

DevOps & Cloud

Linux, containers, Kubernetes and the pipelines between them. You install an app by hand first, containerise those exact steps, run it on a cluster, and ship changes without downtime.

Foundational

5 phases

20 sessions

30 contact hours

Syllabus

5 phases, 20 sessions. Each phase ends in something you have built.

1

Before Containers

Understand the machine an app runs on by running one the hard way first.

3 sessions

\$58 alone

1.1 The Shell and the Filesystem

Move around a Linux machine, read and edit files, and stop fearing the terminal.

[FREE PREVIEW](#)

1.2 Processes, Ports, and Logs

Find what is running, which port it listens on, and what it wrote to its log.

1.3 Install It By Hand

Put a web app and a database on a bare server and write down every step you needed.

By the end: **A web app and database installed by hand on a Linux server, plus written notes for every command it took.**

2

Containers

Turn a manual install into an image that runs identically on any machine.

5 sessions

\$58 alone

2.1 Images Are Not Containers

Run, list, stop, and remove containers, and explain why editing one changes nothing permanent.

2.2 Your First Dockerfile

Convert last phase's install notes into a Dockerfile that builds and runs.

2.3 Layers, Cache, and Size

Cut a 1.2 GB image down using build order, dockerignore, and a multi-stage build.

2.4 Where the Data Goes

Use volumes and bind mounts so your database survives a container restart.

2.5 Compose for the Whole App

Start a web app, a database, and a cache on one shared network from a single file.

By the end: **A multi-container app a stranger can start with one docker compose up.**

3

Ship It

Make every merge produce an image you can trust and run anywhere.

4 sessions

\$58 alone

3.1 Registries, Tags, and Digests

Push an image to GHCR, pin it by digest, and explain why latest is not a version.

3.2 Builds That Run on Any CPU

Build one image for arm64 and amd64 so your laptop build does not die on the server.

3.3 CI with GitHub Actions

Run tests and a build on every push, cache the layers, and block a merge that fails.

3.4 What Is Actually In Your Image

Scan for known CVEs, generate an SBOM, and sign what you publish.

By the end: **A repository whose main branch publishes a scanned, signed, multi-architecture image to GHCR.**

4

Clusters

Run your images across several machines and deploy by merging instead of typing.

5 sessions

\$58 alone

4.1 Kubernetes Without a Cloud Bill

Stand up a local cluster with k3d and deploy your image as a Deployment and a Service.

4.2 Reading a Broken Pod

Diagnose CrashLoopBackOff, ImagePullBackOff, and OOMKilled instead of guessing.

4.3 Config, Secrets, and Storage

Wire up ConfigMaps, real secret management, and a volume that outlives the pod.

4.4 Getting Traffic In

Route requests into the cluster with the Gateway API and see how DNS works inside it.

4.5 GitOps with Argo CD

Deploy by merging a pull request, then watch a hand-edited change get reverted.

By the end: **A local Kubernetes cluster whose entire state comes from a Git repository, reconciled by Argo CD.**

5

Cloud and Operations

Create real cloud infrastructure from code, know when it breaks, and delete it before it costs you.

3 sessions

\$58 alone

5.1 Infrastructure as Code

Create and destroy real cloud resources with OpenTofu and explain what the state file holds.

5.2 The Bill Is Not a Cap

Read a pricing page, set a budget alert, and shut an account down before it charges you.

5.3 Knowing When It Breaks

Instrument the app with OpenTelemetry and build one dashboard and one alert worth waking up for.

By the end: **A cloud environment defined in OpenTofu with one dashboard, one alert, a short runbook, and a teardown you have actually run.**

Tools you will use

Docker Engine 29

Docker Compose v2

Git + GitHub Actions

GitHub Container Registry

Kubernetes 1.36 on k3d

Argo CD 3.x

OpenTofu 1.12

OpenTelemetry + Grafana

What you will build

1 One Command Up

App, database, and cache from a single compose file

2 Merge, Build, Publish

A pipeline that ships a scanned multi-arch image

3 Delete It All

Wipe the cluster, rebuild it from Git, prove it with a dashboard

Registration

Phase 1: Before Containers	\$58
Phase 2: Containers	\$58
Phase 3: Ship It	\$58
Phase 4: Clusters	\$58
Phase 5: Cloud and Operations	\$58

Whole track, all 5 phases **\$240**

Buying every phase separately costs \$290, so the whole track saves \$50.

No payment is taken online. Registering creates an order and payment instructions are sent with it. Access opens once payment is confirmed.