

Frontend CI/CD Workflow Documentation (React Portfolio)

1. Overview

The React frontend of the portfolio is set up with continuous integration and continuous deployment (CI/CD) to automate testing, building, and deployment.

Key Features:

- Automatic builds and deployments on every push to the main branch.
- Integration with Vercel for production deployment.
- Automated linting and testing to ensure code quality.
- Smooth workflow with minimal manual intervention.

2. Repository Structure

The project follows a standard React structure:

- **public/** – Static assets like images and icons.
- **src/** – Source code including components, state management, and CSS.
- **.github/workflows/** – Contains GitHub Actions workflow files (frontend-ci.yml).
- **package.json** – Project configuration with scripts for build, lint, and test.

3. CI/CD Pipeline Components

The pipeline uses **GitHub Actions** to automate the following steps:

- **Trigger:** On push or pull_request events to the main branch.
- **Checkout Code:** Pulls the latest code from GitHub.
- **Setup Node.js:** Ensures the environment has the correct Node.js version for React.
- **Install Dependencies:** Uses npm ci for a clean installation.
- **Linting:** Checks for code style issues using configured linters.
- **Testing:** Runs all unit tests (npm run test) before deployment.
- **Build:** Creates optimized static files (npm run build) for deployment.
- **Deploy:** Pushes the build to **Vercel** using a GitHub Action configured with secrets.

4. Deployment on Vercel

- **Production URL:** <https://sowndarya-ragavan.vercel.app>
- Vercel provides **preview deployments** for pull requests and **automatic production**

5. Advantages of the Workflow

- **Automation:** Reduces manual deployment steps.
- **Consistency:** Linting and testing prevent errors from reaching production.
- **Fast Feedback:** Preview deployments allow quick testing of new features.
- **Scalability:** Easily extendable to new projects or microservices.

6. CI/CD Workflow Summary

The CI/CD process ensures that **any code pushed to the repository is automatically validated and deployed** to production with minimal downtime.

Textual Flow:

