

GitHub Actions Workflow Quick Reference

Workflow Overview

Name: CI/CD Pipeline

File: `.github/workflows/ci-cd.yml`

Status: <https://github.com/salimemp/pro-pdf/actions>

Quick Links




- **Actions Dashboard:** <https://github.com/salimemp/pro-pdf/actions>
- **Latest Run:** <https://github.com/salimemp/pro-pdf/actions/workflows/ci-cd.yml>
- **Workflow File:** <https://github.com/salimemp/pro-pdf/blob/master/.github/workflows/ci-cd.yml>

Pipeline Stages






Jobs Breakdown

1. Lint and Type Check

-  TypeScript compilation
-  ESLint (warnings only)
-  ~2-3 minutes

2. Build Application

-  Install dependencies
-  Generate Prisma client
-  Build Next.js app

- 📁 Upload artifacts
- ⌚ ~5-7 minutes

3. E2E Tests

- 📦 PostgreSQL database
- 🧑‍💻 Playwright + Chromium
- 🖋 Database seed
- 🛠 Run tests
- 📸 Upload screenshots on failure
- ⌚ ~8-10 minutes

4. Security Scan

- 🔒 Vulnerability audit
- 📅 Dependency check
- ⌚ ~1-2 minutes

5. Build Status Summary

- 📊 Aggregate results
- ⌚ ~30 seconds

Total Pipeline Time: ~15-20 minutes

Triggers

Automatic

```
# Pushes to master/main
git push origin master

# Pull requests to master/main
git push origin feature/my-branch
# Then create PR on GitHub
```

Manual

1. Go to Actions tab
2. Select "CI/CD Pipeline"
3. Click "Run workflow"
4. Choose branch
5. Run

Viewing Results

Status Badge

Add to your README:

```
[![CI/CD Pipeline](https://github.com/salimemp/pro-pdf/actions/workflows/ci-cd.yml/badge.svg)](https://github.com/salimemp/pro-pdf/actions/workflows/ci-cd.yml)
```

Job Logs

1. Go to Actions tab
2. Click on workflow run
3. Click on job name
4. View detailed logs

Artifacts

1. Open workflow run
2. Scroll to “Artifacts” section
3. Download:
 - build-artifacts (7 days retention)
 - playwright-report (7 days retention)
 - test-screenshots (7 days, on failure)



Common Issues

✖ TypeScript Errors

```
# Local check  
yarn tsc --noEmit
```

✖ Build Failures

```
# Local build  
yarn build
```

✖ Test Failures

```
# Run specific test  
yarn playwright test e2e/auth.spec.ts  
  
# Debug mode  
yarn playwright test --debug  
  
# View report  
yarn playwright show-report
```



Security Vulnerabilities

```
# Check vulnerabilities  
yarn audit  
  
# Fix automatically  
yarn audit fix  
  
# Interactive upgrade  
yarn upgrade-interactive
```



Environment Variables

```
# Required for all runs
NODE_OPTIONS: '--max-old-space-size=4096'
__NEXT_TEST_MODE: '1'
NEXT_DIST_DIR: '.build'

# E2E tests only
DATABASE_URL: postgresql://testuser:testpassword@localhost:5432/testdb
NEXTAUTH_SECRET: test_secret_key_for_github_actions
NEXTAUTH_URL: http://localhost:3000
```



Configuration Files

- `.github/workflows/ci-cd.yml` - Main workflow
- `playwright.config.ts` - E2E test config
- `tsconfig.json` - TypeScript config
- `.eslintrc.json` - Linting rules
- `package.json` - Scripts and dependencies



Additional Resources

- **Full Guide:** See `CI_CD_GUIDE.md`
- **GitHub Actions:** <https://docs.github.com/en/actions>
- **Playwright CI:** <https://playwright.dev/docs/ci>
- **Next.js CI/CD:** <https://nextjs.org/docs/deployment>



Pre-Push Checklist

- ☐ Run `yarn tsc --noEmit`
- ☐ Run `yarn lint`
- ☐ Run `yarn build`
- ☐ Run `yarn playwright test`
- ☐ Commit with meaningful message
- ☐ Push to feature branch first
- ☐ Create PR for review
- ☐ Check CI/CD status
- ☐ Merge when green

Need Help? Check the [full CI/CD guide](#) (`./CI_CD_GUIDE.md`)