



CLOUD ENGINEER PROGRAM

PROGRAM SYLLABUS

A 12-week, cohort-based program that produces a job-ready entry-level cloud engineer: weekly graded labs, a public portfolio, and a deployed capstone.

AWS-First · Infrastructure as Code · Containers · CI/CD · AI on Cloud

DURATION	LIVE HOURS	SCHEDULE	SELF-STUDY
12 WEEKS live cohort	72 HOURS instructor-led	3 DAYS / WEEK 2 hours per day	3-5 HRS / WEEK labs & portfolio

PROGRAM POSITIONING

This program produces a job-ready entry-level **Cloud / DevOps Engineer**, not a generalist who has only watched tutorials. It is built **AWS-first** because AWS has the most job listings and the fastest path to a first role, and the skills transfer cleanly to Azure and GCP later. Infrastructure as Code, containers, and CI/CD form the spine because that is the actual daily work of the job. Cloud security, cost optimization (FinOps), and deploying AI workloads are included as the modern differentiators that set our graduates apart.

TARGET FIRST ROLES

Cloud Support Engineer, Junior Cloud Engineer, Junior DevOps Engineer, Cloud / Platform Intern.

SECONDARY PATH

Freelance cloud setup, migration, and infrastructure automation for small businesses.

Honest expectation set with students up front: the first job is often a cloud support or associate role, not "architect" on day one. Students who skip the weekly labs will not be job-ready; the portfolio of deployed projects, not attendance, is what gets them hired.

LEARNING OUTCOMES

By completion, students will be able to:

- 01 Operate confidently in Linux and the cloud command line, and understand core networking
- 02 Provision core cloud infrastructure on AWS — identity (IAM), compute, storage, and networking
- 03 Design private, secure networks (VPC) and choose the right managed databases
- 04 Define entire environments as reproducible code with Terraform (Infrastructure as Code)
- 05 Containerize applications with Docker and run them on Kubernetes
- 06 Build CI/CD pipelines that automatically test and deploy on every commit
- 07 Instrument systems with monitoring, logging, and alerting, and apply SRE reliability principles
- 08 Secure cloud environments and optimize spend using FinOps practices
- 09 Deploy and operate AI workloads (LLM apps, RAG, vector databases) on the cloud
- 10 Graduate with a public GitHub portfolio, a deployed capstone, a cloud resume, and a cert roadmap

12-WEEK CURRICULUM

Three live sessions per week, each pairing theory with a graded hands-on lab.

WEEK 01 CLOUD & LINUX FOUNDATIONS

Goal: Build the shared language. You can't operate the cloud without the command line and networking basics.

DAY	TOPIC	THEORY	PRACTICAL / LAB
1	Cloud Fundamentals & Career Map	Service models (IaaS/PaaS/SaaS), shared responsibility, regions & AZs, how cloud hiring actually works	Create an AWS free-tier account; tour the console; set a billing alarm
2	Linux Essentials	Filesystem, permissions, users, processes, packages, SSH	Launch a Linux VM, connect over SSH, run core commands
3	Networking for the Cloud	TCP/IP, DNS, HTTP/S, ports, subnets, how a request reaches a server	Trace a DNS/HTTP request; diagram the path end to end

SELF-STUDY 3–5 HRS · AWS Cloud Practitioner Essentials (free) — begin.

WEEK 02 CORE CLOUD BUILDING BLOCKS

Goal: Provision the fundamental resources every architecture is built from — identity, compute, and storage.

DAY	TOPIC	THEORY	PRACTICAL / LAB
1	Identity & Access (IAM)	Users, groups, roles, policies, least privilege	Build and test least-privilege IAM users and roles
2	Compute (EC2)	Instances, types, AMIs, key pairs, security groups	Launch, connect to, and configure an EC2 web server
3	Storage (S3 & EBS)	Object vs block storage, buckets, lifecycle, encryption	Host a static site on S3; attach and use an EBS volume

SELF-STUDY 3–5 HRS · AWS free-tier compute & storage labs.

WEEK 03 NETWORKING & DATABASES

Goal: Design private, secure networks and pick the right database. Networking is the skill employers value most.

DAY	TOPIC	THEORY	PRACTICAL / LAB
1	VPC Deep Dive	Subnets, route tables, internet & NAT gateways, security groups vs NACLs	Build a custom VPC with public and private subnets
2	Load Balancing & DNS	Application load balancers, target groups, Route 53, health checks	Put a load balancer in front of two instances
3	Managed Databases	Relational (RDS) vs NoSQL (DynamoDB), backups, connections	Deploy an RDS database and connect an app to it

SELF-STUDY 3–5 HRS · Assemble the classic 3-tier architecture (this week's portfolio project).

WEEK 04 INFRASTRUCTURE AS CODE (TERRAFORM)

Goal: Stop clicking. Define everything as reproducible code — the leap from learner to real engineer.

