Engineer

Built GitHub Actions CI/CD pipelines

Built GitHub Actions CI/CD pipelines with a distroless production frontend image and multi-environment promotion.

2025

one pipeline, three environments, a distroless image

Situation. Shipping shouldn't depend on someone remembering the steps, and what ends up running in production shouldn't be a fat general-purpose container carrying a shell and a package manager it'll never use — that's just attack surface sitting there for no reason.

Task. Make the path from commit to running-in-Azure automatic, and keep the production images as small and locked-down as each workload allows.

Action. The automation runs on GitHub Actions. Separate workflows handle the code-quality gate, the tests, and the per-environment deploys, with CodeQL, dependency review and an SBOM step alongside them, so nothing reaches an environment without passing the checks first. The images are multi-stage builds, and the base for each part was picked on its merits rather than as one blanket choice: the frontend ships on a distroless image (gcr.io/distroless/cc-debian13 — no shell, no package manager), the Go API on a slim Alpine, and the database image on postgres-slim. Promotion moves a build through the environments along a defined route rather than by hand.

Result. Releases stopped being a careful manual ritual and became a routine, boring event, which is exactly what you want from releases. The production frontend runs on about as little as you can give it, the checks catch problems before they land, and "deploy" is something the workflows do rather than something anyone sweats through.

Bring this experience to your project

Use “Built GitHub Actions CI/CD pipelines” as a starting point for a similar challenge in your organisation. The services behind this work are available for your project. Tested, released and reversible, so changes reach customers on a Tuesday afternoon rather than late on a Friday night.

Explore our services Get in touch →

Look behind the result

Read the situation, decisions and result in this example. We can discuss which parts transfer to your constraints and which need another approach. Built GitHub Actions CI/CD pipelines — one pipeline, three environments, a distroless image

Explore our services Get in touch →

Need a proper CI/CD pipeline?

Talk through your project with the engineer responsible for delivery. We agree the scope, price basis and handover in writing before work begins.

Role context: DevOps Engineer. It draws on DevOps, Cloud, Containers (Docker/Kubernetes), Automation & CI/CD, Reliability & Backups, Security and Testing & QA. And while you are here, we are rather good at DevOps & CI/CD Automation and Containerization & Orchestration.

Get in touch Explore our services

One piece of **a longer record** — and it is all on this site.

Every entry is written the same way: the situation, the task, what we did and what changed.

See the whole record

Define the improvement you need

Bring the deployment path, the operational risks and the people on call. We can scope a change with checks, a recovery route and clear ownership.

See what it costs Get in touch →

Need a proper CI/CD pipeline?

Ask about the decisions behind “Built GitHub Actions CI/CD pipelines”. You will speak with the people who can explain the work and assess your own project.

Explore our services

welcome@engineer.company +45 50 10 61 84 Copenhagen, Denmark

Take this page with you

Contact card

Carefully Crafted With Love by D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Copyright © 2025 – present · Engineer ApS · All rights reserved.

Printed from https://engineer.company/portfolio/built-github-actions-ci-cd-pipelines/