

Модуль 1 — Вступ у DevOps + AI-augmented інженер 2026

- **Заняття 1. Що таке DevOps і куди веде ринок** — DevOps як культура і практика, SDLC, CI/CD-філософія; ландшафт ролей (DevOps / SRE / Platform / MLOps / AI Infra); ринок 2026 — джуни DevOps vs дефіцит AI Infra; дві траєкторії з одного фундаменту.
- **Заняття 2. AI як напарник інженера: налаштування й практика** — як змінюється професія з AI; інструментарій (IDE-copilot, CLI-LLM); як ставити задачі AI; межа довіри; налаштування середовища.

Модуль 2 — Linux і Bash-скриптування

- **Заняття 1. Linux: файлова система, процеси, права, пакети** — навігація й файли, процеси / systemd, користувачі / права / sudo, пакетні менеджери, SSH.
- **Заняття 2. Bash-скриптування + AI-генерація скриптів** — синтаксис bash, патерни (бекапи, моніторинг, обробка логів), cron; AI генерує / пояснює / ревіює скрипт.

Модуль 3 — Мережі, HTTP і як працює веб

- **Заняття 1. Мережеві основи для DevOps** — OSI/TCP-IP, IP / порти / NAT, DNS, HTTP(S) / TLS; діагностика (ping, dig, curl, traceroute).
- **Заняття 2. Веб-сервери, reverse проху + AI-діагностика** — Nginx, reverse proxy, HTTPS, читання логів; AI-аналіз виводу curl / логів для гіпотези про проблему.

♦ Модуль 4 — Git: командна робота з кодом

- **Заняття 1. Git-модель і базовий воркфлоу** — коміти / staging / гілки, віддалені репозиторії, .gitignore, теги, релізи.
- **Заняття 2. Git flow, PR і code review + AI** — Git flow / trunk-based, pull request, code review, вирішення конфліктів; AI як ревіює PR.

♦ Модуль 5 — Python для DevOps

- **Заняття 1. Python-мінімум для інженера** — синтаксис, структури даних, функції, робота з файлами, JSON/YAML, venv і пакети.
- **Заняття 2. Автоматизація й API + AI-generated utilities** — робота з HTTP / cloud API, утиліти автоматизації, базові тести; AI генерує утиліти (з перевіркою).

Модуль 6 — Хмара: базові концепції (AWS core)

- **Заняття 1. Хмарна модель і AWS core** — IaaS/PaaS/SaaS, регіони / зони; EC2, S3, IAM (мінімум прав), VPC / security groups; білінг і costs.

- **Заняття 2. Ручний деплой у хмару + AI-навігація** — розгортання на EC2, S3, IAM-доступи, базова безпека; AI-асистент як гід по сервісах і оцінювач costs.

→ **Проект 1. Базова інфраструктура в хмарі** (*Linux + Nginx + Git + AWS*)

Блок 2 — Контейнери, оркестрація, автоматизація

Модуль 7 — Контейнеризація: Docker

- **Заняття 1. Docker: образи, контейнери, Compose** — образи / шари / кеш, Dockerfile, Docker Compose, мережі / томи, реєстри.
- **Заняття 2. Оптимізація образів + AI-Dockerfile + контейнер ML-моделі** — multi-stage, безпека, сканування; AI генерує / оптимізує Dockerfile; контейнеризація ML-моделі як міст до Блоку 3.

Модуль 8 — Infrastructure as Code: Terraform

- **Заняття 1. IaC і Terraform: основи** — провайдери / ресурси / state, plan / apply / destroy, змінні / outputs, модулі, remote state, costs-дисципліна.
- **Заняття 2. Terraform для реальної інфра + AI-генерація** — інфра під застосунок, заготовка під EKS, секрети; AI генерує й ревіює Terraform.

Модуль 9 — Kubernetes: основи

- **Заняття 1. Архітектура Kubernetes** — control plane / ноди, pod / deployment / service, namespaces / labels, declarative model; локальний кластер (kind / minikube).
- **Заняття 2. kubectl, конфігурація, storage + AI-діагностика подів** — kubectl і YAML, ConfigMap / Secret, storage, probes; AI-асистент для діагностики (CrashLoopBackOff).

Модуль 10 — Деплой у Kubernetes (EKS) + Helm

- **Заняття 1. Керований Kubernetes (EKS), масштабування, ingress** — EKS, ingress, автоскейл (HPA), requests / limits, costs кластера.
- **Заняття 2. Helm: чарти й релізи + AI-генерація чартів** — чарти / templates / values, релізи / rollback, repositories; AI генерує й ревіює чарти.