DAY	TOPIC	THEORY	PRACTICAL / LAB
1	IaC & Terraform Basics	Providers, resources, state, variables, the plan/apply workflow	Provision an EC2 instance and an S3 bucket with Terraform
2	Reusable Infrastructure	Modules, outputs, remote state, keeping code DRY	Refactor into modules; move state to a remote backend
3	Rebuild the Stack	Turning a real architecture into version-controlled code	Recreate the Week 3 3-tier stack entirely in Terraform

SELF-STUDY 3–5 HRS · HashiCorp Terraform Associate learning path — begin.

WEEK 05

CONTAINERS WITH DOCKER

Goal: Package applications the way modern teams actually ship them.

DAY	TOPIC	THEORY	PRACTICAL / LAB
1	Container Fundamentals	Images, containers, Dockerfiles, layers, registries	Write a Dockerfile; build and run a container locally
2	Multi-Container Apps	Docker Compose, container networking, volumes	Run an app plus its database together with Compose
3	Registries & Cloud	Tagging, pushing images, running containers in the cloud	Push an image to a registry (ECR) and run it on the cloud

SELF-STUDY 3-5 HRS · Containerize a personal project and push it to GitHub.

WEEK 06

KUBERNETES

Goal: The default runtime for modern applications. Mid-program practical assessment week.

DAY	TOPIC	THEORY	PRACTICAL / LAB
1	Kubernetes Core	Pods, deployments, services, the kubectl workflow	Deploy an app to a cluster and expose it
2	Scaling & Config	Replicas, autoscaling, ConfigMaps & Secrets, health checks	Scale and self-heal a deployment
3	Managed Kubernetes (EKS)	Provisioning, ingress, exposing apps to the internet	Deploy the containerized app to EKS. Mid-program assessment due

SELF-STUDY 3-5 HRS · Continue Kubernetes practice; document the deployment.

WEEK 07

CI/CD & AUTOMATION

Goal: Ship automatically. Code push → tested → deployed, with no manual steps.

DAY	TOPIC	THEORY	PRACTICAL / LAB
1	CI/CD Concepts & Pipelines	Build/test/deploy stages, GitHub Actions, runners	Build a CI pipeline that tests on every push
2	Continuous Deployment	Environments, promotion, automated cloud deploys	Auto-deploy the app to the cloud on merge
3	Pipeline Security & GitOps	Secrets management, deploying containers through a pipeline	Full pipeline: code → image → deployed service

SELF-STUDY 3–5 HRS · Wire CI/CD into your Terraform stack.

WEEK 08

OBSERVABILITY & RELIABILITY (SRE)

Goal: Know what's happening in production and keep it running.

DAY	TOPIC	THEORY	PRACTICAL / LAB
1	Monitoring & Metrics	CloudWatch, dashboards, the metrics that matter	Build a monitoring dashboard for your app
2	Logging & Alerting	Centralized logs, alarms, on-call basics	Set up log aggregation and trigger a real alert
3	Reliability & SRE	SLIs/SLOs, incident response, autoscaling for resilience	Load-test and auto-scale; simulate and resolve an incident

SELF-STUDY 3–5 HRS · Instrument your capstone-candidate application.

WEEK 09 CLOUD SECURITY & FINOPS

Goal: Secure it and control the bill — two skills that make even a junior instantly valuable.

DAY	TOPIC	THEORY	PRACTICAL / LAB
1	Security Hardening	IAM best practices, encryption, secrets, network security	Security-audit and harden your own environment
2	Detection & Compliance	CloudTrail, GuardDuty concepts, compliance frameworks	Detect suspicious activity in cloud logs
3	FinOps & Cost Control	Cost visibility, tagging, right-sizing, savings plans	Cost-audit the account; cut spend with a written report

SELF-STUDY 3–5 HRS · Run a Well-Architected review of your project.

WEEK 10 AI ON THE CLOUD

Goal: The modern differentiator. Cloud is the engine for AI, and this is where the premium roles are.

DAY	TOPIC	THEORY	PRACTICAL / LAB
1	Running AI Workloads	GPU compute, managed model services (Bedrock/SageMaker), LLM APIs	Call a hosted LLM from your own application
2	Building an AI App (RAG)	Embeddings, vector databases, retrieval-augmented generation	Build a RAG chatbot over a custom document set
3	Deploying & Operating AI	Model endpoints, monitoring cost & latency, guardrails	Deploy the RAG app with a cost & usage dashboard

SELF-STUDY 3–5 HRS · Extend the AI app and add it to your portfolio.

WEEK 11 CAPSTONE PROJECT

Goal: Prove it. The portfolio centerpiece that replaces a meaningless certificate.

DAY	TOPIC	THEORY	PRACTICAL / LAB
1	Capstone Kickoff	Choose a track: IaC-deployed app / automated pipeline / AI-powered product	Scope and architect the capstone
2	Build & Mentorship	Guided build with live instructor feedback	Continue the build; incorporate feedback
3	End-to-End Deployment	Shipping a full system: IaC + CI/CD + monitoring + security	Complete and deploy the capstone

SELF-STUDY 3-5 HRS · Finalize the GitHub repo, README, and architecture diagrams.

