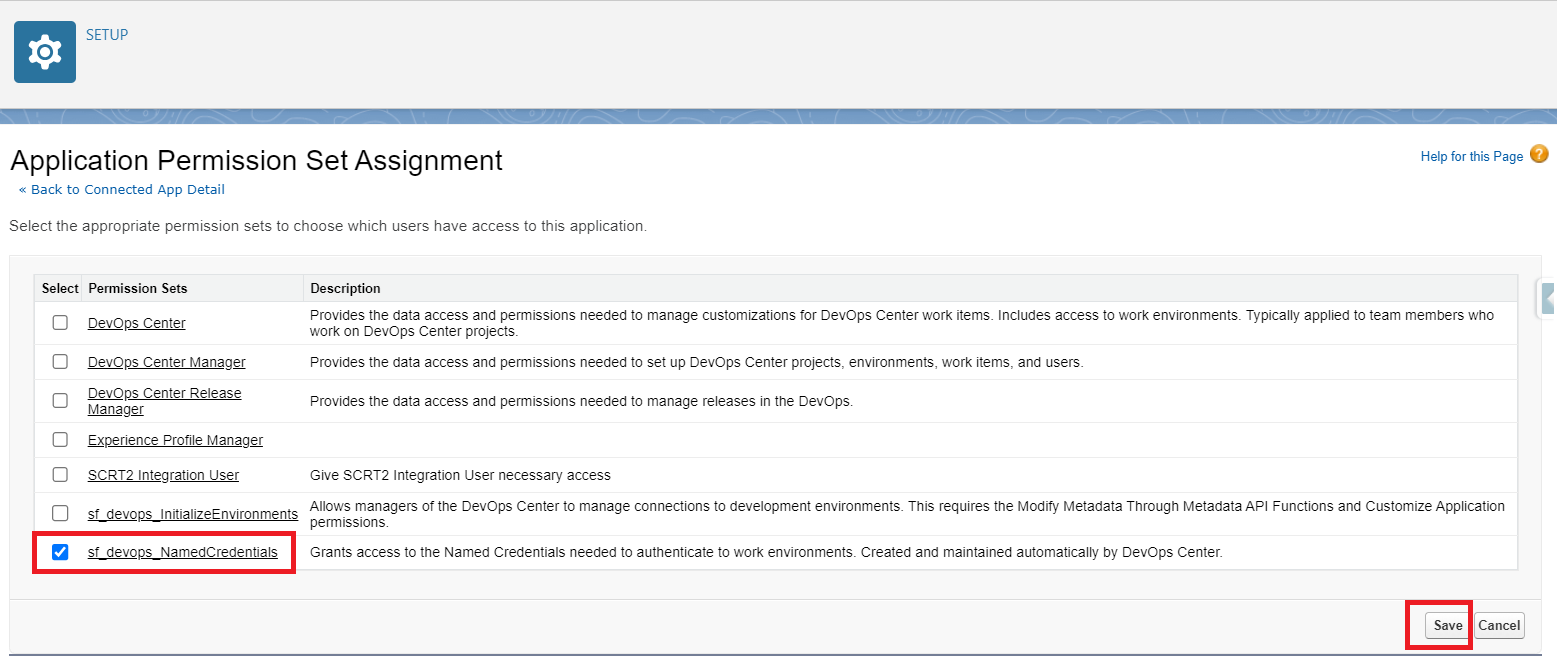
Step:4 Click on save Button.

Step:5 In Manage Connected Apps, click Manage.

Step:6. In the Permissions Sets section, click Manage Permission Sets.



Step:7 Select the permission set: “sf\_devops\_NamedCredentials”

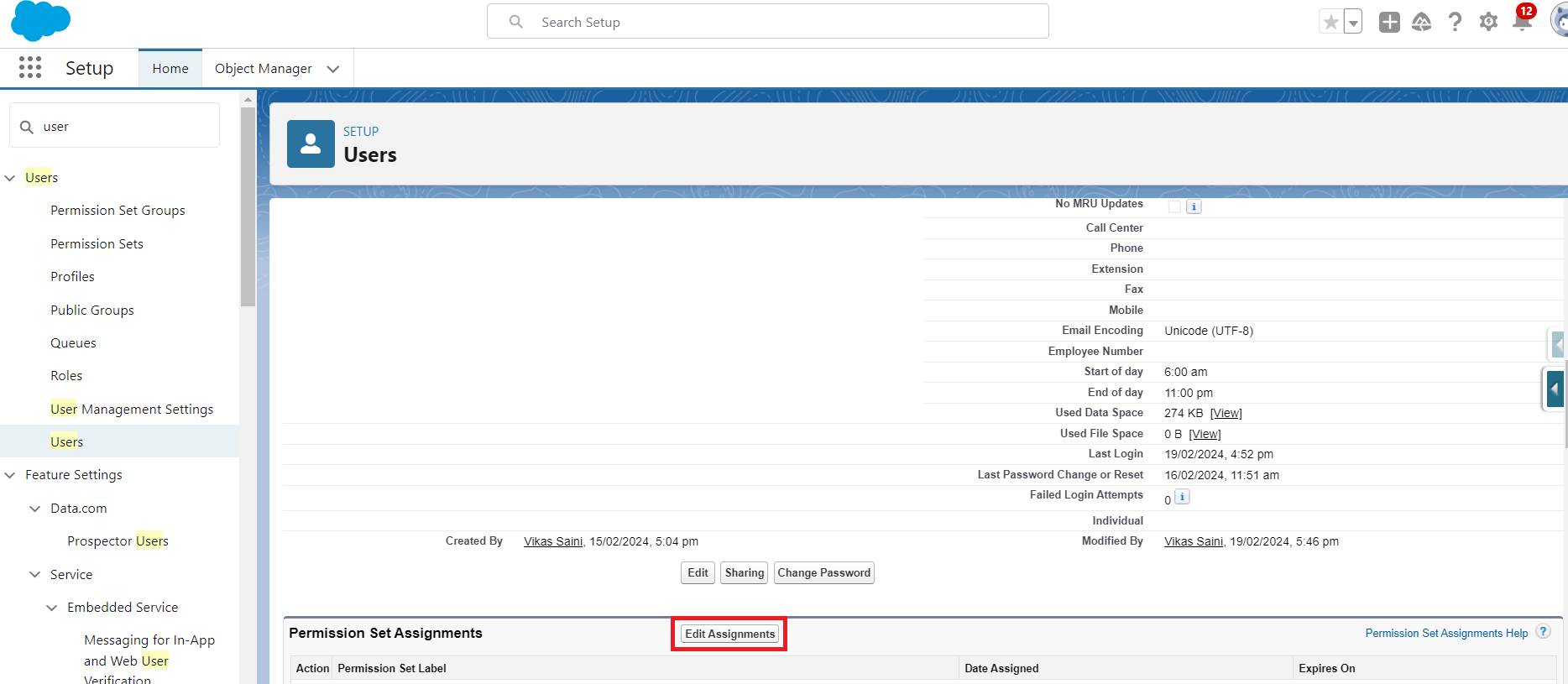


Step:8 click on save

**Assign permissions to use the Salesforce devOps center**

Step:1 From Setup, enter users in the Quick Find box, then select users.

Step:2 Click on the user you want to add permissions on and scroll down to “Permission Set Assignments” and click on Edit Assignments.



Step:3 Add the following permissions set to the user:

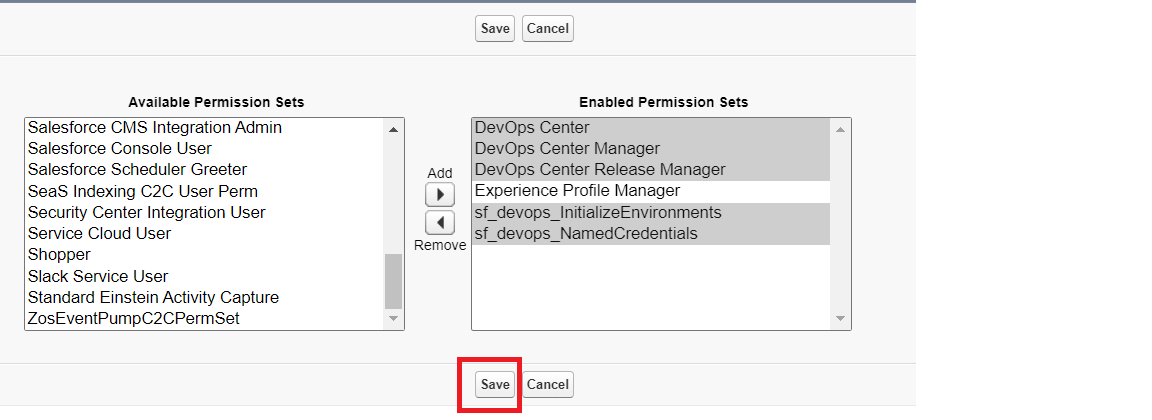
DevOps Center

DevOps Center Manager

DevOps Center Release Manager

sf\_devops\_InitializeEnvironments

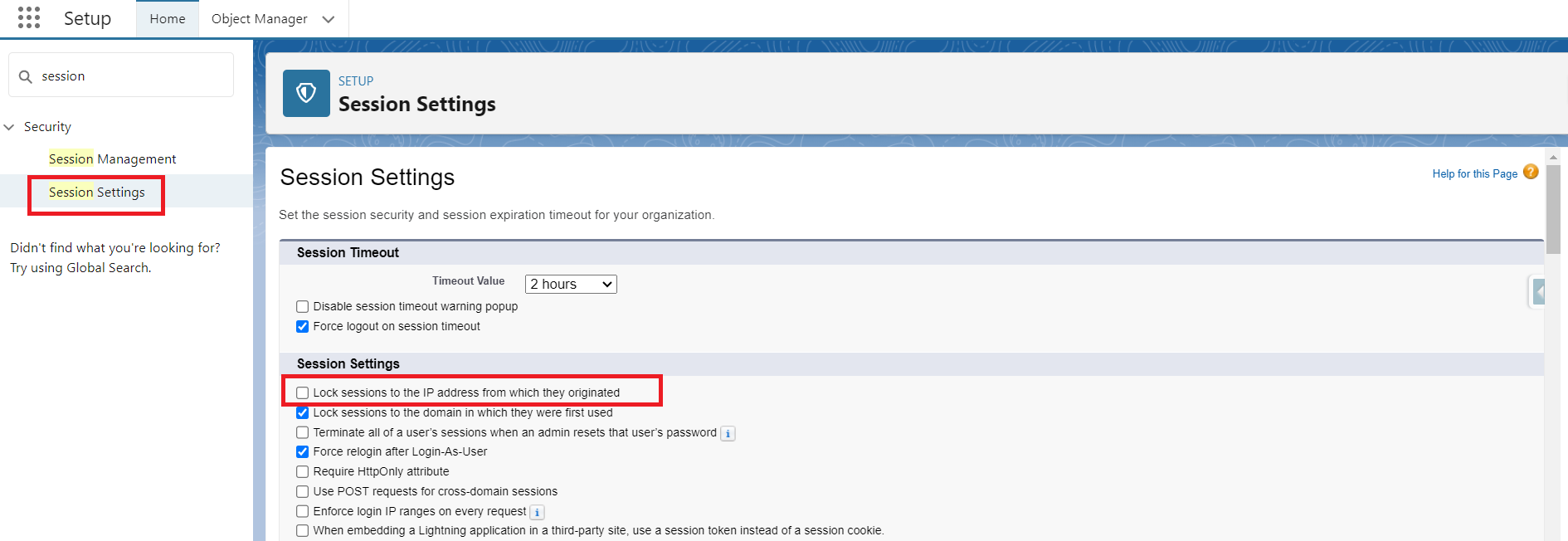
sf\_devops\_NamedCredentials



**For Session Setting**

Step:1 From Setup, enter Session Settings in the Quick Find box, then select Session Settings.

