

Lab Exercise 7- Automation Testing using GitHub Actions (For Spring Boot API + Postman + Newman)

1. Objective

In this lab, learners will:

1. Push a Spring Boot project to GitHub
 2. Add Postman collection tests
 3. Create a GitHub Actions workflow
 4. Run automated API tests using Newman in CI
 5. View test reports directly in GitHub Actions
 6. Make the build fail if any test fails
-

2. Prerequisites

Install:

- Git
- GitHub account
- Jenkins not required
- NodeJS (if you want to test locally)
- Postman to create and export test collections

You should already have:

- Spring Boot API

- Postman collection (.json)
-

3. Folder Structure for the Lab

Your GitHub repository structure should include:

```
project-root/
├── src/
├── pom.xml
├── Dockerfile (optional)
├── postman/
│   └── springboot-tests.postman_collection.json
└── .github/
    ├── workflows/
    │   └── api-tests.yml
```

4. Step 1: Upload Spring Boot Project to GitHub

In your system:

```
git config --global user.name "Your Name"
```

```
git config --global user.email "your_email@example.com"
```

```
git init
```

```
git add .
```

```
git commit -m "Initial Spring Boot API commit"
```

```
git branch -M main
```

```
git remote add origin https://github.com/<username>/<repo>.git
```

```
git push -u origin main
```

6. Step 3: Add GitHub Actions Workflow

Inside it create:

```
api-tests.yml
```

7. Complete GitHub Actions Workflow

This workflow:

- Checks out your code
- Installs Java (Maven)
- Builds Spring Boot project
- Starts Spring Boot API server
- Installs Newman
- Runs Postman collection tests

Copy this EXACT file:

```
name: API Automation Testing

on:
  push:
    branches: [ "main" ]
  pull_request:
    branches: [ "main" ]

jobs:
  run-api-tests:
    runs-on: ubuntu-latest

    steps:
      - name: Checkout code
        uses: actions/checkout@v4

      - name: Set up JDK 17
        uses: actions/setup-java@v4
        with:
          java-version: "17"
          distribution: "temurin"

      - name: Build Spring Boot App
        run: mvn -B package --file pom.xml

      - name: Run Spring Boot App in background
        run: |
          nohup java -jar target/*.jar &
          sleep 20
```

```
- name: Install Newman
  run: |
    npm install -g newman
    npm install -g newman-reporter-htmlextra

- name: Run Postman Tests with Newman
  run: |
    newman run postman/springboot-tests.postman_collection.json \
    -r htmlextra --reporter-htmlextra-export newman-report.html

- name: Upload Test Report
  uses: actions/upload-artifact@v4
  with:
    name: API-Test-Report
    path: newman-report.html
```

8. Step 4: Commit and Push the Workflow

```
git add .github/workflows/api-tests.yml
git commit -m "Added automated API testing workflow"
git push
```

9. Step 5: Run the CI Pipeline

Go to GitHub:

GitHub → Actions → API Automation Testing → Artifacts You will see the workflow

running automatically after each push.

10. Understanding the Workflow Stages

1. Checkout code

Clones your repository.

2. Set up Java

Installs JDK (needed for Spring Boot build).

3. Build application

Runs Maven to build a .jar file.

4. Run application

Starts the API on port 8080 inside GitHub Actions VM.

5. Install Newman

Installs CLI runner for Postman tests.

6. Execute tests

Runs your Postman collection through Newman.

7. Upload report

Stores report as downloadable artifact.