

Task 2 – CI/CD Pipelines Implementation

Description:

This task focuses on designing and implementing a Continuous Integration and Continuous Deployment (CI/CD) pipeline to automate the build, test, and deployment process of an application. The pipeline ensures faster delivery, reduced manual errors, and consistent deployments across environments. It integrates version control with automated workflows triggered on code changes.

Overview of the Task:

The CI/CD pipeline was configured to automatically trigger whenever code is pushed to the repository. The pipeline performs code checkout, dependency installation, application build, and basic validation steps. This setup demonstrates how modern DevOps teams ensure code quality and rapid deployment using automation tools.

Tools & Technologies Used:

- Git & GitHub
- GitHub Actions (CI/CD)
- Docker
- Docker Compose
- Linux shell commands

Pipeline Workflow:

1. Code push to GitHub repository triggers the pipeline.
2. Workflow checks out the source code.
3. Dependencies are installed automatically.
4. Docker image is built from the Dockerfile.
5. Containers are started using Docker Compose.
6. Basic validation ensures the application runs successfully.

Commands Used:

```
git add .  
git commit -m "Initial CI/CD pipeline setup"  
git push origin main
```

```
docker build -t app .  
docker compose up --build
```

```
docker ps
```

Configuration Files:

- .github/workflows/ci-cd.yml
- Dockerfile
- docker-compose.yml

Sample CI/CD Configuration (GitHub Actions):

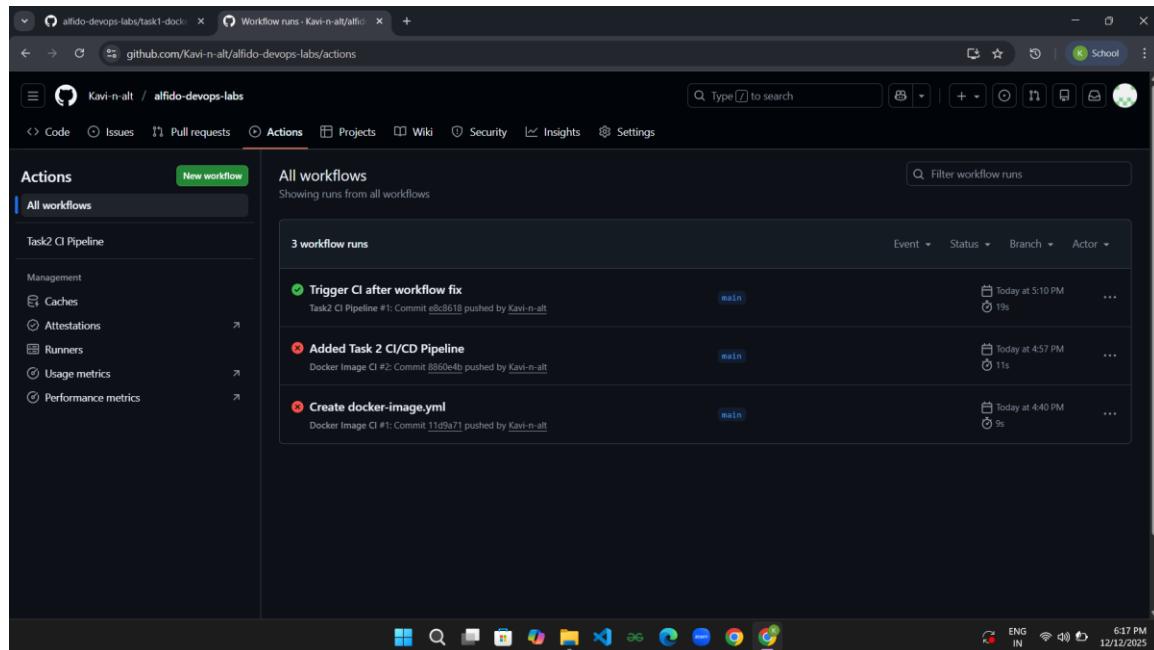
The workflow defines jobs and steps to automate the build and deployment process using YAML configuration. It specifies triggers, runner environment, and execution steps.

Outcome:

The CI/CD pipeline successfully automates application build and deployment on every code update. This task improved understanding of DevOps automation, pipeline configuration, and real-world CI/CD practices.

