

Experiment 5.2

Node.js CI/CD Pipeline

Code-

```
1 # This name will appear in the "Actions" tab of your GitHub repository.
2 name: Node.js CI/CD
3
4 # --- 1. Trigger Configuration ---
5 # This workflow runs on:
6 # - A push to the 'main' branch
7 # - A pull request targeting the 'main' branch
8 on:
9   push:
10     branches: [ main ]
11   pull_request:
12     branches: [ main ]
13
14 # --- 2. Job Configuration ---
15 # A workflow is made of one or more jobs. This file has one job named 'build-and
16   -test'.
17 jobs:
18   build -and -test:
19     # The type of virtual machine to run the job on. 'ubuntu-latest' is a good
20     # default.
21     runs-on: ubuntu-latest
22
23     # 'steps' are the sequence of tasks that will be executed.
24     steps:
25       # --- Step 1: Check out your code ---
26       # This action checks-out your repository's code so the job can access it.
27       - name: Checkout repository
28         uses: actions/checkout@v4
29
30       # --- Step 2: Set up Node.js ---
31       # This action sets up a specific Node.js version.
32       - name: Set up Node.js v20
33         uses: actions/setup-node@v4
34         with:
35           node-version: 20
36           # This caches your 'node_modules' directory for faster subsequent runs
37           -
38             cache: 'npm'
39
40       # --- Step 3: Install Dependencies ---
41       # 'npm ci' is recommended for CI as it's faster and uses 'package-lock.
42       # json'.
43       - name: Install dependencies
44         run: npm ci
45
46       # --- Step 4: Run Tests ---
47       # Runs the 'test' script defined in your 'package.json'.
48       - name: Run tests
```

```

45         run: npm test
46
47     # --- Step 5: Build Project ---
48     # Runs the 'build' script defined in your 'package.json'.
49     - name: Build project
50       run: npm run build
51
52 # --- 3. Optional Deployment to GitHub Pages ---
53 # To enable deployment, you need to:
54 # 1. Add the 'permissions' block (see below) to the top level (right under 'name
55 # 2. Uncomment the entire 'Deploy to GitHub Pages' step.
56 # 3. Change 'folder: build' to match your project's build output directory (e.g
57 # 4. In your repository settings, under "Pages", set the source to "Deploy from
58 #    and select the 'gh-pages' branch.
59
60 # --- UNCOMMENT THIS BLOCK AND PUT IT AT THE TOP LEVEL TO ENABLE DEPLOY ---
61 # permissions:
62 #   contents: write # Required to push the build to the 'gh-pages' branch
63
64 # --- UNCOMMENT THIS STEP TO ENABLE DEPLOY ---
65 #     - name: Deploy to GitHub Pages
66 #       # This 'if' condition ensures deployment only happens on a push to '
67 #       # not on pull requests.
68 #       if: github.ref == 'refs/heads/main'
69 #       uses: JamesIves/github-pages-deploy-action@v4
70 #       with:
71 #         token: ${{ secrets.GITHUB_TOKEN }} # A token is automatically
72 #         provided by GitHub
73 #         branch: gh-pages # The branch to deploy your built code to
74 #         folder: build # *** IMPORTANT: Change this to your build output
75 #         directory ***

```

main.yml