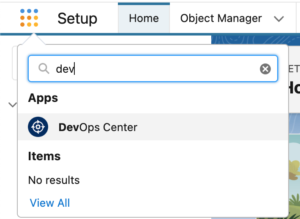
Step:2 Deselect “Lock sessions to the IP address from which they originated”.



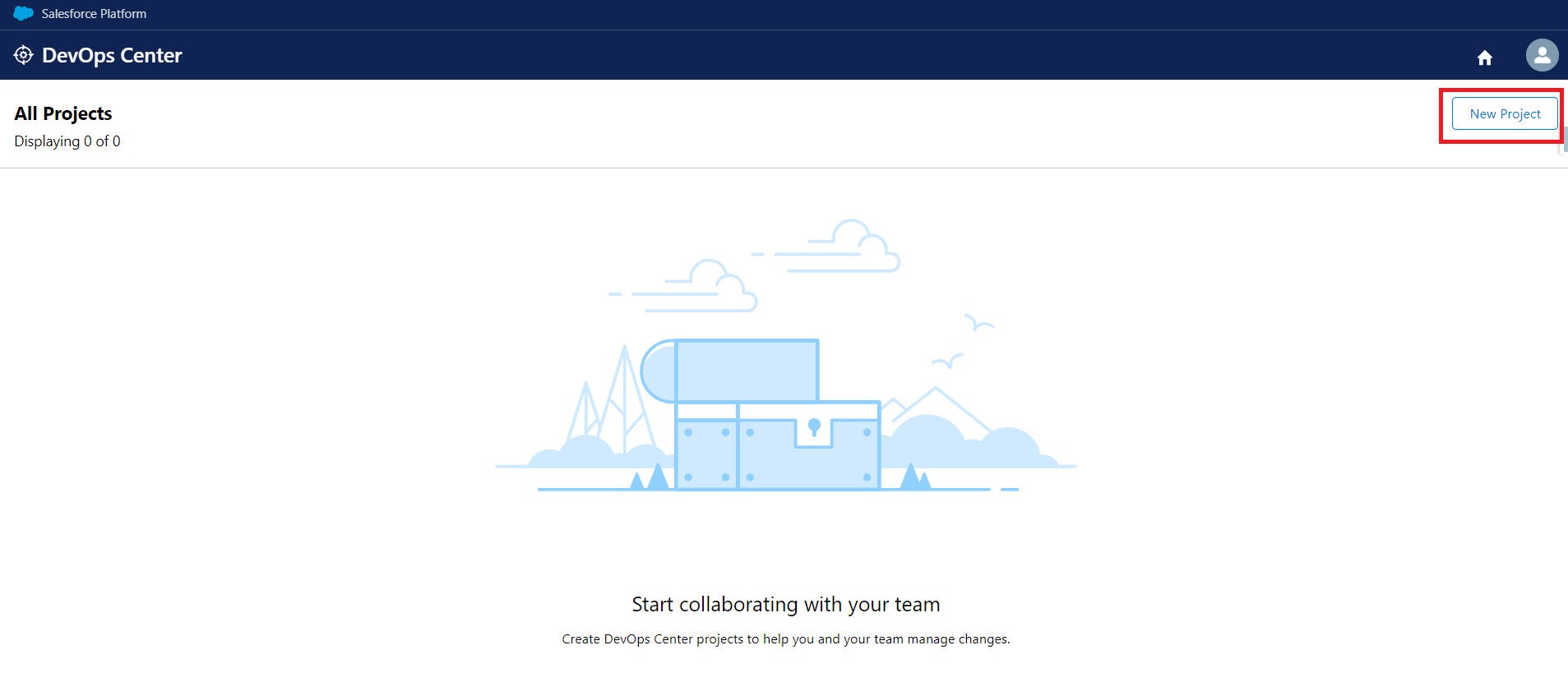
Step:3 Click on Save Button.

**How to open DevOps center**

Step:1 Click on App Launcher and search for DevOps Center and click on.



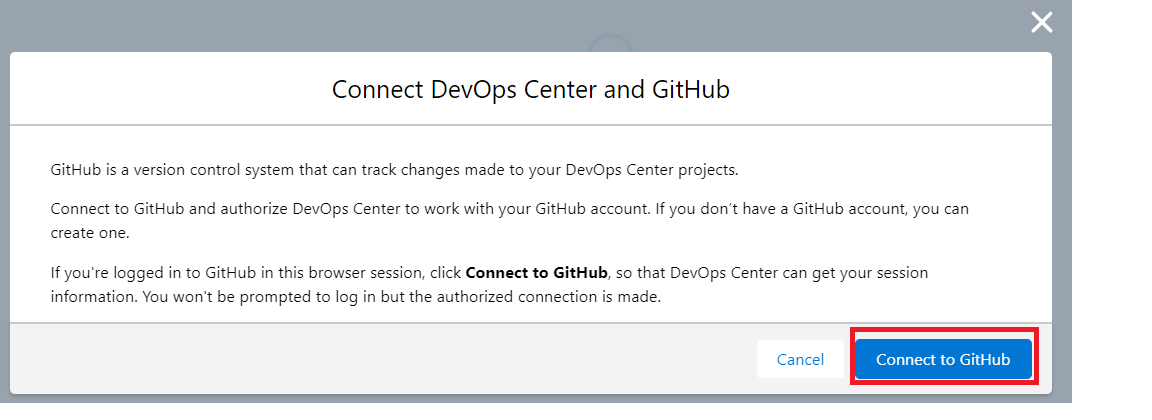
The DevOps Center app page will look like the screenshot below.



**How to configuration Devops Center**

Step:1 Click on New Project below screen will appear and click on “Connect to GitHub” button

Note: This screen will appear only once for github connection.

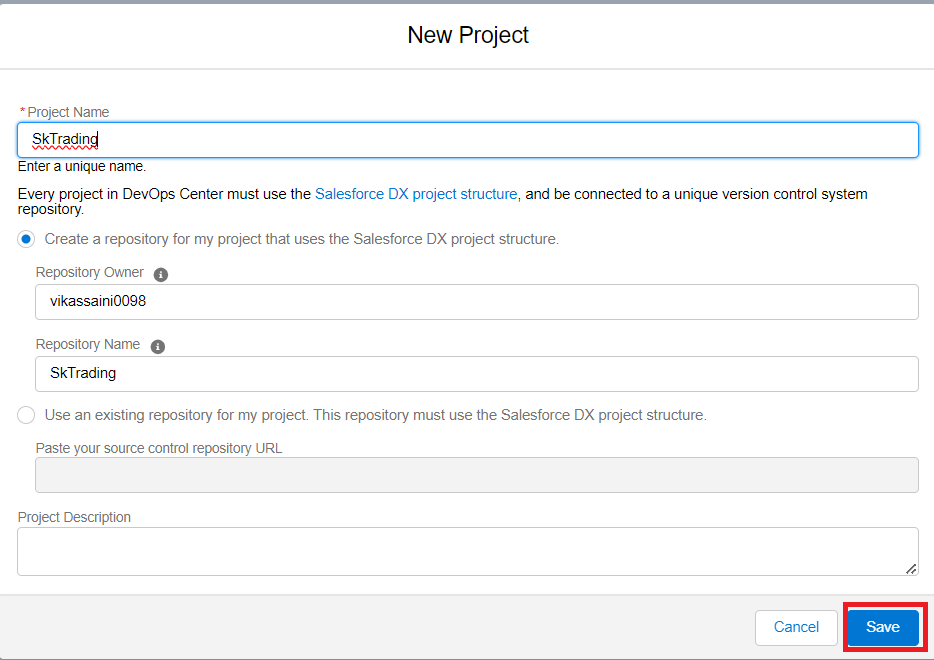


Step:2

Login GitHub with email and password.

Login successfully.

Step:3 again click on a new project.



Here two option available

1. Create new repository
2. Use existing repository

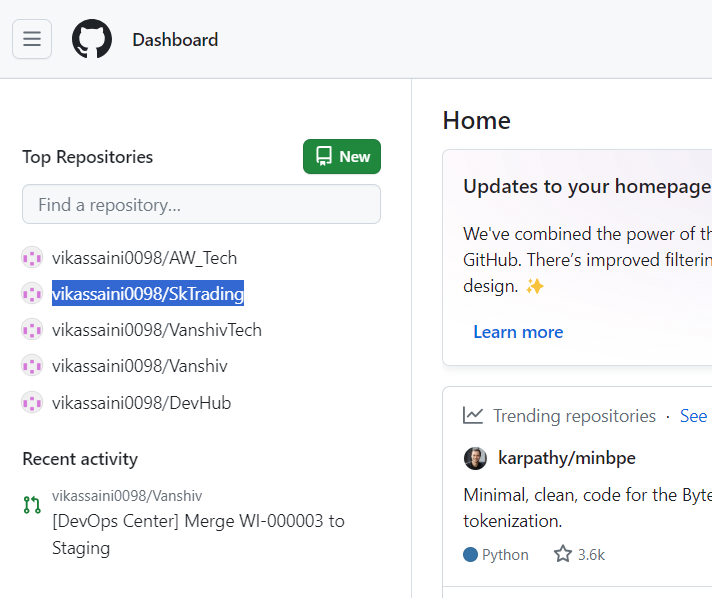
Here I am going to option a

Enter Project Name:SkTrading

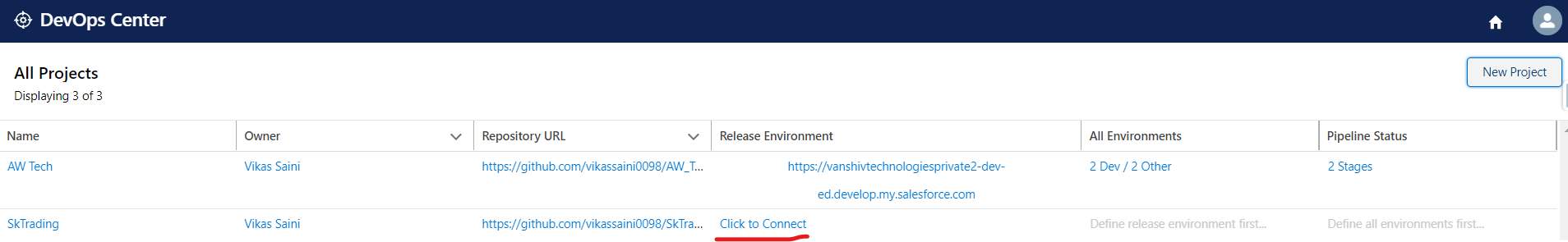
Project Description : as per your requirement

Repository name automatically appears When using option a

Step:4 Click on Save button and git repository automatically created with the same name of the project name.