GitHub Repository:

<https://github.com/Kavi-n-alt/alfido-devops-labs>



The screenshot shows a GitHub Actions build log for a workflow named 'alifido-devops-labs/actions/runs/20175210925/job/57921347101'. The build status is 'succeeded 5 minutes ago in 16s'. The log details the steps taken:

- Build Docker Image**: 8s
- Post Set up Python**: 0s
- Post job cleanup.**: 1s
- Post Checkout repository**: 0s
- Post job cleanup.**: 2s
- /usr/bin/git version**
- git version 2.52.0**
- Temporarily overriding HOME='/home/runner/work/_temp/562901e8-f1f-4c65-a10d-133797d5e7d8' before making global git config changes**
- Add repository directory to the temporary git global config as a safe directory**
- /usr/bin/git config -global --add safe.directory /home/runner/work/alifido-devops-labs/alifido-devops-labs**
- /usr/bin/git config -local --name-only --get-regexp core.sshCommand**
- /usr/bin/git submodule foreach --recursive sh -c "git config -local --name-only --get-regexp 'core.sshCommand' && git config --local --unset-all 'core.sshCommand' || :"**
- /usr/bin/git config -local --name-only --get-regexp http.extraHeader**
- http://https://github.com/.extraHeader**
- /usr/bin/git config -local --unset-all http=https://github.com/.extraHeader**
- /usr/bin/git submodule foreach --recursive sh -c "git config --local --name-only --get-regexp 'http,https://github.com/.extraHeader' && git config --local --unset-all 'http,https://github.com/.extraHeader' || :"**
- /usr/bin/git config -local --name-only --get-regexp 'includeIf,gitdir:'**
- /usr/bin/git submodule foreach --recursive git config --local --show-origin --name-only --get-regexp remote.origin.url**
- Complete job**: 0s
- Cleaning up orphan processes**: 1s

The system tray at the bottom shows icons for file, search, taskbar, and other system status. The date and time are 12/12/2025 at 11:24 PM.

Summary

Jobs

build

Run details

Usage

Workflow file

build

Succeeded 5 minutes ago in 16s

Build Docker Image

```

27 # [internal] load build context
28 # $ context=~/Library/Python/3.10-slim:/ha256:8769c10045c67eac267537ff0950ce644eeebd9f0e88bec8604548a88f1f33d
29 # $ context=~/Library/Python/3.10-slim:/ha256:8769c10045c67eac267537ff0950ce644eeebd9f0e88bec8604548a88f1f33d done
30
31 # $ [1/5] FROM docker.io/library/python:3.10-slim@sha256:8769c10045c67eac267537ff0950ce644eeebd9f0e88bec8604548a88f1f33d
32 # $ resolve docker.io/library/python:3.10-slim@sha256:8769c10045c67eac267537ff0950ce644eeebd9f0e88bec8604548a88f1f33d
33 # $ sha256:8044fa0aa18a079a0972edc8a5f23979a4e098c595e21fb32a01c89eb0a70 0B / 13.82MB 0.1s
34 # $ sha256:8769c10045c67eac267537ff0950ce644eeebd9f0e88bec8604548a88f1f33d 10.37GB / 10.37GB done
35 # $ sha256:d6224bf0950c67eac267537ff0950ce644eeebd9f0e88bec8604548a88f1f33d 1.79GB / 1.79GB done
36 # $ sha256:c53e52ab67dc5704e27375e186f25fdccfc5da301fcfa2e596c7b83ba9e 5.48GB / 5.48GB done
37 # $ sha256:1733acd9540b3470ff17a0954bcde55a51279dd0da0f022d7883f1ad221e5 1.39GB / 29.78MB 0.1s
38 # $ sha256:9f9e626f823769571a9b05e10e91211abb44d5918c5a5f250d1657191548 0B / 1.29MB 0.1s
39 # $ sha256:004f0aa18a079a0973dcda5a5f23979a4e98c595e21fb32a01c89eb0a70 3.15GB / 13.82MB 0.3s
40 # $ sha256:1733acd9540b3470ff17a0954bcde55a51279dd0da0f022d7883f1ad221e5 16.78GB / 29.78MB 0.3s
41 # $ sha256:9f9e626f823769571a9b05e10e91211abb44d5918c5a5f250d1657191548 1.29MB / 1.29MB 0.2s done
42 # $ sha256:7bcaeb27cdff349f8a5e5b4db94d658b505fb3e3a0b7171a4d621298ef7cc3 248B / 248B 0.3s done
43 # $ sha256:004f0aa18a079a0973dcda5a5f23979a4e98c595e21fb32a01c89eb0a70 5.24MB / 13.82MB 0.4s
44 # $ sha256:1733acd9540b3470ff17a0954bcde55a51279dd0da0f022d7883f1ad221e5 27.20GB / 29.78MB 0.4s
45 # $ extracting sha256:1733acd9540b3470ff17a0954bcde55a51279dd0da0f022d7883f1ad221e5
46 # $ sha256:004f0aa18a079a0973dcda5a5f23979a4e98c595e21fb32a01c89eb0a70 8.39MB / 8.39MB 0.5s
47 # $ sha256:1733acd9540b3470ff17a0954bcde55a51279dd0da0f022d7883f1ad221e5 29.78GB / 29.78MB 0.4s done
48 # $ sha256:004f0aa18a079a0973dcda5a5f23979a4e98c595e21fb32a01c89eb0a70 12.58MB / 13.82MB 0.6s
49 # $ sha256:004f0aa18a079a0973dcda5a5f23979a4e98c595e21fb32a01c89eb0a70 13.82MB / 13.82MB 0.6s done
50 # $ extracting sha256:1733acd9540b3470ff17a0954bcde55a51279dd0da0f022d7883f1ad221e5 0.9s done
51 # $ extracting sha256:9f9e626f823769571a9b05e10e91211abb44d5918c5a5f250d1657191548 0.1s
52 # $ extracting sha256:9f9e626f823769571a9b05e10e91211abb44d5918c5a5f250d1657191548 0.1s
53 # $ extracting sha256:004f0aa18a079a0973dcda5a5f23979a4e98c595e21fb32a01c89eb0a70 0.7s done
54 # $ extracting sha256:004f0aa18a079a0973dcda5a5f23979a4e98c595e21fb32a01c89eb0a70 0.7s done
55 # $ extracting sha256:7bcaeb27cdff349f8a5e5b4db94d658b505fb3e3a071b4d621298ef7cc3 done