WEEK 12 FINAL ASSESSMENT & JOB LAUNCH

Goal: Convert skills into interviews and offers.

DAY	TOPIC	THEORY	PRACTICAL / LAB
1	Final Practical Exam	Timed, scenario-based build-and-deploy challenge	Timed hands-on exam
2	Portfolio & Mock Interviews	Capstone presentation; technical, behavioral & basic system design	Peer mock interviews with feedback
3	Career Launch	Job-search strategy, cert roadmap, application sprint	Apply to 5 real roles; polish LinkedIn. Graduation

SELF-STUDY 3-5 HRS · Ongoing cert prep (Solutions Architect Associate) and job applications.

ASSESSMENT & GRADING

Grades are practical and continuous. Attendance alone earns nothing. The portfolio and hands-on labs are the core of the assessment.

COMPONENT	WEIGHT
Weekly lab submissions (graded pass / redo with feedback)	30%
Weekly quizzes	10%
Mid-program practical assessment (Week 6, Kubernetes deployment)	15%
Capstone project (Week 11)	25%
Final practical exam + presentation (Week 12)	20%

GRADUATION GATES: MANDATORY, PASS / FAIL

Minimum 80% attendance; a complete public portfolio (GitHub repos with Infrastructure as Code + the capstone); a live, deployed capstone project with a shareable URL; and a finished cloud resume + LinkedIn. The Edversity certificate is issued only when these are met, so it means something to an employer.

CERTIFICATION & PROOF STRATEGY

Edversity does not pay for student certifications. Instead, the program is designed so graduates leave with free, verifiable proof of skill and are fully prepared to sit paid exams later at their own choice.

FREE / NO-COST PROOF: BUILT INTO THE PROGRAM

- **Public GitHub portfolio** of Terraform, pipelines, and the capstone: Edversity's core proof artifact
- **A live, deployed capstone** with a shareable URL: third-party-verifiable, hands-on proof
- **AWS Skill Builder — Cloud Practitioner & Solutions Architect learning paths:** free official study
- **HashiCorp & freeCodeCamp cloud content:** free foundational training in IaC and cloud engineering

PAID CERTS STUDENTS CAN PURSUE LATER: WE PREPARE THEM, THEY BUY THE EXAM

AWS Certified Cloud Practitioner (CLF-C02)

Foundational, beginner-friendly cert and a strong early résumé win. Fully covered by the first weeks of the program.

~\$100

AWS Certified Solutions Architect – Associate (SAA-C03)

The most in-demand cloud cert for getting hired. Our curriculum maps directly to it; passing an AWS exam earns a 50% voucher toward the next.

~\$150

HashiCorp Certified: Terraform Associate

Validates Infrastructure as Code skills — a high-ROI, cloud-agnostic credential that gets juniors past résumé filters.

~\$70.50

TOOLS COVERED

All tools are free or free-tier, with no license cost to students.

DOMAIN	TOOLS
Cloud Platform	AWS (free tier) — EC2, S3, VPC, IAM, RDS, CloudWatch
Infrastructure as Code	Terraform
Containers & Orchestration	Docker, Docker Compose, Kubernetes (EKS)
CI/CD & Version Control	Git, GitHub, GitHub Actions
Observability	CloudWatch, Prometheus & Grafana (intro)
Security & Cost	IAM, CloudTrail, GuardDuty, AWS Cost Explorer
AI on Cloud	Amazon Bedrock, LLM APIs, a vector database
Languages	Python, Bash
Practice & Portfolio	GitHub, live-deployed projects

CAREER & JOB-READINESS OUTCOMES

Every graduate leaves with a complete, employer-facing package, not just knowledge:

- A public GitHub portfolio of Infrastructure as Code, pipelines, and projects
- A live, deployed capstone application with a shareable URL
- A cloud-specific resume and an optimized LinkedIn profile
- Mock-interview reps — technical, behavioral, and basic system design
- A personalized certification roadmap (Cloud Practitioner → Solutions Architect Associate → Terraform Associate)
- At least 5 live job or internship applications submitted before graduation

DESIGN RATIONALE

Internal note for the Edversity team.

AWS-first: AWS has the most job listings and the fastest path to a first role. Concepts are taught portably, so graduates can add Azure or GCP later with little friction.

Portfolio over certificates: the market is full of cert-holders who can't build. Tying the certificate to a deployed portfolio fixes the "meaningless certificate" problem and is what actually gets a junior hired.

IaC + containers + CI/CD as the spine: this is the actual day-to-day work of a cloud/DevOps engineer. Graduates have done the job, not just toured the console.

AI on the cloud added: cloud and AI have merged, and this is Edversity's clearest differentiator over local competitors — most beginner cloud courses skip it entirely.

Security and FinOps folded into one week: enough to make a junior valuable and cost-aware from day one, without turning the course into a dedicated security program.

Certificate tied to portfolio, not attendance: reduces the drop-out-and-blame dynamic by setting honest expectations up front — the deployed work is the proof.

Assumes ~3–5 hrs/week self-study on top of live hours. This is stated to students at enrollment; without it, no 12-week program produces a hire.