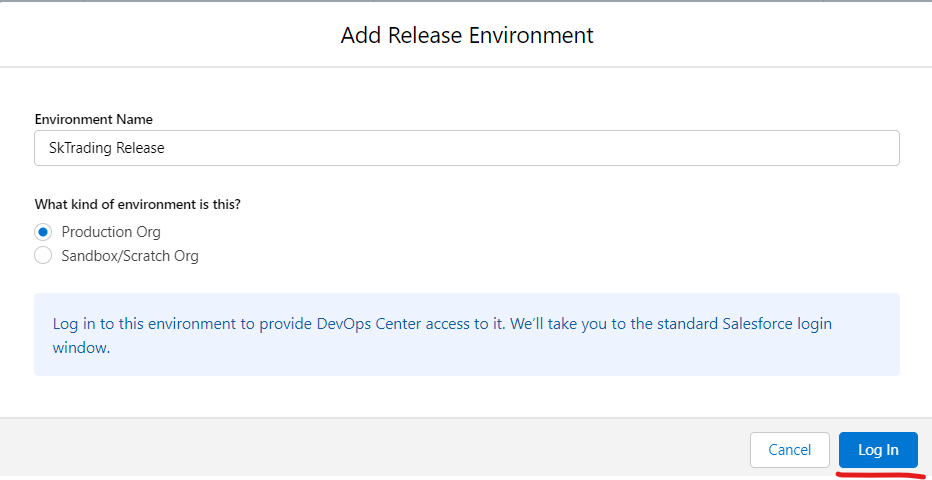


Step:5 after saving project below screen will appear

****

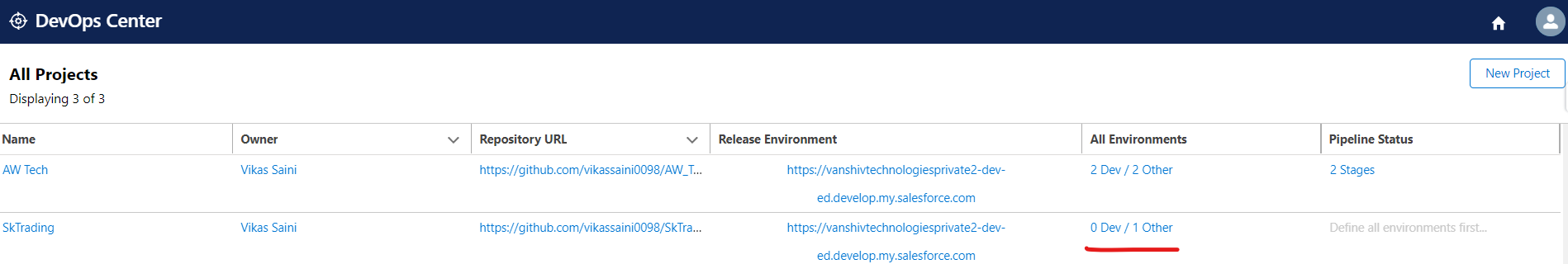
Step:6 Click on ‘Click to connect’ .

Here we are connecting our prod org -> click on login



It will redirect to our production org enter username & password of our production org.

Step:7 after successful login below screen will appear and now click to 0dev/1 other link for manage our pipeline

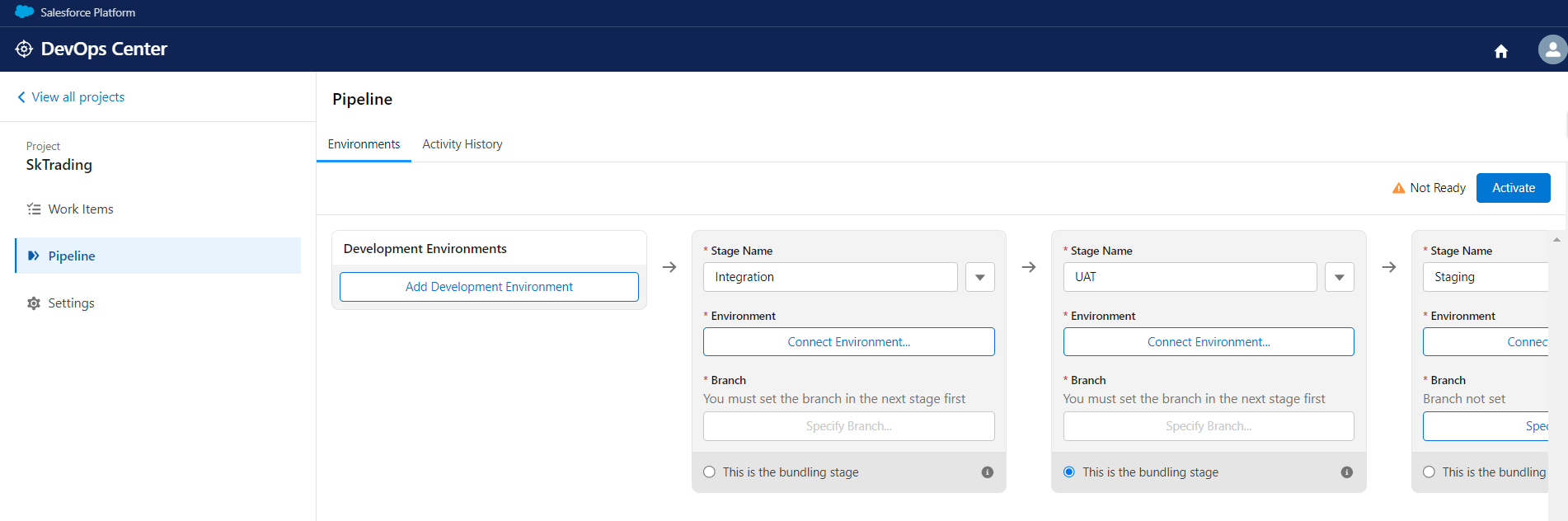


Step:8 after that below screen will appear

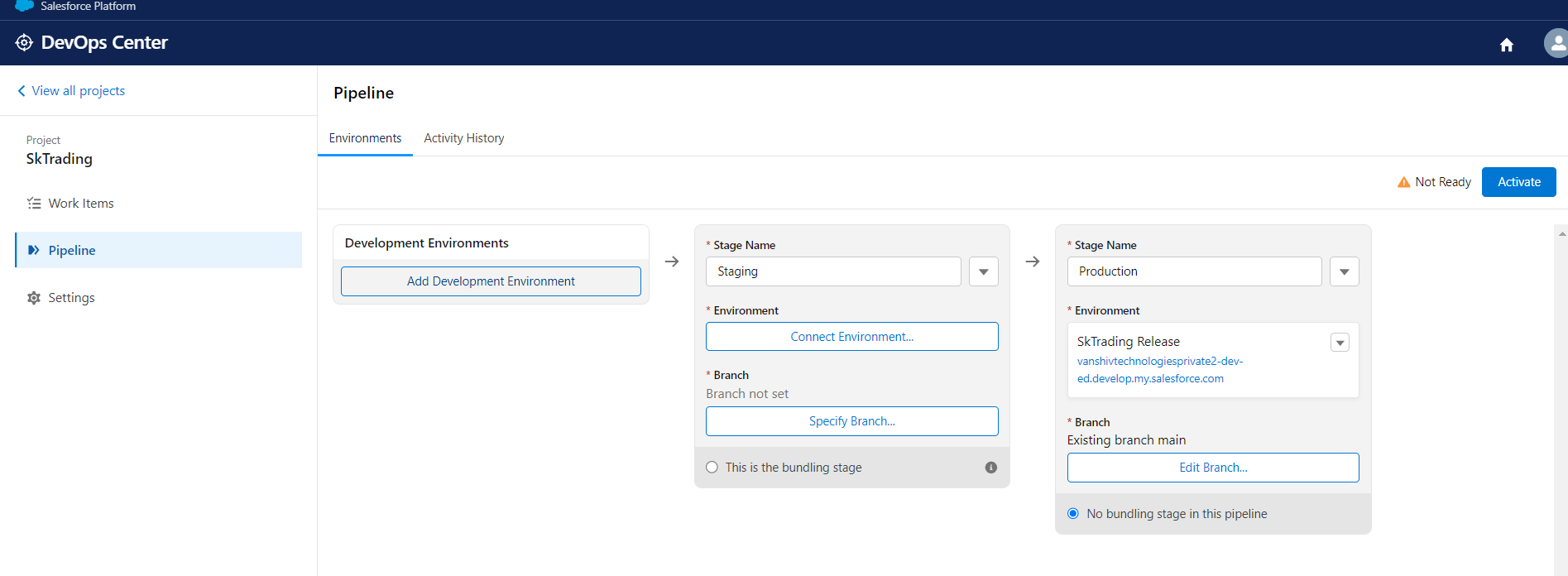
Here we can see various stages

But here I will use a 2dev org environment , one Staging and prod.

Delete Integration and UAT stage from arrow icon.



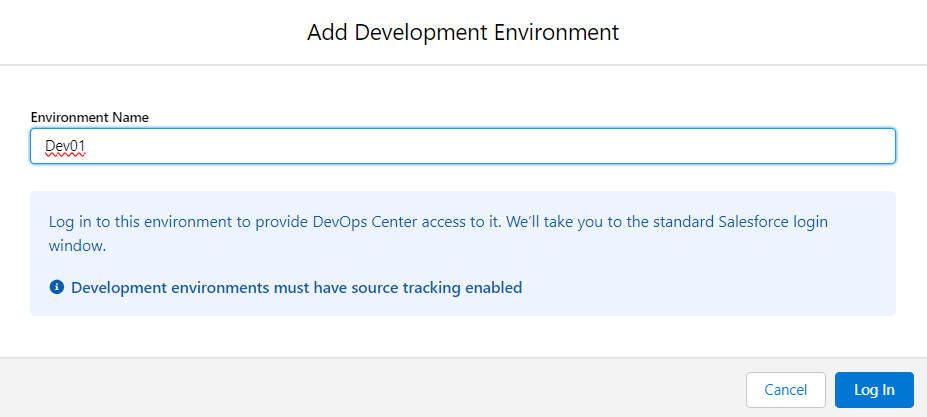
Step:9 Click on add development environment

