

# DevOps CI/CD Pipeline With Local Deployment

---

This is an exercise about building a DevOps CI/CD Pipeline with the following criteria:

## Application Setup

- It's a simple Node application with:
  - A frontend
  - A backend
- They are **dockerised** using:
  - a frontend `Dockerfile`
  - a backend `Dockerfile`
  - `docker-compose.yml`

## GitHub Actions Workflow

On Merge Request to `dev`:

1. Run linting  
Example: `npm run lint`
  2. Run `npm audit`
  3. Run static code security analysis using `snyk`
  4. Execute all tests
  5. Build Docker images
- If all steps pass, allow the merge from `feature` to `dev`

On Merge to `main`:

1. Build production-ready Docker images
2. Copy secrets from GitHub Secrets Manager by creating `.env` files into the image(s)
3. Push the images to Docker Hub:
  - Since free accounts allow only one repository:
    - Use tags like `:frontend` and `:backend`

## Deployment

- A GitHub webhook triggers a `POST` to a URL, the URL is an `ngrok` tunnel to `localhost`
- A NodeJS Express app is listening on this endpoint
- It performs the following:
  1. Brings down the currently running app (if it's running)  
Command: `docker-compose down`
  2. Pulls the latest images from Docker Hub
  3. Restarts the app with the new images  
Command: `docker-compose up -d`

this local node app will be provided for you

# References

---

- [accessing github secrets](#)
- [eslint](#)
- [snyk](#)