

# CI/CD Pipeline with GitHub Actions & Docker

## (No Cloud Needed)

**Name:** Himesh Sutar

### Introduction

The goal of this project was to implement a complete CI/CD pipeline using GitHub Actions and Docker, enabling local deployment without relying on any cloud services. A Node.js application was used to demonstrate the workflow, incorporating automated testing, Docker image creation, and deployment. The pipeline is designed to streamline the development lifecycle by testing, building, and deploying a Docker container entirely on a local environment using tools like Docker and Mini Kube.

### Abstract

This project involved setting up a CI/CD pipeline for a Node.js application, fully automated using GitHub Actions. The pipeline runs tests, builds Docker images, and pushes them to Docker Hub. These images are then pulled and deployed locally using Docker or Mini Kube, eliminating the need for cloud infrastructure. Additional features such as custom routing were implemented to demonstrate the application's extensibility.

### Tools Used

- Node.js & Express
- GitHub & GitHub Actions
- Docker & Docker Hub
- Mini Kube or Local VM
- Git, VS Code

## Steps Involved in Building the Project

1. Created a Node.js app with Express including a new '/api/user/profile' route.
2. Wrote a Dockerfile to containerize the application.
3. Configured GitHub Actions workflow to test, build, and push the image.
4. Set up Docker Hub repository and connected it to the CI workflow.
5. Deployed the image locally using Docker or Mini Kube.
6. Verified deployment and functionality via browser and terminal.
7. Captured screenshots of successful deployment and workflow runs.

## Conclusion

The CI/CD pipeline was successfully implemented, demonstrating automated build and deployment of a Node.js application using GitHub Actions and Docker. The addition of new routes and automatic redeployment validated the pipeline's efficiency. This setup can be scaled to production-level environments.