

# CI/CD Pipeline with GitHub Actions & Docker

Author: Divyanshu Sharma

Date: September 8, 2025

GitHub Repository: <https://github.com/Pt-Divyanshu/ci-cd-docker-actions>

## 1. Project Objective

To establish a complete CI/CD pipeline that builds a Docker image, executes tests, and deploys locally using Minikube — all without relying on external cloud services.

## 2. Tools & Technologies

- GitHub Actions for CI/CD workflow automation
- Docker & Docker Hub for containerization and image registry
- Minikube (Kubernetes) for local deployment
- Python Flask as the demonstration application framework

## 3. Workflow Overview

- Push code to GitHub
- GitHub Actions executes: checkout, dependencies, tests, build, Docker Hub push
- Minikube Deployment: Pull Docker image, deploy to Kubernetes, expose via NodePort
- Access Flask app at `http://127.0.0.1:`

## 4. Implementation Steps

Commands to set up and run locally:

```
minikube start kubectl apply -f deployment.yaml kubectl apply -f service.yaml minikube tunnel #
```

Keep this terminal open

Visit the app at: `http://127.0.0.1:64812`

Expected output: Hello from CI/CD with GitHub Actions & Docker!

## 5. Final Deliverables

Deliverable	Description
GitHub repository	Includes workflows and project files
Docker Hub image link	<code>divyanshush/ci-cd-demo:latest</code>
CI/CD workflow result	Screenshot showcasing successful GitHub Actions run

Deployed app screenshot	Proof of Flask app running via Minikube on your machine
-------------------------	---

## 6. Conclusion & Learnings

This project solidified understanding of:

- Setting up CI/CD pipelines via GitHub Actions
- Creating and managing Docker containers
- Deploying containerized apps locally using Minikube
- Seamless integration of tooling for a full development pipeline