```

11:24 PM
12/12/2025

Summary

Jobs

build

Run details

Usage

Workflow file

build

Succeeded 4 minutes ago in 16s

Install dependencies

```

16 Downloading click-8.3.1-py3-none-any.whl.metadata (2.6 kB)
17 Collecting itsdangerous==2.2.0 (from flask->r requirements.txt [line 1])
18 Downloading itsdangerous-2.2.0-py3-none-any.whl.metadata (1.9 kB)
19 Collecting jinja2<3.1.2 (from flask->r requirements.txt [line 1])
20 Downloading jinja2-3.1.6-py3-none-any.whl.metadata (2.9 kB)
21 Collecting markupsafe<2.1.1 (from flask->r requirements.txt [line 1])
22 Downloading markupsafe-2.1.1 (from flask->r requirements.txt [line 1])
23 Collecting werkzeug>3.1.1.0 (from flask->r requirements.txt [line 3])
24 Downloading werkzeug-3.1.4-py3-none-any.whl.metadata (4.0 kB)
25 Downloading flask-3.1.2-py3-none-any.whl (103 kB)
26 Downloading blinker-1.9.0-py3-none-any.whl (8.5 kB)
27 Downloading click-8.3.1-py3-none-any.whl (108 kB)
28 Downloading itsdangerous-2.2.0-py3-none-any.whl (16 kB)
29 Downloading jinja2-3.1.6-py3-none-any.whl (134 kB)
30 Downloading markupsafe-3.0.3-cp310-cp310-manylinux2014_x86_64_manylinux_2_17_x86_64_manylinux_2_28_x86_64.whl.metadata (2.7 kB)
31 Downloading werkzeug-3.1.4-py3-none-any.whl (224 kB)
32 Installing collected packages: markupsafe, itsdangerous, click, blinker, werkzeug, jinja2, flask
33
34 Successfully installed blinker-1.9.0 click-8.3.1 flask-3.1.2 itsdangerous-2.2.0 jinja2-3.1.6 markupsafe-3.0.3 werkzeug-3.1.4

