

# Experiment 5.2

## Aim

To set up a CI/CD pipeline using GitHub Actions to automate testing, building, and optionally deploying a project upon code changes.

## Introduction

Continuous Integration (CI) and Continuous Deployment (CD) are essential practices in modern software development. GitHub Actions allows developers to automate workflows triggered by events like code pushes, pull requests, or scheduled intervals. This experiment demonstrates how to configure a GitHub Actions workflow to test, build, and optionally deploy a Node.js project automatically.

## Procedure

1. Create a Node.js project and initialize Git in your repository.
2. Create a `.github/workflows/main.yml` file in your project repository.
3. Configure the workflow to trigger on every push to the `main` branch.
4. Set up Node.js in the workflow environment.
5. Install project dependencies using `npm install`.
6. Run tests automatically using `npm test`.
7. Build the project using `npm run build`.
8. Optionally, add deployment steps to publish the build artifacts to platforms like GitHub Pages, Netlify, or AWS S3 using appropriate GitHub Actions.
9. Commit and push the changes to the `main` branch and verify the workflow execution in the repository's "Actions" tab.

## Code (main.yml)

```
name: Node.js CI/CD Pipeline

on:
  push:
    branches: [ main ]

jobs:
  build:
    runs-on: ubuntu-latest

    steps:
      - uses: actions/checkout@v3
      - name: Set up Node.js
```

```
uses: actions/setup-node@v3
with:
  node-version: '18'

- name: Install dependencies
  run: npm install

- name: Run tests
  run: npm test

- name: Build project
  run: npm run build

# Optional deployment step
# - name: Deploy to GitHub Pages
#   uses: peaceiris/actions-gh-pages@v3
#   with:
#     github_token: ${ secrets.GITHUB_TOKEN }
#     publish_dir: ./build
```

## Output

- The workflow runs automatically on every push to the `main` branch.
- Tests are executed, and the project is built automatically.
- Workflow status is visible in the repository's "Actions" tab.
- Optional deployment step deploys the build to the chosen hosting platform.

## Learning Outcomes

- Understand how to automate CI/CD workflows using GitHub Actions.
- Learn to trigger workflows on repository events like code push.
- Gain knowledge of running automated tests and builds in a workflow.
- Learn to integrate deployment steps for continuous delivery.
- Understand monitoring workflow status and troubleshooting failed jobs.