Project 1 – CI/CD Pipeline with Github Actions and Docker

Introduction

Continuous Integration and Continuous Deployment (CI/CD) are fundamental in modern software development to automate builds, tests, and deployments.

This project demonstrates a practical implementation of a CI/CD pipeline for a simple application that includes:

- A backend built with Node.js and Express, exposing health and counter APIs with Redis as the in-memory database.
- A frontend built with React that interacts with the backend to display health status and counter value.
- Services containerized using Docker and orchestrated via Docker Compose.

The CI/CD pipeline is configured in GitHub Actions and pushes built images to Docker Hub, enabling automated testing and deployment.

Tools Used

- **GitHub Actions** : Automates pipeline tasks: checkout, test, build, and push
- **Docker & Docker Compose** : Containerizes backend, frontend, and Redis service
- **Docker Hub** : Public Docker image repository
- **Node.js + Express** : Backend service
- **React** : Frontend UI framework
- **Redis** : In-memory key-value store
- **Jest :** Backend testing framework

Steps Involved in Building the Project

1. **Application Development**
    - Backend provides /health and /counter.
    - Frontend displays health and counter, incrementing the counter with a button click.
2. **Containerization**
    - Created Dockerfiles for backend and frontend.
    - Docker Compose file sets up services (Redis, backend, frontend).

3. **CI/CD Pipeline**
    - GitHub Actions configured to run on main branch push or PR.
    - Steps:
        1. Checkout code
        2. Install dependencies and run Jest tests
        3. Build Docker images
        4. Push images to Docker Hub
4. **Local Deployment**
    - Pull images from Docker Hub.
    - Start services with docker compose up -d.
    - Verified working at http://localhost:3000.

Conclusion

The project successfully demonstrates a real CI/CD pipeline for a simple full-stack app. It automates building, testing, and deploying the application, helping developers deploy consistently without manual steps.

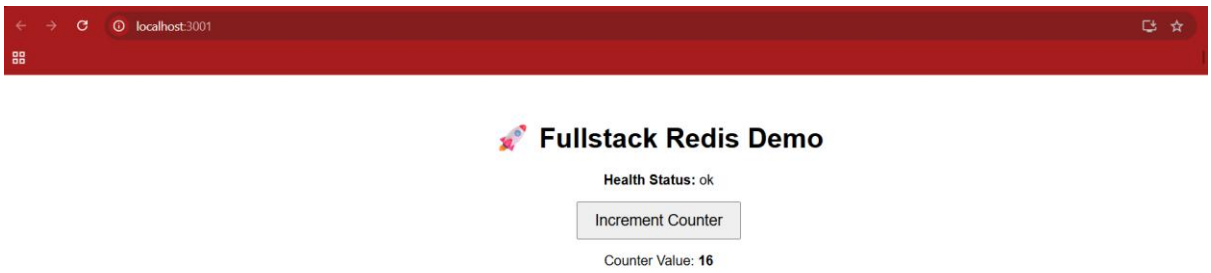

Fig.1 Deployed app