GitHub Actions

Crash Course!

Dr. Firas Moosvi   **July 13, 2024**

Outline

- Continuous Integration, Delivery, Deployment

- Introduction to GitHub Actions

    - Motivation

    - Syntax

    - Practical Example

- Deploy a Django site on render.com

    - Repository Secrets

    - Troubleshooting GitHub Actions

- Q&A

CI/CD

CI/CD

CONTINUOUS INTEGRATION

BUILD → TEST → MERGE

CONTINUOUS DELIVERY

AUTOMATICALLY RELEASE TO REPOSITORY

CONTINUOUS DEPLOYMENT

AUTOMATICALLY DEPLOY TO PRODUCTION

Introduction to GitHub Actions
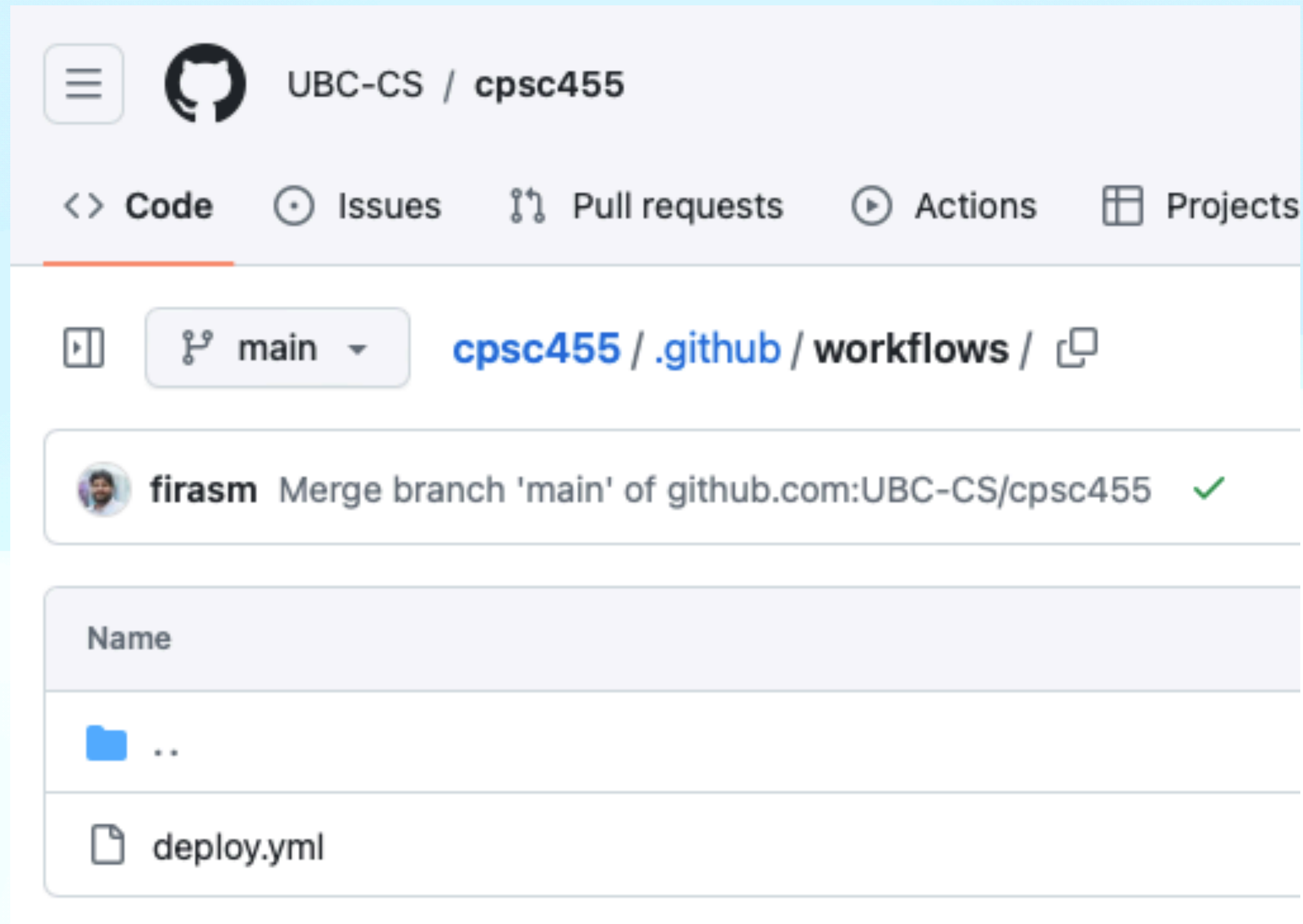
Clone URL (to follow along): https://classroom.github.com/a/c_ZEKiyH