♦ Модуль 11 — CI/CD: GitLab CI

- **Заняття 1. Основи CI/CD і пайплайн GitLab CI** — стадії / jobs / runners, `.gitlab-ci.yml`, артефакти, змінні / секрети.

- **Заняття 2. Повний пайплайн build→test→deploy + AI** — збірка образу, тести в пайплайні, деплой; AI генерує конфіг і аналізує падіння job.

♦ Модуль 12 — GitOps: ArgoCD

- **Заняття 1. Принцип GitOps і ArgoCD** — git як єдине джерело правди, pull vs push деплой, встановлення й підключення ArgoCD.
- **Заняття 2. Декларативний деплой, синхронізація, rollback** — синхронізація кластера з репо, drift-detection, rollback, деплой Helm-чартів через ArgoCD.

Модуль 13 — Моніторинг і логування + AIOps

- **Заняття 1. Метрики, Prometheus, alerting** — метрики / логи / трейси, Prometheus + PromQL, exporters, alerting, golden signals.
- **Заняття 2. Grafana, логування + AIOps** — дашборди, логування (Loki / ELK), runbook; AIOps — AI-аналіз логів і метрик до гіпотези про причину інциденту.

→ **Проект 2. Класичний DevOps end-to-end** (*Docker → Terraform → K8s/Helm → CI/CD/ArgoCD → моніторинг · портфоліо Junior DevOps*)

Блок 3 — Інфраструктура для AI

Модуль 14 — Вступ у MLOps + MLflow

- **Заняття 1. Життєвий цикл ML-моделі й роль MLOps** — життєвий цикл (дані→тренування→inference→деградація), чим ML-система відрізняється, ролі MLOps / AI Infra, технічний борг ML.
- **Заняття 2. Experiment tracking і Model Registry: MLflow** — MLflow (параметри / метрики / артефакти), Model Registry (версії / статуси), версіонування, порівняння експериментів.

Модуль 15 — Деплой ML/inference-сервісів у K8s

- **Заняття 1. Inference-сервіс: від моделі до API в кластері** — Model Serving (REST / gRPC), контейнеризація моделі, деплой у K8s, health checks, логування передбачень.
- **Заняття 2. Масштабування inference й стратегії деплою** — автоскейл, deployment strategies (canary / blue-green / shadow), оновлення без простою, CPU vs GPU.

♦ Модуль 16 — LLMOps ч.1: serving LLM

- **Заняття 1. Чим LLM-навантаження відрізняється + serving-пуші** — GPU vs CPU, латентність / вартість, vLLM / Triton, batching, KV-cache (рівень розуміння).
- **Заняття 2. Розгортання LLM-serving + cost-aware inference** — базове розгортання serving (мала модель / локальний fallback), inference optimization оглядово, керування costs GPU.

♦ Модуль 17 — LLMOps ч.2: RAG + vector DB

- **Заняття 1. RAG і vector DB: концепція та інфраструктура** — обмеження «голоного» LLM, embeddings, vector DB, що деплоїть AI Infra-інженер.
- **Заняття 2. Побудова RAG-pipeline + моніторинг** — індексація → пошук → генерація, збірка мінімального RAG, моніторинг і вартість системи.

♦ Модуль 18 — Моніторинг якості моделей + дрейф

- **Заняття 1. Якість моделі в проді vs health** — метрики якості окремо від health, збір передбачень, зв'язок із Prometheus / Grafana.
- **Заняття 2. Data / concept drift і реакція** — детекція дрейфу, коли донавчати / замінювати, автоматизоване донавчання оглядово.

♦ Модуль 19 — Безпека ML-інфраструктури (MLOpsSec)

- **Заняття 1. Загрози ML-систем** — отруєння даних, крадіжка / інверсія моделі, adversarial-атаки (оглядово), специфіка LLM / RAG (витік через промпти).
- **Заняття 2. Захист інфраструктури й inference-API** — безпека API (доступи, rate limiting, валідація вводу), захист моделей / даних, секрети в ML-пайплайнах, чекліст безпеки.

Модуль 20 — Власний AI-агент для інфраструктури

- **Заняття 1. AI-агенти для DevOps: патерни й архітектура** — агент vs асистент, патерни (ChatOps, аналіз метрик / логів, асистент деплою, incident-помічник), архітектура (LLM + інструменти + доступ до kubectl / Prometheus), **безпечні межі (read-only / людина в циклі)**.
- **Заняття 2. Побудова агента + збірка фінального проєкту** — практична збірка агента, обмеження прав, інтеграція в наскрізну інфра, збірка й підготовка до захисту капстону.

→ **Проект 3. AI Infrastructure end-to-end (ML/LLM-serving + MLflow + моніторинг якості + власний AI-агент · портфоліо AI Infra/MLOps)**