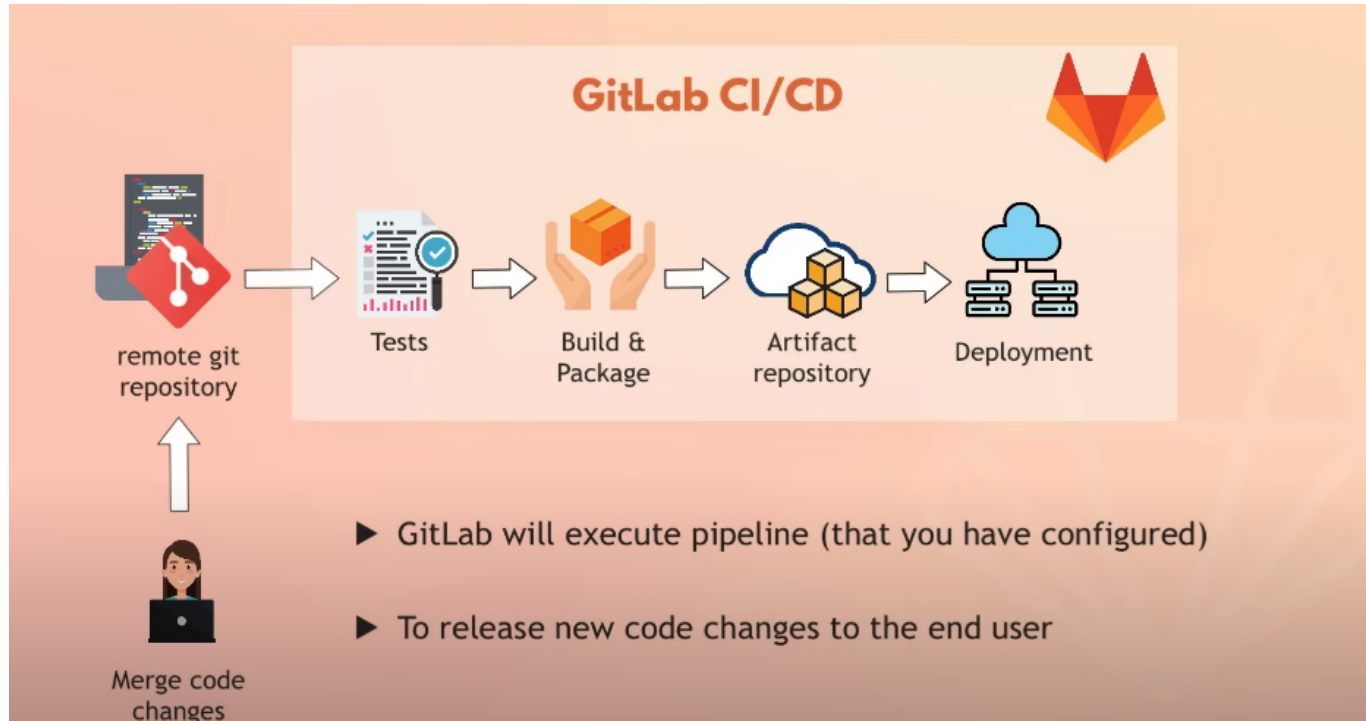


python app ->Run Test->Docker Image ->push to Docker Registry->Deploy on Server

Gitlab is platform which manages complete DevOps workflow

CI/CD It means continuously building testing release

![[Pasted image 20250703093256.png



continuously release code changes to end environment.

user ->changes into remote git repository ->test->build &package ->artifact repository ->deploy to dev ->deploy to staging ->deploy to production

Multiple CI/CD tools : Azure Pipelines , AWS Pipeline , Travis CI ,Jenkins most widely used

Benefit Gitlab : Allows keeping Ci/CD & code management in the same place

Email notification , can easy display build stage test through editor ,Self managed

Architecture :

Gitlab Instances or Gitlab Server where we can store manage , host application and CI/CD pipelines.

there are multiple machines are connected to server Gitlab Runners

server assigns jobs to available Runners

WWW.gitlab.com is actually instances that manages Runners -easy

available on gitlab to all users

www.gitlab.mycompany.com

Runners , Gitlab

For creating CI/CD pipeline
create file with name .gitlab-ci.yml

Runners are responsible for running your pipelines
Docker container -Image Of Ruby -execute in this env all this file

Stages-which sequence should follow

before-script:

- echo " Before build Step"

after-script:

- echo "cleaup activity"

bash-execute:

script:

- bash ./basic.sh

artifacts : To store the files

paths:

- "./myfolder"

when : on_success

access : all

expire_in : 30 days

we can create Environment Variables
and can use with \$

Pipeline Schedule : There is feature of creating build and pipeline with specific time and schedule
and can deploy manually.

Working With Git:

Create Project

build:

image: python:latest

script:

- python test.py

image: alpine:latest

pages:

script:

- mkdir public
- cp index.html public/

artifacts:

paths:

- "public"

only:

- main

The screenshot displays the GitLab CI-CD web interface for a project named 'GitLabCI-CD'. The interface includes a sidebar on the left with navigation options like 'Project', 'Pinned', 'Issues', 'Merge requests', 'Manage', 'Plan', 'Code', 'Build', 'Secure', 'Deploy', 'Operate', 'Monitor', and 'Analyze'. The main content area shows the 'main' branch with a commit history table. The table has columns for 'Name', 'Last commit', and 'Last update'. The commit history includes files like '.gitlab-ci.yml', 'README.md', 'index.html', and 'test.py'. The right sidebar provides 'Project information' such as '4 Commits', '1 Branch', '0 Tags', and '4 KIB Project Storage'. It also lists project files like 'README' and 'CI/CD configuration' with options to add various files like 'LICENSE', 'CHANGELOG', 'CONTRIBUTING', 'Kubernetes cluster', 'Wiki', and 'Integrations'. The 'Created on' date is 'July 03, 2025'.

Name	Last commit	Last update
.gitlab-ci.yml	Update .gitlab-ci.yml file	1 minute ago
README.md	Initial commit	22 minutes ago
index.html	Create index.html	22 minutes ago
test.py	Create test.py	21 minutes ago

on Left side there is Deploy option ->pages ->link

<https://my-static-resume-15a01f.gitlab.io/>