

# Challenges encountered & solutions

## 1. Self-hosted runner setup stuck

- **Issue:** Runner was stuck “Listening for Jobs” after update, and couldn’t return to terminal.
- **How I fixed it:** Used sudo ./svc.sh start to run the runner as a system service. This freed the terminal while keeping the runner active.

## 2. Git / repository issues

- **Issue:** Tried to add and push files in the wrong folder (actions-runner) and confusion about GitHub credentials.
- **How I fixed it:**
  - Navigated to the correct project folder (nodejs-ci-cd-app) before running git add and git push.
  - Used **GitHub Personal Access Token** for authentication instead of account password.

## 3. Invalid package.json

- **Issue:** npm test failed with EJSONPARSE due to invalid JSON syntax.
- **How I fixed it:** Rewrote package.json with valid JSON, ensured proper braces, double quotes, no trailing commas, and included necessary dependencies and scripts.

## 4. Test module errors

- **Issue 1:** Cannot find module './app' → test file had wrong path.
- **Issue 2:** Cannot find module 'chai' → missing dependencies.

- **How I fixed it:**
  - Updated test file paths to correctly point to ../src/server.js.
  - Removed unnecessary server.test.js that required Chai.
  - Installed missing packages if needed: npm install --save-dev chai chai-http.

## 5. Server not running

- **Issue:** curl http://localhost:3000 failed; node index.js failed because index.js didn't exist.
- **How I fixed it:**
  - Created src/server.js with Express app and export for tests.
  - Started the server manually using node src/server.js or in the background using PM2.

## 6. PM2 permission error

- **Issue:** npm install -g pm2 gave EACCES: permission denied.
- **How I fixed it:**
  - Installed globally with sudo npm install -g pm2.
  - Alternatively, installed locally with npm install pm2 --save-dev and ran with npx pm2.

## 7. Workflow verification

- **Issue:** GitHub Actions test job was passing but deploy job wasn't visible.
- **How I fixed it:** Planned to create a .github/workflows/ci-cd.yml that:
  - Has deploy job dependent on test (needs: test)

- Deploys the Node.js app to PM2-managed EC2 runner