```

0s

Syntax Check

```

1 ▶ Run python -m py_compile app.py
2 python -m py_compile app.py
3 shell: /usr/bin/bash -e 0
4 env:
5 pythonLocation: /opt/hostedtoolcache/Python/3.10.19/x64
6 PKG_CONFIG_PATH: /opt/hostedtoolcache/Python/3.10.19/x64/lib/pkgconfig
7 Python_ROOT_DIR: /opt/hostedtoolcache/Python/3.10.19/x64
8 Pythonm_Root_Dir: /opt/hostedtoolcache/Python/3.10.19/x64

```

11:23 PM
12/12/2025

The screenshot shows two identical views of a GitHub Actions build log for a workflow named 'Trigger CI after fixing workflow'. The build was triggered by a commit message containing 'COMMIT_EDITMSG' and succeeded 4 minutes ago.

Checkout repository

```
+627091864aa0a3e3c5bf8684cd602ae3a6c234b:refs/remotes/origin/main
  From https://github.com/Kavi-n-alt/alfido-devops-labs
  * [new ref] 627091864aa0a3e3c5bf8684cd602ae3a6c234b -> origin/main
  Determining the checkout info
/usr/bin/git sparse-checkout disable
/usr/bin/git config --local --unset-all extensions.worktreeConfig
Checking out the ref
/usr/bin/git log -1 --format=SH
627091864aa0a3e3c5bf8684cd602ae3a6c234b
```

Set up Python

```
Run actions/setup-python@v4
  with:
    python-version: 3.10
    check-latest: false
    token: ***
    update-environment: true
    allow-prereleases: false
  Installed versions
  Successfully set up Python (3.10.19)
```

Install dependencies

```
Ran pip install -r requirements.txt
Collecting Flask (from -r requirements.txt (line 1))
  Downloading flask-2.1.2-py3-none-any.whl.metadata (3.2 kB)
Collecting blinker<1.9.0 (from flask->r requirements.txt (line 1))
  Downloading blinker-1.9.0-py3-none-any.whl.metadata (1.6 kB)
Collecting click>8.1.3 (from flask->r requirements.txt (line 1))
```

Summary

Jobs: build (green icon, successful)

Run details, Usage, Workflow file

Search logs: 1s

System status: ENG IN 11:23 PM 12/12/2025

The screenshot shows two consecutive GitHub Actions build logs for the 'build' job. Both logs were successful and completed 4 minutes ago in 16s. The logs are displayed in a dark-themed interface with a sidebar on the left.

Job Log 1 (Top):

```
build
succeeded 4 minutes ago in 16s

Set up job
  27 Complete job name: build

Checkout repository
  1 Run actions/checkout@v4
  2 with:
  3   repository: Kavi-n-alt/alfido-devops-labs
  4   token: ***
  5   ssh-strict: true
  6   ssh-user: git
  7   persist-credentials: true
  8   clean: true
  9   sparse-checkout-cone-mode: true
 10  fetch-depth: 1
 11  fetch-tags: false
 12  show-progress: true
 13  lfs: false
 14  submodules: false
 15  set-safe-directory: true
 16 Syncing repository: Kavi-n-alt/alfido-devops-labs
 17 Getting Git version info
 18 Working directory is '/home/runner/work/alfido-devops-labs/alfido-devops-labs'
 19 /usr/bin/git version
 20 git version 2.52.0
 21 Temporarily overriding HOME='/home/runner/work/_temp/7bd36915-1144-4773-9004-052cd2a56487' before making global git config changes
 22 Adding repository directory to the temporary git global config as a safe directory
 23 /usr/bin/git config --global --add safe.directory /home/runner/work/alfido-devops-labs/alfido-devops-labs
 24 Deleting the contents of '/home/runner/work/alfido-devops-labs/alfido-devops-labs'
 25 Initializing the repository
 26 /usr/bin/git init /home/runner/work/alfido-devops-labs/alfido-devops-labs
```