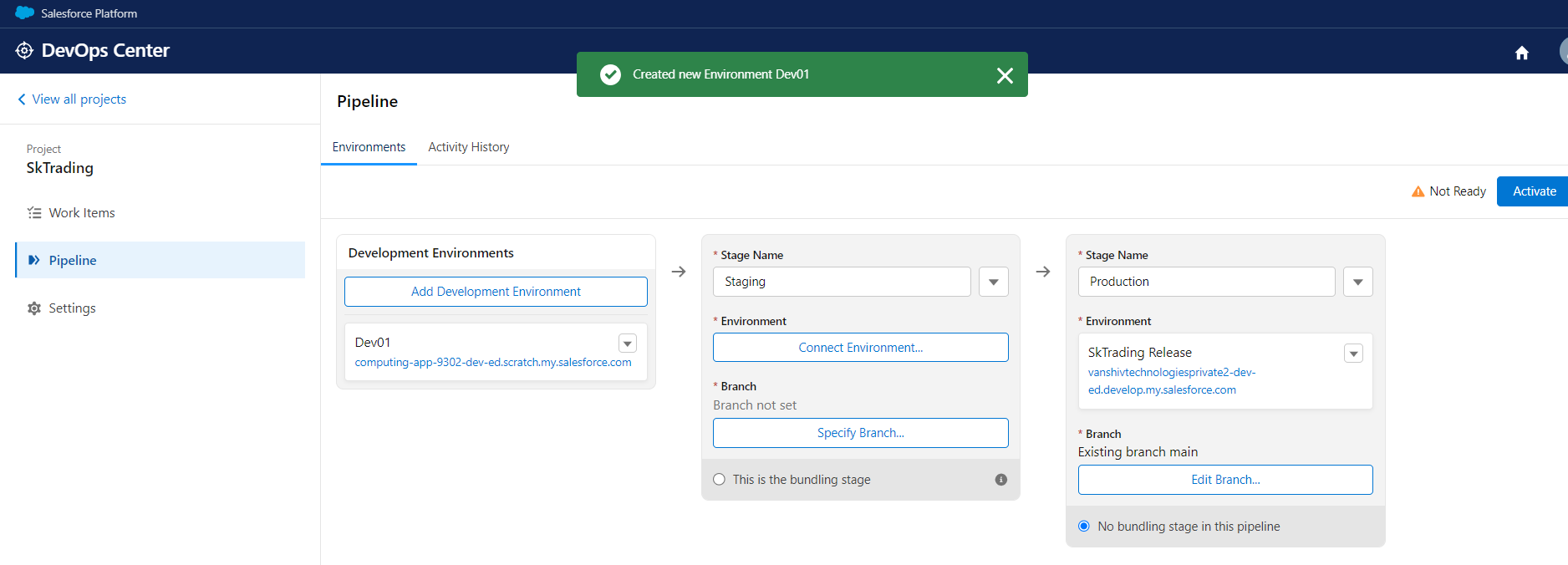


Here I will add dev1 and dev2 org

Environment Name: Dev01

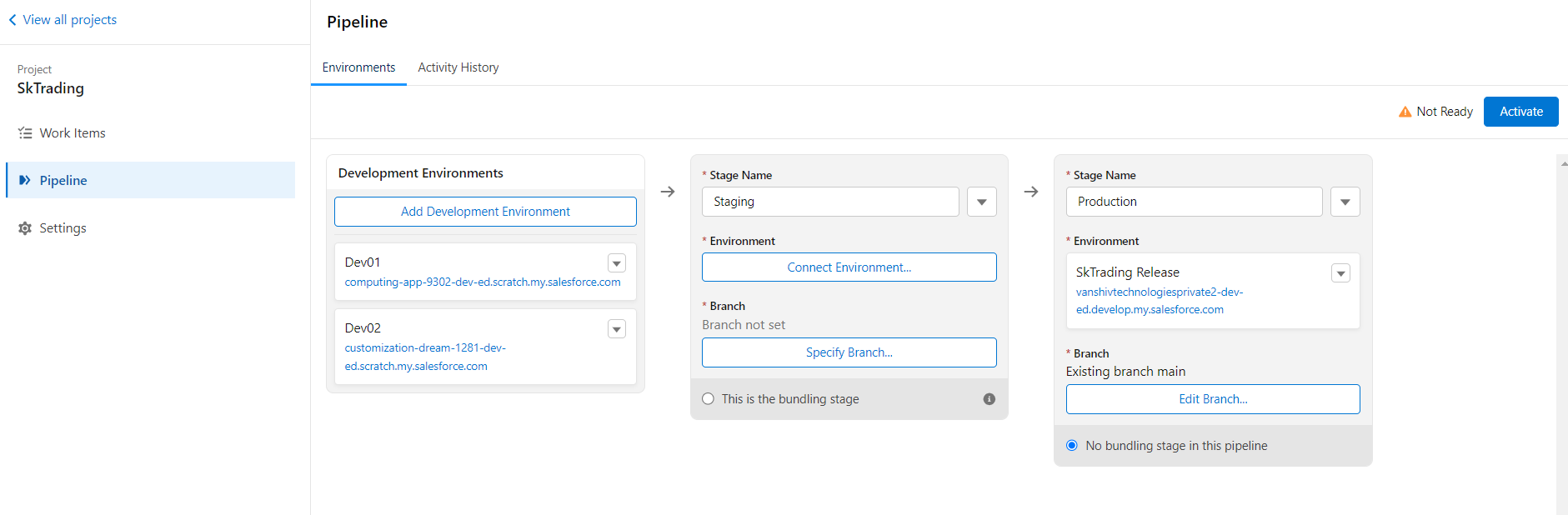
Click on login and it will redirect to dev1 sandbox enter username & password.





Follow above step for adding dev2

Environment Name: Dev02



Step:10 For staging stage click on connect environment . it will redirect to our staging org and enter username & password.

Step:11 Click on specify branch

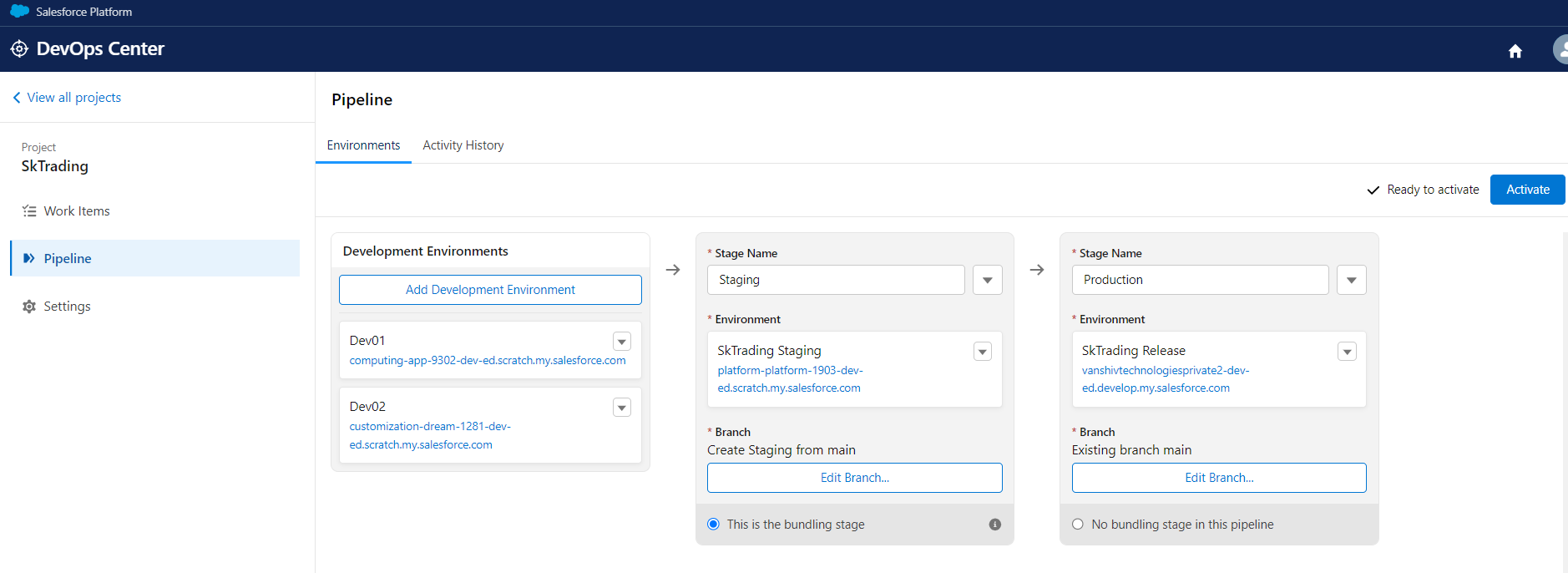
Here two option available

* 1. Create a branch for me from main for this stage
  2. Use an existing branch for this stage

Here i will use option (i)

Enter branch name:Staging

Step: 12 Click on save button.



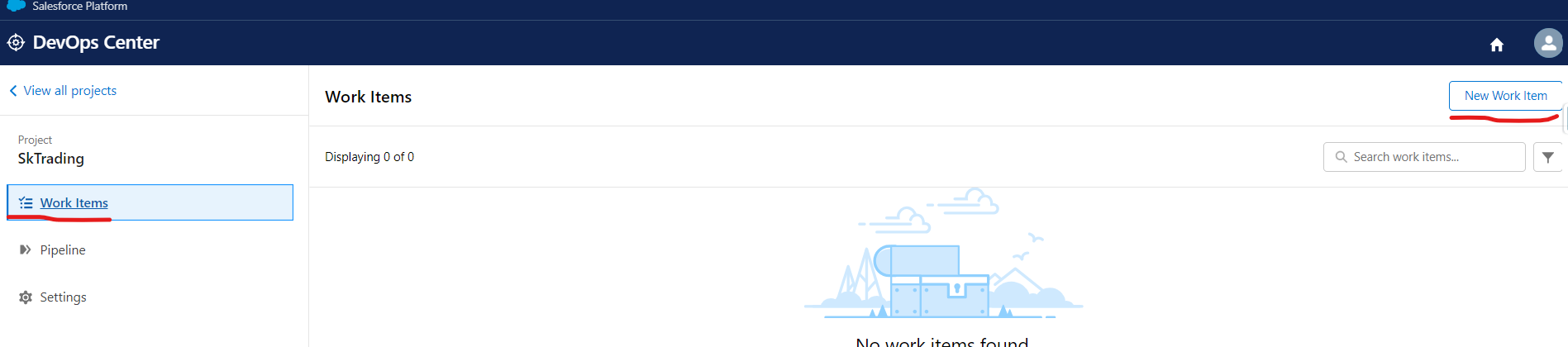
Our pipeline is ready to use.

Select This is the bundling stage radio button in the staging stage after that click on the Activate button.

After Activating a new staging branch will automatically be created in github.

**How to Create Work Item**

Step:1 Click on the Work items and click on the new work item button. Work item like a user story.

