

## CI/CD или Непрерывная поставка с Docker и Kubernetes

**Код:** PTRN-043

**Длительность:** 16 ч.

### Описание:

В рамках курса слушатели узнают, как внедрить практику непрерывной поставки в команду, чтобы ускорить процесс выкатки и исключить простои. Участники настроят автоматизированный конвейер поставки микросервисного приложения в Kubernetes, научатся управлять всем жизненным циклом приложения. Курс является продолжением курса "[Введение в Docker](#)".

После прохождения курса слушатели смогут:

- Сделать выкатку кода простым и приятным процессом в духе "нажми одну кнопку", который можно выполнять десятки раз в день;
- Сделать сетап новой машины полностью автоматизированным: вы запускаете одну команду, выпиваете чашку кофе и, вуаля, полностью готовый к работе сервер;
- Создать прозрачный и понятный Continuous Delivery Pipeline и выкатывать изменения часто и небольшими частями;
- Делать десятки тестовых стендов в день для разных команд и разных версий продукта, при этом конфигурация стенда будет практически идентичной боевой среде;
- Увеличить скорость разработки, тестирования и выкатки за счет применения современной системы автоматизации развертывания контейнеров Docker;
- Развернуть масштабируемую инфраструктуру для запуска микросервисов;
- Получать быструю обратную связь от систем мониторинга и логирования на всех основных этапах жизненного цикла ПО.

### Цели:

Научить участников разворачивать инфраструктуру проекта, настраивать автоматизированный конвейер поставки ПО и использовать системы получения быстрой обратной связи о состоянии проекта. По окончании курса участники смогут управлять всем жизненным циклом приложения.

### Разбираемые темы:

1. Методология DevOps.
  - Развитие подходов в управлении ИТ-инфраструктурой, путь к DevOps, CAMS (Culture, Automation, Measurement, Knowledge Sharing), подходы к автоматизации, работа в команде, основные DevOps практики;
  - Чему учит DevOps и откуда такая популярность.
2. Технология непрерывной поставки ПО.
  - Continuous Integration, Continuous Delivery, тестирование, управление конфигурацией, виртуализация, контейнеризация;
  - Автоматизация сборок, тестирование Docker контейнеров. Практика:
  - Построение базового CI/CD процесса на Google Cloud Platform (GCP), выкатка новой версии приложения, решения часто возникающих проблем.
3. Оркестрация и стратегии деплоя:
  - Что такое оркестрация, преимущества и недостатки актуальных оркестраторов;
  - Введение в Kubernetes, основные компоненты;
  - Микросервисная архитектура, преимущества и недостатки сервисного подхода;
  - Стратегии деплоя микросервисов;
  - Сетевое взаимодействие в Docker кластере: overlay network, Docker ipvs load balancer. Практика:
  - Создание и настройка Kubernetes кластера на GCP;
  - Интеграция Kubernetes с Gitlab CI.
4. Непрерывная обратная связь (мониторинг и логирование).
  - Как получать обратную связь от проекта;
  - Что и зачем нужно мониторить в проекте;
  - Сбор метрик инфраструктуры и бизнес логики приложений, end-user experience мониторинг;
  - Поставка систем сбора метрик как сервис внутри проекта, разделение ответственности между разработкой и эксплуатацией;
  - Обзор актуальных систем мониторинга и логирования;
  - Особенности сбора метрик с микросервисов и Docker контейнеров. Практика:
  - Развертывание системы мониторинга Prometheus и системы логирования Elastic Stack;
  - Настройка централизованного сбора метрик с Docker контейнеров;
  - Получение обратной связи при деплое, анализ собранных метрик с помощью Grafana.

### **Целевая аудитория:**

Системные администраторы, инфраструктурные инженеры, разработчики, продвинутые тестировщики.

### **Предварительная подготовка - общее:**

Базовое знание ОС Linux. Для прохождения этого курса потребуются начальные знания Docker и Docker Compose. Эти знания вы можете получить как самостоятельно, так и на нашем курсе [“Введение в Docker”](#).