The components of GitHub Actions 🔗

You can configure a GitHub Actions *workflow* to be triggered when an *event* occurs in your repository, such as a pull request being opened or an issue being created. Your workflow contains one or more *jobs* which can run in sequential order or in parallel. Each job will run inside its own virtual machine *runner*, or inside a container, and has one or more *steps* that either run a script that you define or run an *action*, which is a reusable extension that can simplify your workflow.
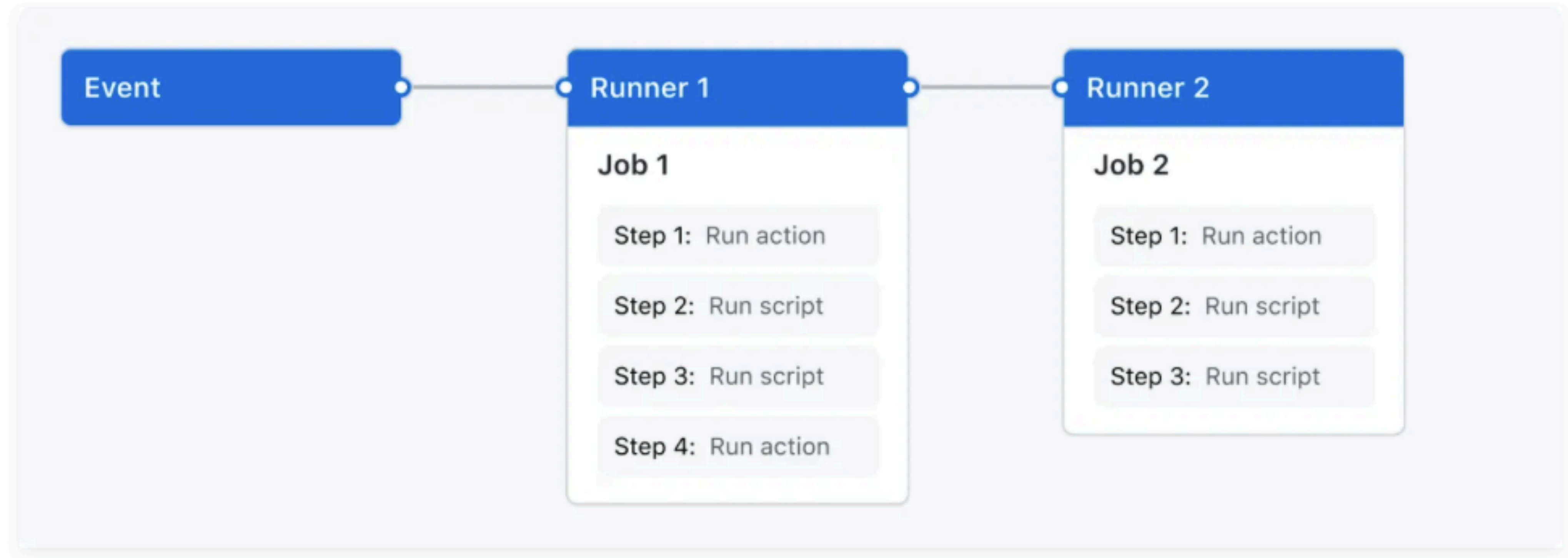
Understanding the workflow file 🔗

To help you understand how YAML syntax is used to create a workflow file, this section explains each line of the introduction's example:

YAML	Beside	Inline	📋

```yaml
name: learn-github-actions
```
Optional - The name of the workflow as it will appear in the "Actions" tab of the GitHub repository. If this field is omitted, the name of the workflow file will be used instead.

```yaml
run-name: ${{ github.actor }} is learning GitHub Actions
```
Optional - The name for workflow runs generated from the workflow, which will appear in the list of workflow runs on your repository's "Actions" tab. This example uses an expression with the `github` context to display the username of the actor that triggered the workflow run. For more information, see "Workflow syntax for GitHub Actions."