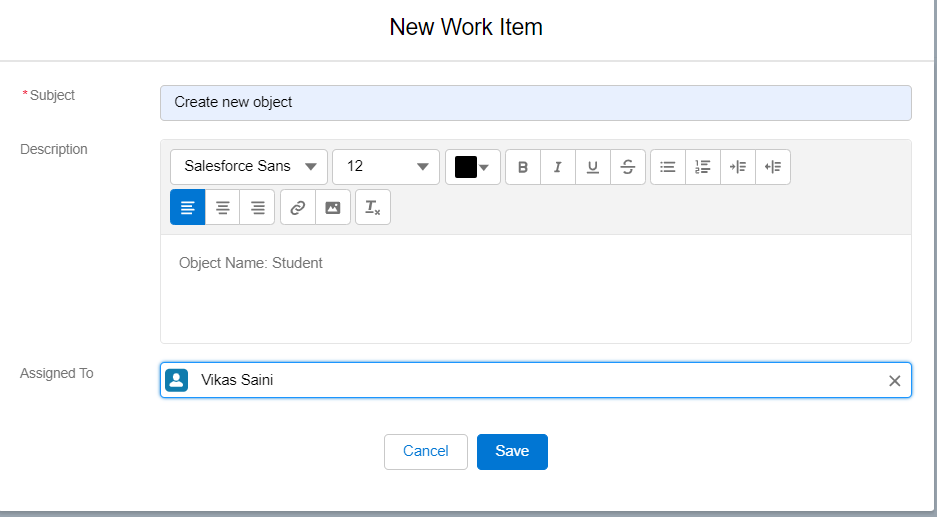


Step:2 below screen will appear

Enter subject: Create new object

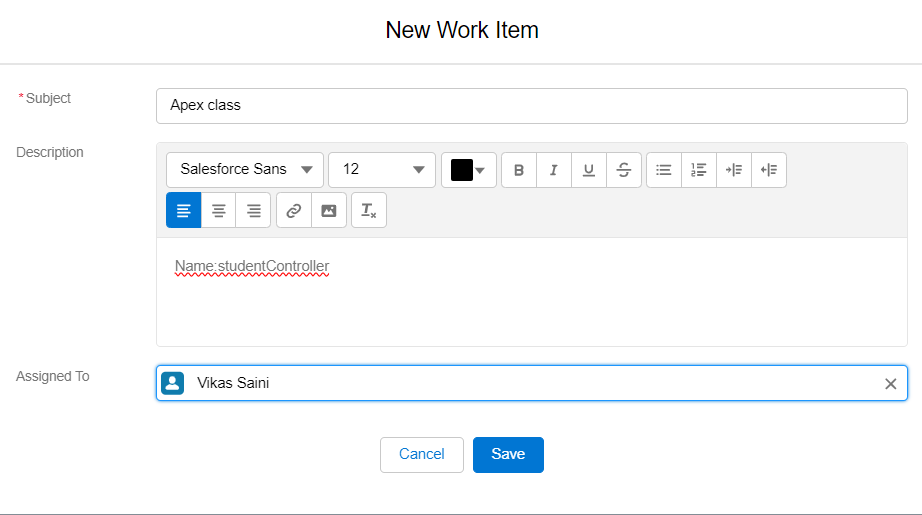
Enter description: as per requirement

Assigned to: user



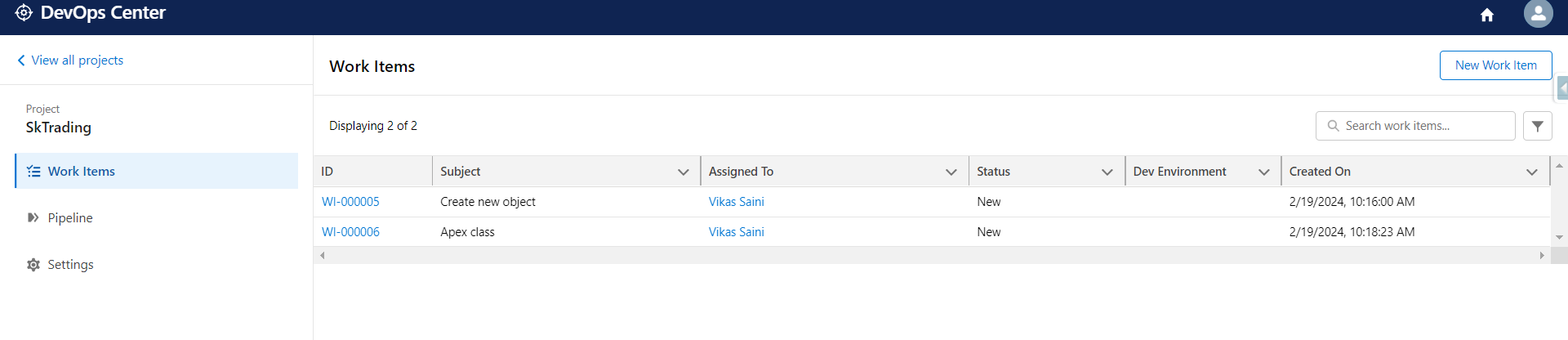
Step:3 Click on save button

Follow above steps 1 to 3 to create another work item.



Click on the save button.

Here we can see two work item.



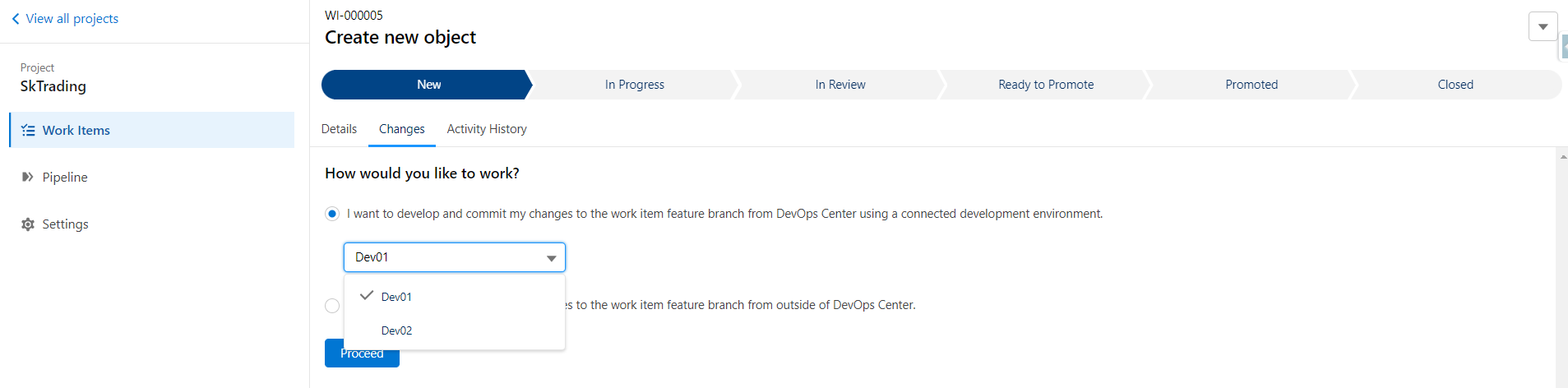
**How we can work on work item**

Step:1 Click on the first work item.

Here two option available

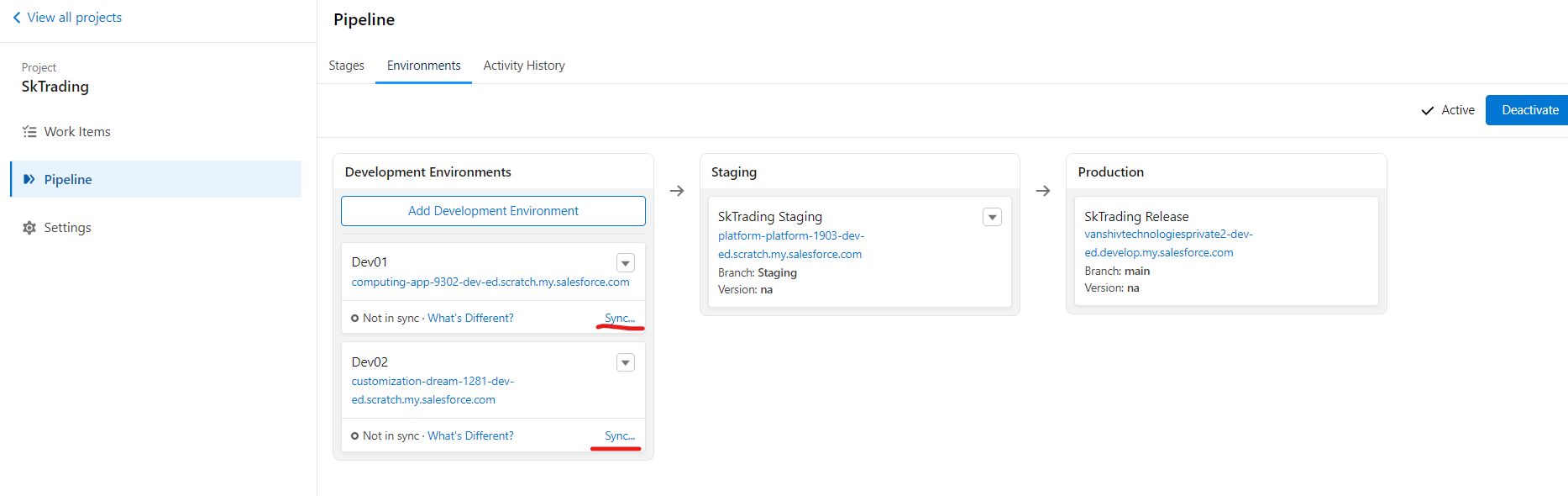
1. I want to develop and commit my changes to the work item feature branch from DevOps Center using a connected development environment
2. I want to develop and commit my changes to the work item feature branch from outside of DevOps Center

