

Milestone 4: Continuous Integration and Continuous Delivery (Jenkins)

<https://github.com/abduh3131/SQLAB4>

<https://drive.google.com/file/d/1eJ2vt-KH8enloRIFZLiuGGmPuVVEs2-T/view?usp=sharing>

Terms that can be used to describe the key concepts in its continuous integration and continuous delivery (CI/CD) processes include but are not limited to:

Pipeline: A pipeline in Jenkins is a collection of interconnected events or jobs. It represents the overall workflow process of delivering a piece of software. In the Jenkins file, a pipeline can be defined with domain-specific language (DSL) syntax. Pipelines describe complex workflows such as build, test, and deploy; often these are stages with arrows between them.

Node: A node is a machine that is part of the Jenkins environment and is capable of executing a pipeline or projects. Jenkins contains two types of nodes: master and agent. A master node is a central entity that holds control of Jenkins, scheduling the jobs, dispatching them to the worker nodes for execution, and later observing their execution. It can execute jobs directly.

Agent: In this context, the agent is a worker node. It is a configured system that offloads build projects from the master and executes jobs dispatched by the master. The agents may run on different operating systems as physical or virtual machines, Docker containers, or cloud instances. The terms "agent" and "slave" are more or less interchangeable, but "agent" is a newer term and is to be preferred.

Stage: Stage is the step that has been taken in the pipeline but has been implemented to act as a logical boundary in the CI/CD process. This is made to create a group of steps into a phase of the process. These stages are run in sequence, and the pipeline makes it possible that the user know where they are in the build/test/deploy process.

Steps: Steps are the smallest units of work within a pipeline and are contained within stages. Steps can be a specific command or task to be executed, such as code checkout from the version control system, or execution of any shell script or application deployment. Steps are running in order within a stage, and the success or failure of the steps can influence the advancement of the pipeline to the next stage. This is where Jenkins works in automating code integration, testing, and deployment processes, which makes it easier for the developers' team to integrate the change more often and hence identify the problem at an earlier stage of the development cycle.

Abdullah Hanoosh

```
Seen 1 remote branch
> git show-ref --tags -d # timeout=10
ERROR: Couldn't find any revision to build. Verify the repository and branch configuration for this job.
ERROR: Maximum checkout retry attempts reached, aborting
ERROR: Couldn't find any revision to build. Verify the repository and branch configuration for this job.
ERROR: Maximum checkout retry attempts reached, aborting
[withMaven] downstreamPipelineTriggerRunListener - Failure to introspect build steps: java.io.IOException: BinaryCalculator_pipeline #7 did not yet start
[withMaven] downstreamPipelineTriggerRunListener - Failure to introspect build steps: java.io.IOException: BinaryCalculator_pipeline #7 did not yet start
Finished: FAILURE
```

Branches to build ?

Branch Specifier (blank for 'any') ?

* / *

Add Branch

Repository browser ?

(Auto)

Additional Behaviours

Add ▾

Script Path ?

BinaryCalculatorWebapp/Jenkinsfile

☒ Lightweight checkout ?

[Pipeline Syntax](#)

GeneralBuild TriggersAdvanced Project OptionsPipeline

Pipeline

DefinitionPipeline script from SCM

SCMGit

Repositories

Repository URLhttps://github.com/jenkinsci/job-dsl-plugin.git

Credentials- none - Add

Advanced...

Add Repository

Branches to build

Branch Specifier (blank for 'any')*/master

Add Branch

Repository browser(Auto)

Additional BehavioursAdd

Script PathJenkinsfile

Lightweight checkout☒

[Pipeline Syntax](#)

SaveApply

abduh3131 / SQ-Delv-Homework

Type to search

CodeIssuesPull requestsActionsProjectsWikiSecurityInsightsSettings

General

Access

Collaborators

Moderation options

Code and automation

Branches

Tags

Rules

Actions

Webhooks

Environments

Codespaces

Pages

Security

Code security and analysis

Deploy keys

Secrets and variables

Integrations

GitHub Apps

Email notifications

Webhooks / Add webhook

We'll send a POST request to the URL below with details of any subscribed events. You can also specify which data format you'd like to receive (JSON, x-www-form-urlencoded, etc). More information can be found in our [developer documentation](#).

Payload URLhttp://abduh3131:8080/github-webhook/

Content typeapplication/json

Secret

Which events would you like to trigger this webhook?

☒ Just the push event.

☐ Send me everything.

☐ Let me select individual events.

☒ Active

We will deliver event details when this hook is triggered.

Add webhook

© 2024 GitHub, Inc.

Terms

Privacy

Security

Status

Docs

Contact

Manage cookies

Do not share my personal information