**CI-CD Workflow**

**CI/CD pipeline:**  
The primary goal of a CI/CD pipeline is to automate the software development lifecycle. It will reduce manual tasks for the development team which in turn reduces the number of human errors while delivering fast results. All these contribute towards the increased productivity of the team.

**Continuous Integration:**  
Continuous integration is a development practice where development teams make small, frequent changes to code. An automated build verifies the code each time developers commit their changes into the remote repository. As a result, development teams can detect problems early and fix them.

**Continuous Delivery:**  
Continuous Delivery is the next part of the pipeline which automates the deployment to the testing environment when the CI step is successful. The core idea is to deliver the code to QA environment so that it can be constantly and regularly validated.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Feature Branch Workflow** | | | | | |
|
| **Branch** | **Stage 1** | **Stage 2** | **stage 3** | **stage 4** |  |
| feature/\* | Git Checkout | Code Analysis(sonar) | Code Build | Deploy to Dev Env |  |
| **Develop Branch Workflow** | | | | | |
|
| **Branch** | **Stage 1** | **Stage 2** | **stage 3** | **stage 4** | **stage 5** |
| develop | Git Checkout | Code Analysis(sonar) | Code Build | Artifacts | Deploy to Test Env |
| **Master Branch Workflow** | | | | | |
|
| **Branch** | **Stage 1** | **Stage 2** | **stage 3** | **stage 4** | **stage 5** |
| master/\* | Git Checkout | Code Analysis(sonar) | Code Build | Artifacts | Deploy to prod Env |

**Requirement:**

* **Ubuntu 20.04 server with SSH access for Jenkins setup.**
* **Java 11**
* **Jenkins (Latest Version: Jenkins 2.375.4).**
* **SonarQube testing tools.**
* **GitHub with Admin rights.**