Here I will select option a

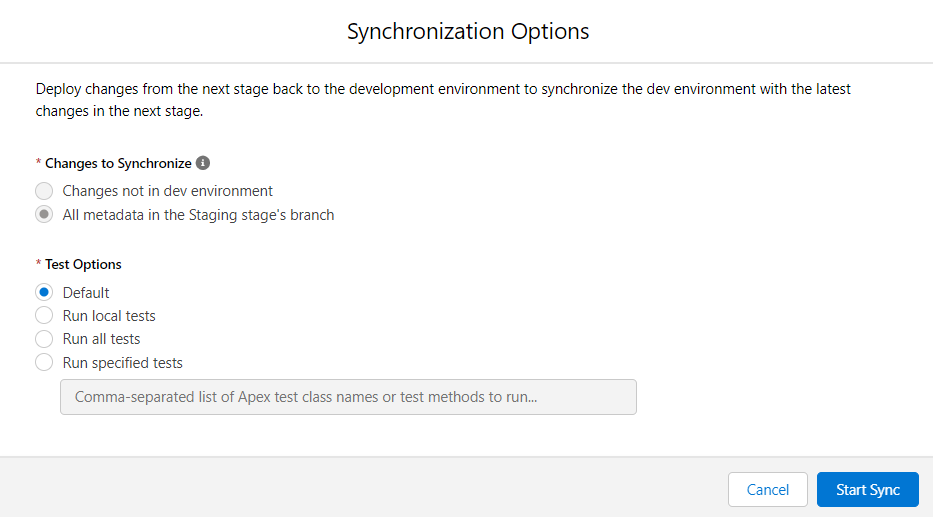


Step:2 Before proceed we need to synchronize dev01/dev02 from staging branch

So goto pipeline option



Step:3 Click start sync

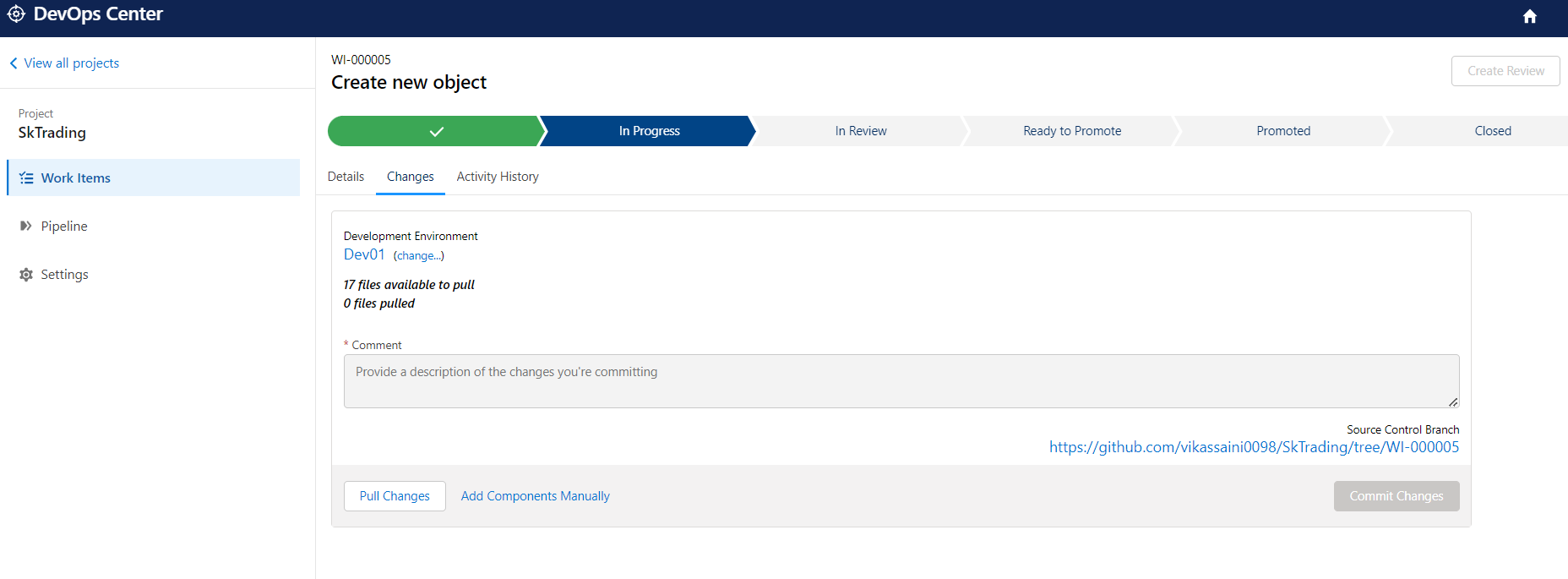


Step:4 Goto work items. And click on first work item

Select dev1 and click on the proceed button.

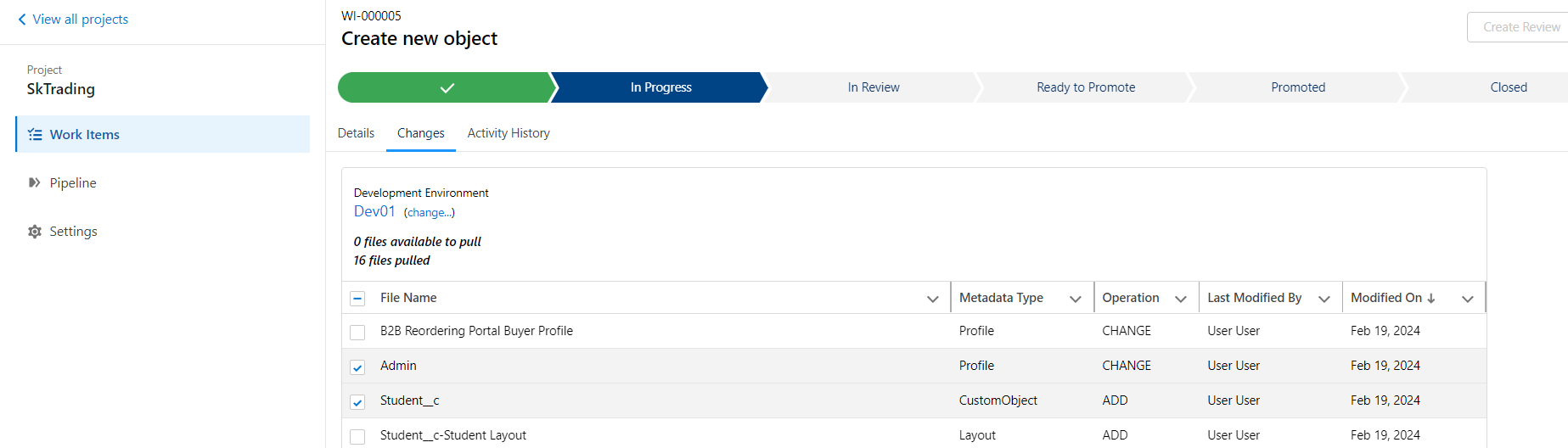
Feature branch will automatically create when we click on the proceed button.

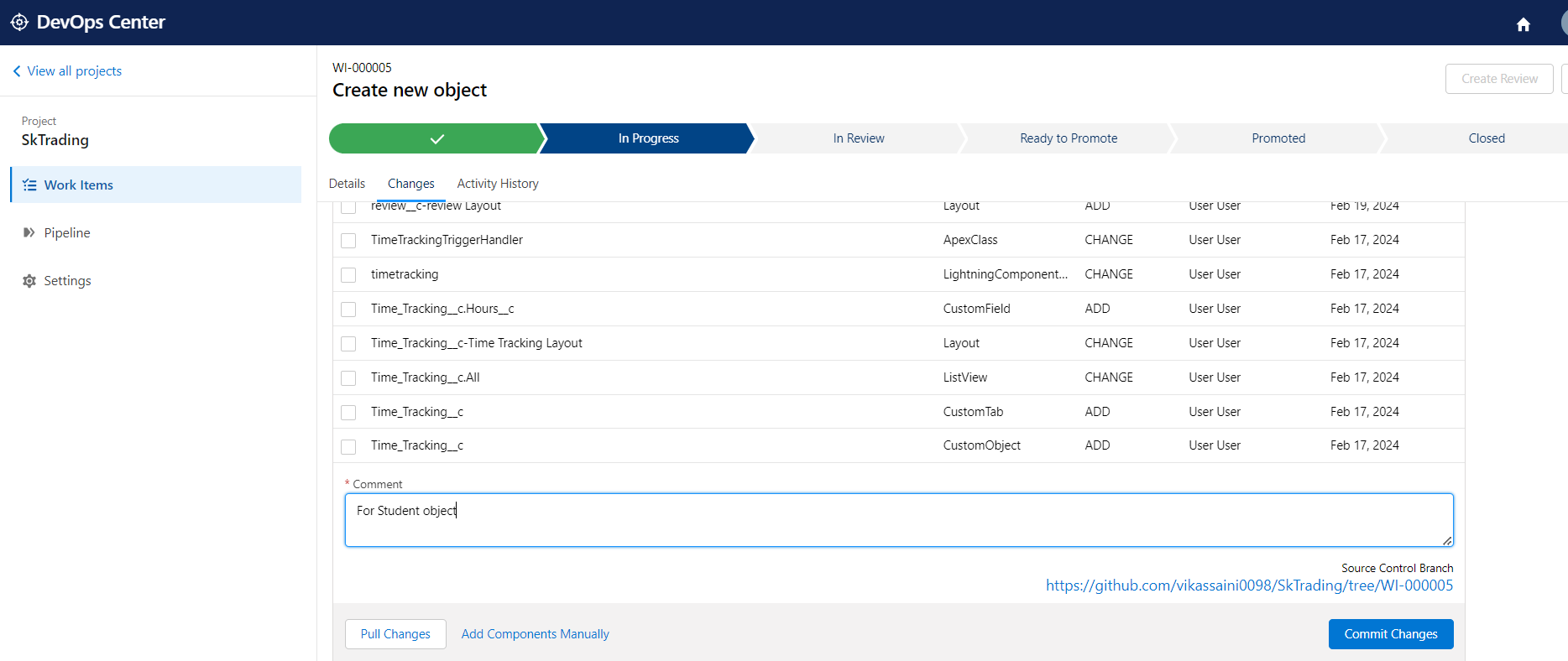
Step:5 Now click on dev1 .It will redirect to the dev1 sandbox and do work for the first work item.



Step:6 after work done click on pull changes button from bottom and

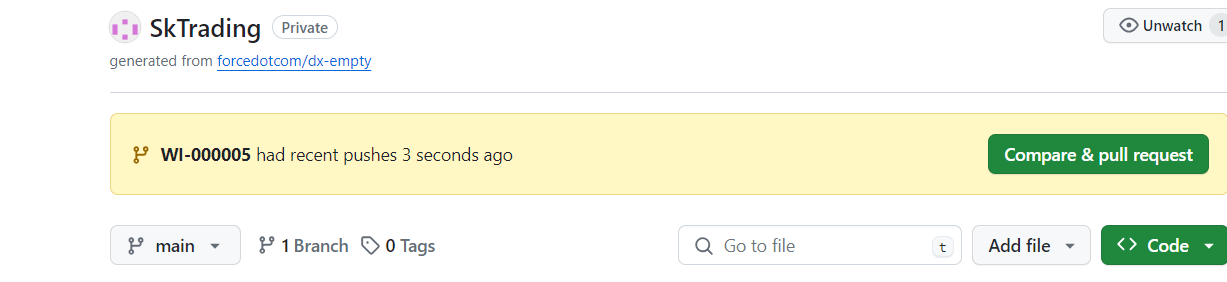
Select our changes , enter description related to work item





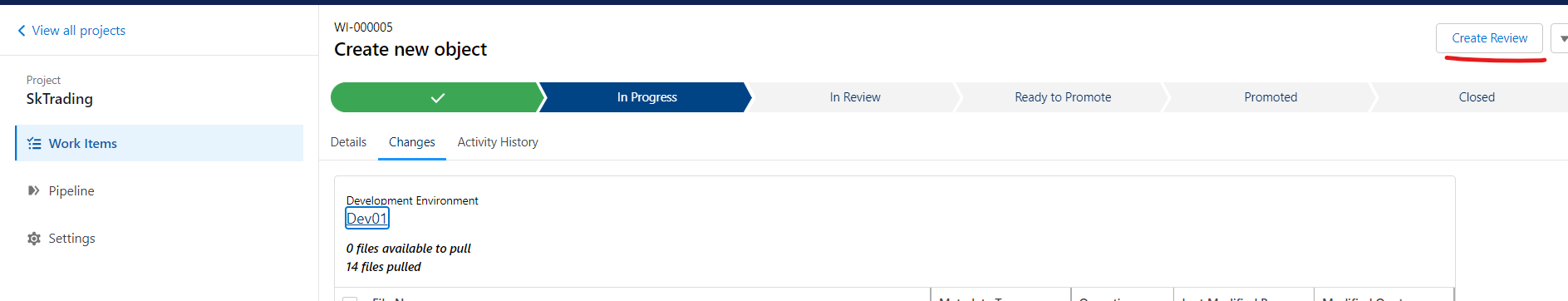
Step:7 Click on Commit changes button.

