Main Steps for Setting Up the CI/CD Pipelines

Last Modified on 11/14/2022 12:36 pm EST

Following is a checklist of the main steps required to set up and execute CI/CD pipelines. Setting up CI/CD deployment pipelines requires the collaboration of a few people in the organization. Not all the required steps are performed by the CI/CD Administrator. Some need to be performed by the IT Administrator or the Project Tech Lead.

> **NOTE**
> The roles described below follow the structure of the Genpact Orchestration Professional Services group. Your organization may be structured differently.

Before you begin, make sure that:

- You have access to the server. *See step 1*.
- You have access to the "Customizations" Repo. This Repo includes the installation files for the CICD tool. *See step 2*.

Main procedure	Steps (performed by)
Set up the CICD tool	1. Create agents on the servers. **(IT and CI/CD Admin)** 2. Import the “Customizations” Repo from “Cloud DevOps IT Team” TFS project. (**CI/CD Admin)** 3. Import all relevant Task Groups for the whole process. (**CI/CD Admin)** 4. Import the “Standalone Pipeline” to install the CICD tool on the servers. (**CI/CD Admin)** 5. Configure the “Standalone Pipeline” and fill in the relevant values in its variables. (**CI/CD Admin and Project Tech Lead)** 6. Run the “Standalone Pipeline” on all relevant servers and make sure it’s finished successfully. (**CI/CD Admin and Project Tech Lead)**
Set up the workflows pipelines	7. Create Repos for the workflow pipelines: Deployment Packages and TFVC Project Repo. (**CI/CD Admin)** 8. Import two workflow pipelines: “Unit Package” and “Processes Release”. (**CI/CD Admin)** 9. Configure the workflow pipelines and fill in the relevant values in the variables. (**CI/CD Admin and Project Tech Lead)**

Main procedure	Steps (performed by)
Commit a workflow to Azure DevOps	10. Create a workflow in “Dev” environment in order to deploy it to “Test”. (**Project Tech Lead and CI/CD Admin)** 11. Create a TFS item that related to the workflow change. (**Project Tech Lead)** 12. Commit the relevant workflow to Azure DevOps and attach it to the created TFS Item. (**Project Tech Lead and CI/CD Admin)**
Package the workflow and deploy it to the target environment	13. Run the workflow pipelines to pack the created workflow from “Dev” and deploy it to “Test”. First run the “Unit Package” pipeline. **(Project Tech Lead and CI/CD Admin)** 14. Verify that the relevant package has been added to the “Deployment Packages” Repo. **(Project Tech Lead and CI/CD Admin)** 15. Run the “Processes Release” pipeline to deploy the package to the relevant environments. **(Project Tech Lead and CI/CD Admin)** 16. Check that indeed the process finished successfully and run a few tests again. For example, make a change in the same workflow and try to deploy the change as well. **(Project Tech Lead and CI/CD Admin)**
Set up the customizations pipeline	17. Create a GIT Repo with the project name and the folders’ structure (Shared Resources). **(Project Tech Lead and CI/CD Admin)** 18. Upload a file that we wish to deploy. **(Project Tech Lead and CI/CD Admin)** 19. Run the “Release Shared Resources” pipeline. **(Project Tech Lead and CI/CD Admin)** 20. Check that the file has been deployed successfully and that you see the change in the application. **(Project Tech Lead and CI/CD Admin)**

Check out these demos for more details on setting up the CI/CD pipelines.

1. CI/CD installation process (part 1)
2. CI/CD installation process (part 2)
3. Customizations pipeline