Job Log 2 (Bottom):

```
Trigger CI after fixing workflow #5
Re-run all jobs

build
succeeded 4 minutes ago in 16s

Set up job
  1 Current runner version: '2.329.0'
  2 Runner Image Provisioner
  3 Hosted Compute Agent
  4 Version: 20251208.163
  5 Commit: 6c10caca4910e198d66de23adf20ad317c474e3
  6 Build Date: 2025-12-02T15:56:59Z
  7 Worker ID: (c1bb5fe3-2852-4fc8-b6da-7839fb1a07a1)
  8 Operating System
  9 Ubuntu
 10 24.04.3
 11 LTS
 12 Runner Image
 13 Image: ubuntu-24.04
 14 Version: 20251208.163.1
 15 Included Software: https://github.com/actions/runner-images/blob/v20251208.163/images/ubuntu/Ubuntu_2404/Readme.md
 16 Image Release: https://github.com/actions/runner-images/releases/tag/ubuntu24%2F20251208.163
 17 GITHUB_TOKEN Permissions
 18 Contents: read
 19 Metadata: read
 20 Packages: read
 21 Secret source: Actions
 22 Prepare workflow directory
 23 Prepare all required actions
 24 Getting action download info
 25 Download action repository 'actions/checkout@v4' (SHA:34e114876ab011c390a56381ad16ebd13914#8d5)
 26 Download action repository 'actions/setup-python@v4' (SHA:7f4fc3e22c37d6ff65e8874f53bd3157c663f7c)
```

A screenshot of a GitHub Actions run summary. The pipeline is named "Trigger CI after workflow fix #1". The "build" job is highlighted in green, indicating success. The job summary shows it succeeded 1 hour ago in 15s. The steps listed are: Set up job, Checkout repository, Set up Python, Install dependencies, Syntax Check, Build Docker Image, Post Set up Python, Post Checkout repository, and Complete job. Each step is shown with its duration.

A screenshot of the GitHub Actions workflow file editor. The workflow file is named ".github/workflows/task2-ci.yml" and has a revision ID of e8c8618. The code defines a pipeline named "Task2 CI Pipeline" triggered by pushes to the "task2-ci-cd" branch. It uses an Ubuntu-latest runner and includes steps for checkout, setup Python (version 3.10), install dependencies (using pip install -r requirements.txt), syntax check (using python -m py_compile app.py), and build a Docker image (using docker build -t task2-image .).

```
name: Task2 CI Pipeline
on:
  push:
    paths:
      - "task2-ci-cd/**"
jobs:
  build:
    runs-on: ubuntu-latest
    steps:
      - name: Checkout repository
        uses: actions/checkout@v4
      - name: Set up Python
        uses: actions/setup-python@v4
        with:
          python-version: "3.10"
      - name: Install dependencies
        working-directory: task2-ci-cd/app
        run: pip install -r requirements.txt
      - name: Syntax Check
        working-directory: task2-ci-cd/app
        run: python -m py_compile app.py
      - name: Build Docker Image
        working-directory: task2-ci-cd/app
        run: docker build -t task2-image .
```

The screenshot shows a GitHub Actions build log for a workflow named 'task2-ci-cd'. The build step, labeled 'build', has completed successfully in 16 seconds. The log output is as follows:

```
9 Python3_ROOT_DIR: /opt/hostedtoolcache/Python/3.10.19/x64
10 LD_LIBRARY_PATH: /opt/hostedtoolcache/Python/3.10.19/x64/lib
1 Run docker build -t task2-image .
2 docker build < task2-image .
3 shell: /usr/bin/bash -e {0}
4 env:
5 pythonlocation: /opt/hostedtoolcache/Python/3.10.19/x64
6 PKG_CONFIG_PATH: /opt/hostedtoolcache/Python/3.10.19/x64/lib/pkgconfig
7 Python_ROOT_DIR: /opt/hostedtoolcache/Python/3.10.19/x64
8 Python2_ROOT_DIR: /opt/hostedtoolcache/Python/3.10.19/x64
9 Python3_ROOT_DIR: /opt/hostedtoolcache/Python/3.10.19/x64
10 LD_LIBRARY_PATH: /opt/hostedtoolcache/Python/3.10.19/x64/lib
11 #0 building with "default" instance using docker driver
12
13 #1 [internal] load build definition from Dockerfile
14 #1 transferring dockerfile: 2BIB done
15 #1 DONE 0.0s
16
17 #2 [auth] library/python:pull token for registry-1.docker.io
18 #2 DONE 0.0s
19
20 #3 [internal] load metadata for docker.io/library/python:3.10-slim
21 #3 DONE 0.5s
22
23 #4 [internal] load .dockignore
24 #4 transferring context: 2B done
25 #4 DONE 0.0s
```

The system tray at the bottom right shows the date and time as 12/12/2025, 11:23 PM, with battery level, signal strength, and other status icons.