

CI/CD Pipeline with GitHub Actions & Docker

Abstract

This project demonstrates a complete CI/CD pipeline using GitHub Actions and Docker. The pipeline automates the process of building, testing, and deploying a simple Node.js application. The Docker image is pushed to Docker Hub and can be pulled and run on any machine or cloud server. The goal of this project is to showcase automation, containerization, and deployment best practices.

Tools Used

- Node.js & Express.js (application) - Docker (containerization) - Docker Hub (registry) - GitHub Actions (CI/CD automation) - AWS EC2 Ubuntu (deployment & testing)

Steps Involved in Building the Project

1. Created a simple Node.js app with an Express server. 2. Wrote a basic test script to validate the setup. 3. Created a Dockerfile to containerize the app. 4. Built and ran the image locally using Docker. 5. Configured GitHub Actions workflow to install dependencies, run tests, build image, and push to Docker Hub. 6. Deployed and tested the container on an AWS EC2 instance, confirming successful response at <http://:3000/>

Conclusion

This project successfully automated the build and deployment pipeline for a Node.js application using GitHub Actions and Docker. The use of CI/CD ensured continuous integration and delivery, while Docker provided portability and consistency. The app was successfully tested on an AWS EC2 instance, proving the reliability of the pipeline. The project highlights how DevOps practices streamline software delivery in real-world scenarios.