Step:8 Goto GitHub and see our commit here

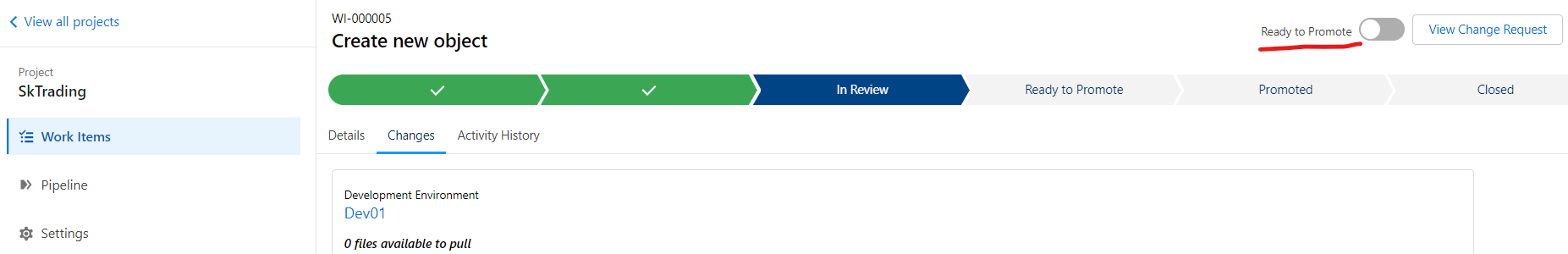


Step:9 Click on create review button

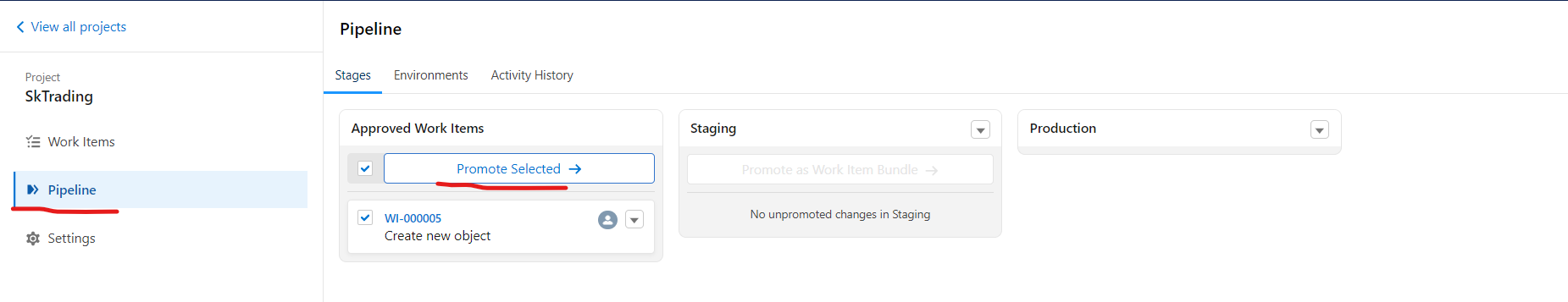
When we click on this button then PR will automatically be created.



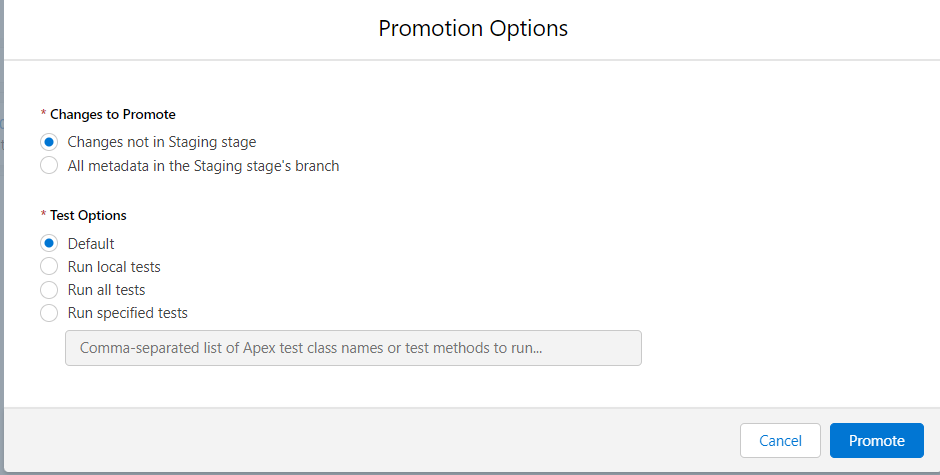
Step:10 after release manager check our pr then all changes are done then it will move to ready to promote button from right corner.



Step:11 Click on pipeline and select work item and click on promote Selected



Step:12 Click on Promote button



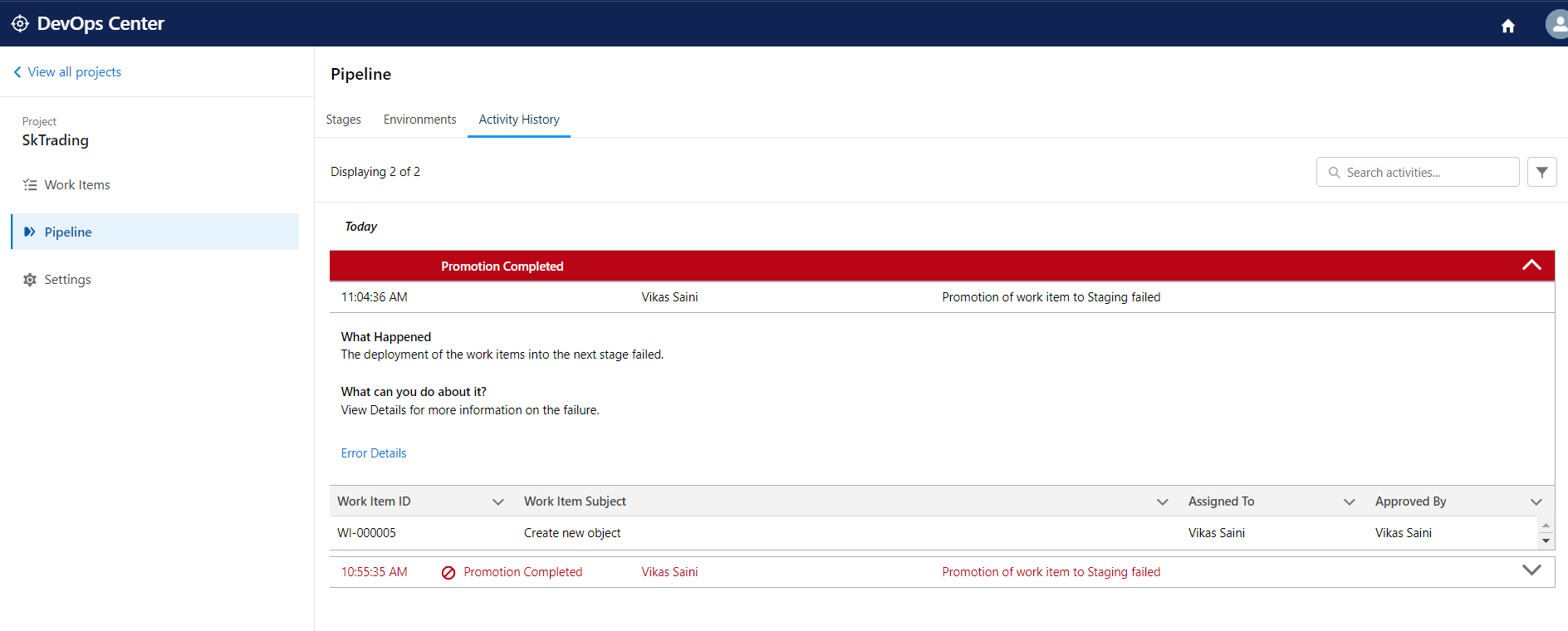
We we click on promote button two things will happen:

1. Merged the pull request in GitHub
2. Deploy code/changes in the org.

Note:Sometimes errors may occur or not.

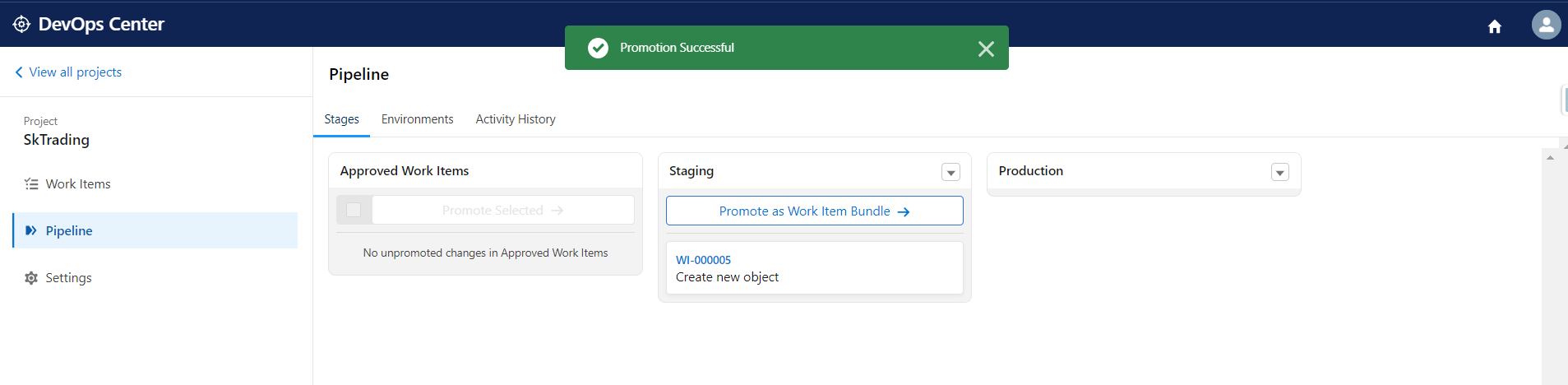
after clicking on the promotion button,maybe it’s showing an error.

* click on error details here we can see our error message.
* Fix them from dev01 org
* Back to DevOps Center
* Click on the work item first
* Click on pull changes
* select changes and enter description as per requirement
* commit them. It will resolve



* Goto pipeline
* select first work item
* click on the promote selected button.

