

DevOps практики и инструменты

DevOps решает задачу взаимодействия инженеров между собой с целью помочь им быстрее создавать и обновлять сервисы и приложения.

Общая стоимость
70 000

В месяц: 17 000

Продолжительность
5 месяцев

4 часа в неделю

Начало занятий
27 ноября

1 DevOps. Необходимость меняться

- 1 Что такое DevOps и история его развития.**
Как DevOps меняет жизнь и работу компаний. DevOps как профессия. Обзор курса.
- 2 DevOps как система.**
Практики и методики. DevOps компетенции.
- 3 Система контроля версий как основа разработки и поставки ПО. Знакомство с Git.**

4 **Локальное окружение инженера. ChatOps и визуализация рабочих процессов. Командная работа с Git. Работа в GitHub.**

5 **Знакомство с облачной инфраструктурой и облачными сервисами.**

Практики безопасного доступа к ресурсам (SSH, Bastion Host, VPN).

6 **Основные сервисы Google Cloud Platform (GCP).**

Способы управления ресурсами в GCP.

2 Управление инфраструктурой и конфигурацией

1 **Модели управления инфраструктурой.**

Работа с образами VM в облаке. Знакомство с Packer и экосистемой компании HashiCorp.

2 **Практика Infrastructure as a Code (IaC).**

Знакомство с Terraform.

3 **Принципы организации инфраструктурного кода и работа над инфраструктурой в команде на примере Terraform.**

4 **Управление конфигурацией.**

Основные DevOps инструменты: Chef, Ansible, Puppet, Salt. Знакомство с Ansible.

5 **Продолжение знакомства с Ansible: templates, handlers, dynamic inventory, vault, tags.**

6 **Принципы организации кода для управления конфигурацией.**

Ansible роли, управление настройками нескольких окружений и best practices.

7 **Локальная разработка Ansible ролей с Vagrant. Тестирование конфигурации.**

3 Continuous Integration & Continuous Delivery с использованием Docker

1 **Технология контейнеризации. Введение в Docker.**

2 **Docker контейнеры. Docker под капотом**

3 **Docker образы. Микросервисы**

4 **Сетевое взаимодействие Docker контейнеров. Docker Compose. Тестирование образов**

5 **Технология непрерывной поставки ПО**

**6 Устройство
Gitlab CI.
Построение
процесса
непрерывной
интеграции**

**7 Непрерывная
поставка с
Gitlab CI**

4 Fast Feedback Loop (мониторинг и логирование)

**1 Введение в
мониторинг.
Модели и
принципы
работы систем
мониторинга**

**2 Мониторинг
приложения и
инфраструктуры**

**3 Визуализация и
анализ
результатов
мониторинга.
Знакомство с
системами
рассылки
оповещений,
Алертинг**

4 **Применение инструментов для обработки лог данных** Что такое логи и зачем они нужны? Системы централизованного хранения лог данных, требования к ним и примеры. Форматы и варианты логирования приложений

5 **Применение системы логирования в инфраструктуре на основе Docker.** Рассмотрение систем распределенной отладки данных (Distributed tracing)

5 Контейнерная Оркестрация

1 **Контейнерная оркестрация.** Примеры известных и актуальных систем оркестрации (DC/OS, Aurora, Nomad, Docker Swarm)

2 **Введение в Kubernetes.** Основные концепции и архитектура системы.

3 **Основные модели безопасности и контроллеры в Kubernetes**

4 **Ingress-контроллеры и сервисы в Kubernetes.** Работа с удаленными хранилищами данных.

- 5 **Интеграция Kubernetes в GitlabCI.** Использование формата упаковки приложений Charts в Kubernetes Helm.
-
- 6 **Kubernetes. Мониторинг и логирование** Развертывание Prometheus в Kubernetes. Настройка Prometheus и Grafana для сбора метрик кластера и приложений. Развертывание и настройка EFK для сбора логов.