Deploy a Django Site!
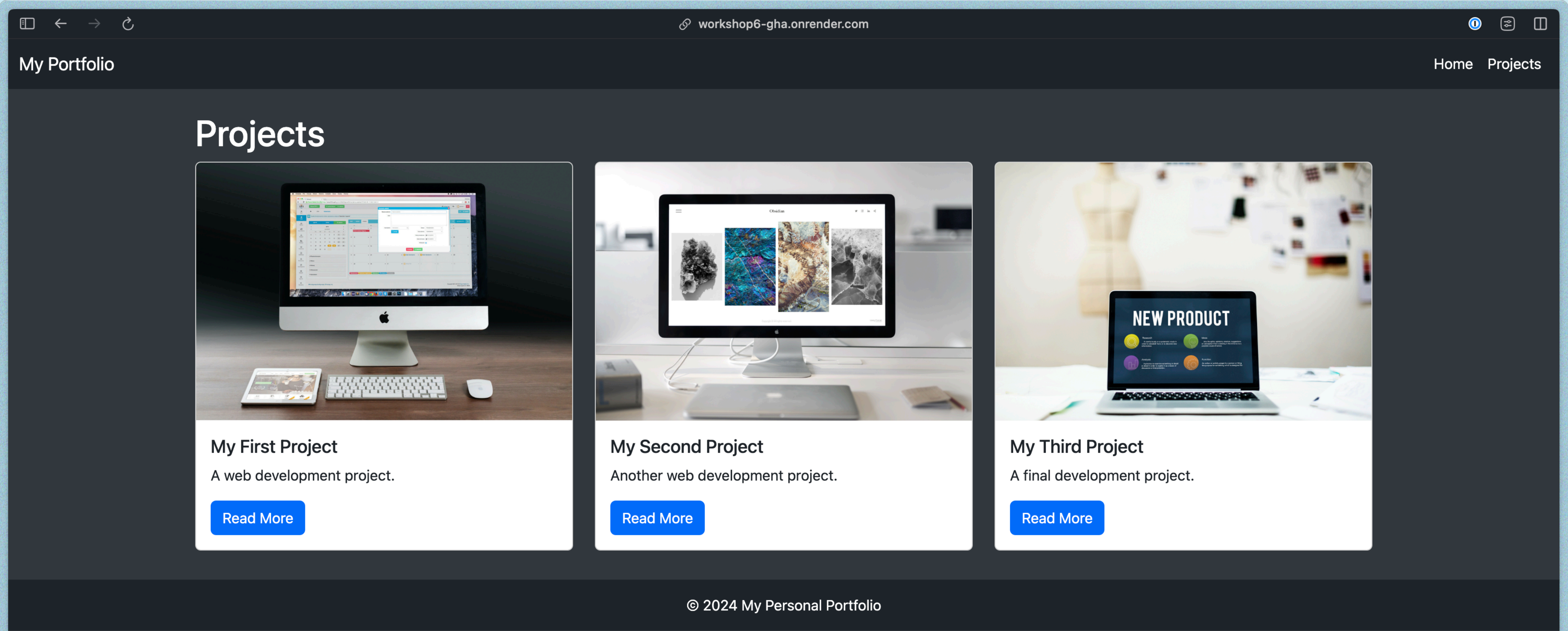
Troubleshooting GitHub Actions

tmate to troubleshoot
firasm committed 2 months ago · ✕ 0 / 1   `Verified`   7a4f5e1 ⧉ <>

update workflow to adjust
firasm committed 2 months ago · ✕ 0 / 1   7333275 ⧉ <>

what
firasm committed 2 months ago · ✕ 0 / 1   8a0dc8d ⧉ <>

-○-   Commits on Apr 30, 2024

remove tmate
firasm committed 3 months ago · ✕ 0 / 1   `Verified`   ab02d59 ⧉ <>

Add both hosts
firasm committed 3 months ago · ✕ 0 / 1   `Verified`   389597a ⧉ <>

not quite...
firasm committed 3 months ago · ✕ 0 / 1   `Verified`   c678206 ⧉ <>

Fixed, commented out tmate
firasm committed 3 months ago · ✕ 0 / 1   `Verified`   557d6e3 ⧉ <>

time for tmate
firasm committed 3 months ago · ✕ 0 / 1   `Verified`   a9318b6 ⧉ <>

Update deploy.yml
firasm committed 3 months ago · ✕ 0 / 1   `Verified`   9d7ffe6 ⧉ <>

send files to CS server
firasm committed 3 months ago · ✕ 0 / 1   49b7c19 ⧉ <>

Repository Secrets

<> Code   ⊙ Issues   ⇅ Pull requests   ▷ Actions   ⊞ Projects   ⛨ Security   ⤴ Insights   ⚙ **Settings**

⚙ General

Access

⚭ Collaborators and teams

Code and automation

⬦ Branches

🏷 Tags

⊟ Rules ⌄

▷ Actions ⌄

⤲ Webhooks

⊞ Environments

⬚ Codespaces

⊡ Pages

☰ Custom properties

Security

⊙ Code security and analysis

⚷ Deploy keys

✳ Secrets and variables ⌃

     Actions

Actions secrets and variables

Secrets and variables allow you to manage reusable configuration data. Secrets are **encrypted** and are used for sensitive data. Learn more about encrypted secrets. Variables are shown as plain text and are used for **non-sensitive** data. Learn more about variables.

Anyone with collaborator access to this repository can use these secrets and variables for actions. They are not passed to workflows that are triggered by a pull request from a fork.

| Secrets | Variables |

Environment secrets

This environment has no secrets.

Manage environment secrets

Repository secrets

New repository secret

Name ⇅↑	Last updated
🔒 RENDER_DEPLOY_HOOK_URL	1 hour ago ✏ 🗑

Q&A