After successfully promotion in staging,the feature branch will automatically deleted from github



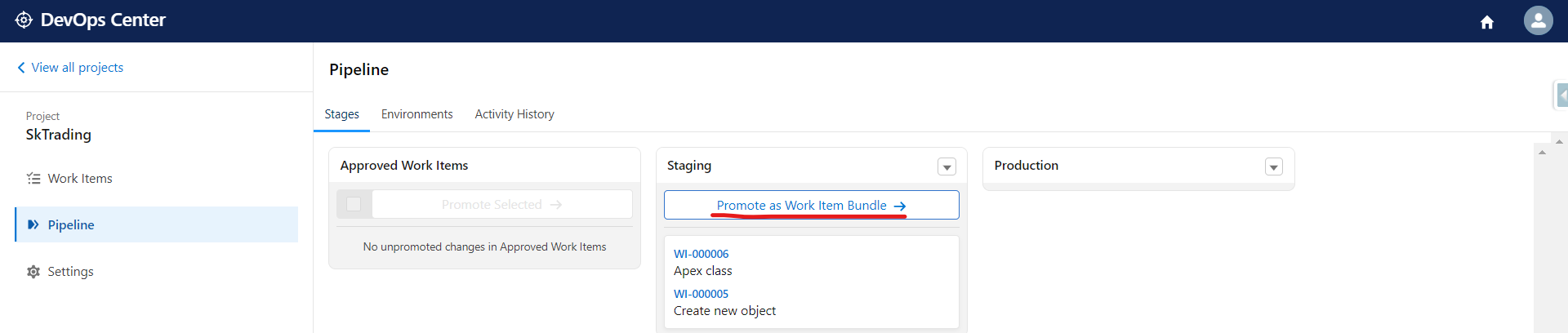
Now we will goto second work item from work item tab

We will use dev02 org with the above following steps 2 to 12 for second

Work item.

After that, we can see the screenshot below.

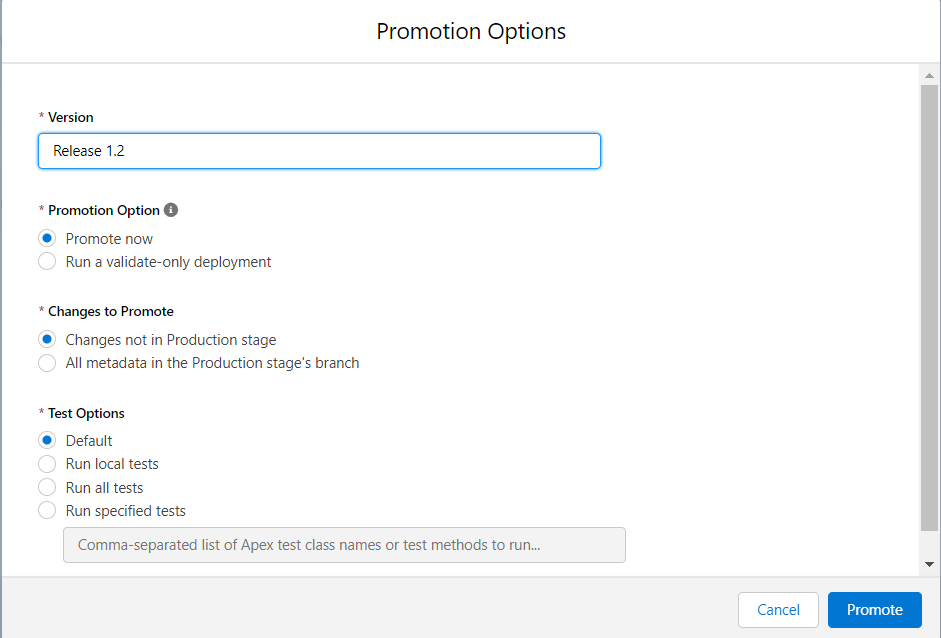
* In that we can see both work items in staging.
* Now we will click on “Promote as Work item Bundle”



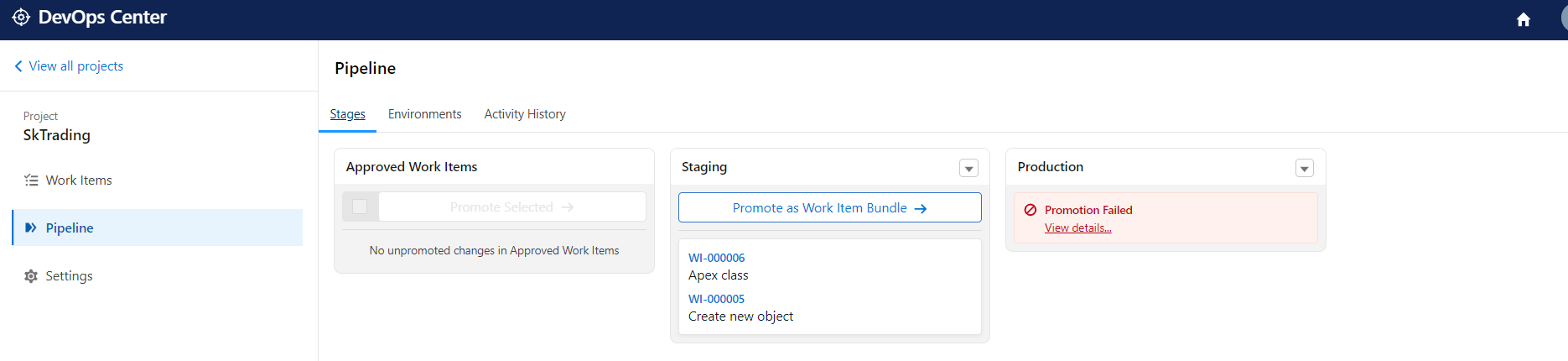
After Clicking below screen will appear

Enter Version name: as per requirement.

Here I am using version name like release 1.2

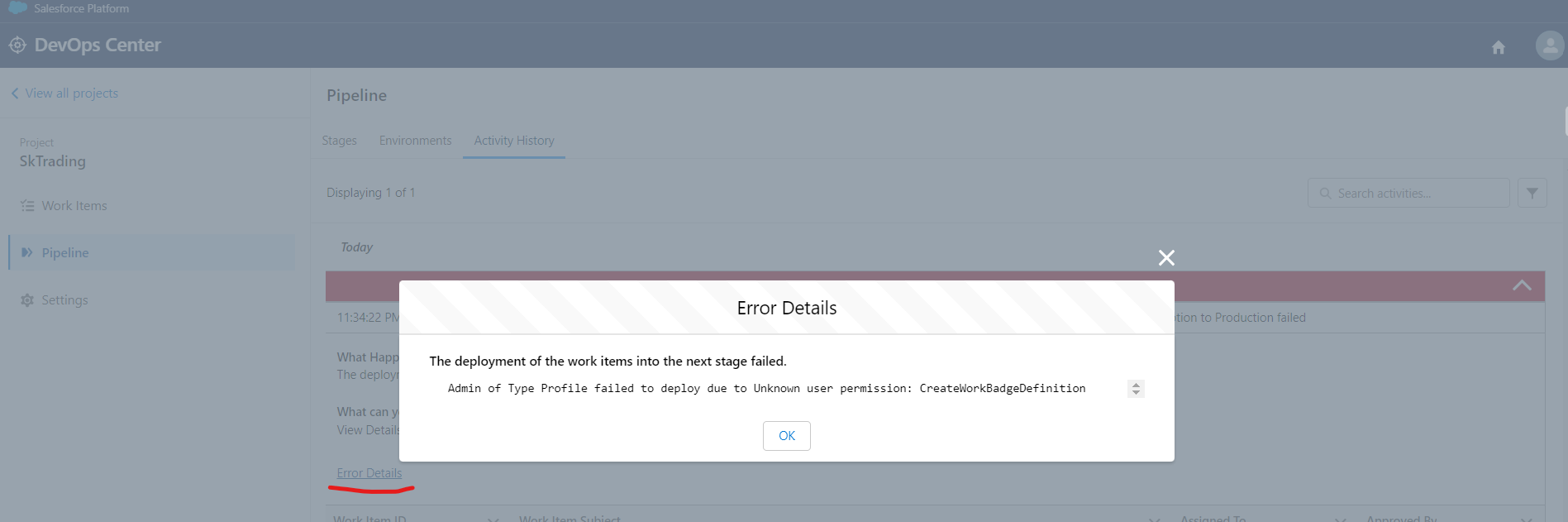


* After clicking promote button, error may occur or not
* But in our case error occur and promotion failed
* Click on “View details” to know to error



After below screen will appears

* Click on error Details to know error
* Here we can see user permission errors.



Now we will create another new work item related to error from the work item tab and work on.

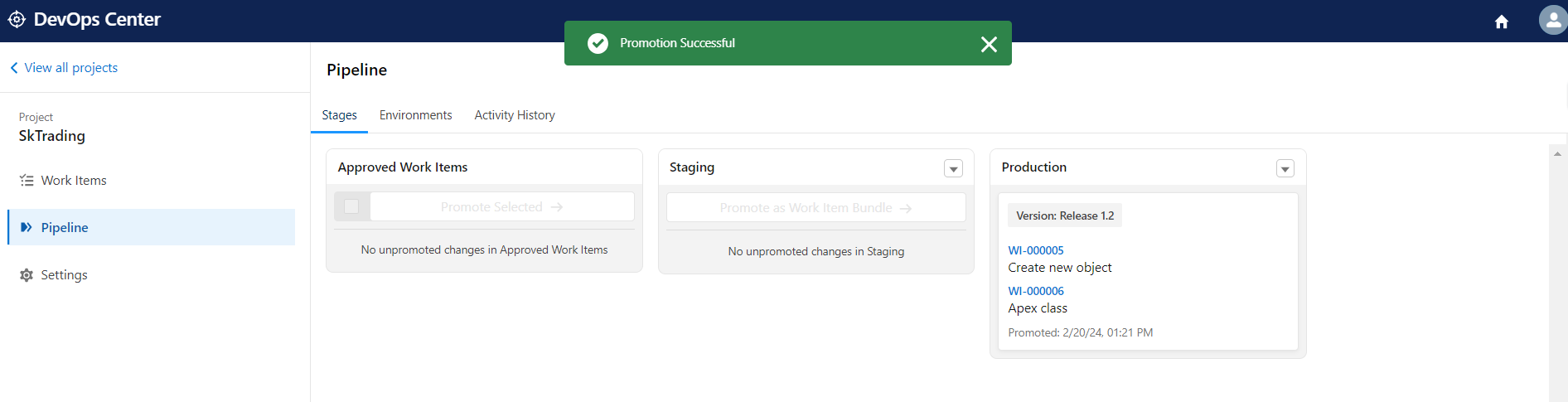
Follow all the above steps to complete a work item.

* Again we will click on “Promote as Work item Bundle" and enter the version name.

After clicking button two things will happen:

1. Merged the pull request in GitHub
2. Deploy code/changes in the org.

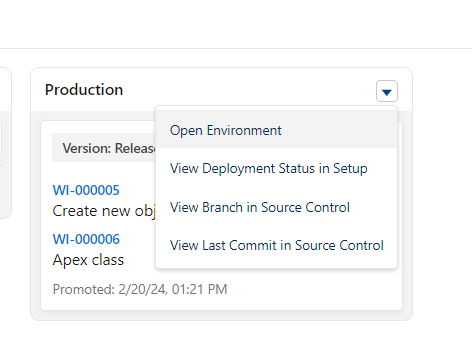
Now here we can see promotion successfully done and deployed to production.



Click on arrow button from right corner side.Below image will appear

Click on Open Environment it will redirect to our production org.

From here we can check our deployment work items.



**Thanks**