

**Федеральное государственное автономное образовательное
учреждение высшего образования
«Московский физико-технический институт
(национальный исследовательский университет)»**

УТВЕРЖДЕНО
Директор

А.В. Малеев

	Рабочая программа дисциплины (модуля)
по дисциплине:	Введение в технологии DevOps
по направлению:	Программная инженерия
профиль подготовки:	Разработка программно-информационных систем высшая школа программной инженерии высшая школа программной инженерии МФТИ - Яндекс
курс:	2
квалификация:	бакалавр

Семестр, формы промежуточной аттестации: 4 (весенний) - Дифференцированный зачет

Аудиторных часов: 60 всего, в том числе:

лекции: 30 час.

семинары: 30 час.

лабораторные занятия: 0 час.

Самостоятельная работа: 48 час.

Всего часов: 108, всего зач. ед.: 3

Программу составил: А.В. Малеев, директор

Программа обсуждена на заседании высшей школы программной инженерии МФТИ - Яндекс 08.06.2022

Аннотация

Данная дисциплина направлена на изучение особенностей внедрения культуры и практик автоматизации при разработке программного обеспечения. По ходу освоения программы курса студенты получают представление об основных процессах DevOps при разработке программного обеспечения, а также практические навыки применения открытых технологий.

1. Цели и задачи

Цель дисциплины

- обучение студентов базовым аспектам разработки программного обеспечения для UNIX-подобных операционных систем, а также отработки навыков написания программ и их тестирования в предельных ситуациях.

Задачи дисциплины

- получение представления об основных принципах DevOps;
- изучение микросервисной архитектуры, систем управления конфигурациями, инструментов обеспечения ресурсами, методов развертывания и настройки кластеров и практик DevOps при разработке программного обеспечения;
- освоение практические инструменты для применения практик DevOps при разработке программного обеспечения и в машинном обучении.

2. Перечень формируемых компетенций

Освоение дисциплины направлено на формирование следующих компетенций:

Код и наименование компетенции	Индикаторы достижения компетенции
УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.2 Находит, критически анализирует и выбирает информацию, необходимую для решения поставленной задачи
	УК-1.3 Рассматривает различные варианты решения задачи, оценивает их преимущества и недостатки
	УК-1.4 Грамотно, логично, аргументированно формирует собственные суждения и оценки
	УК-1.5 Определяет и оценивает практические последствия возможных вариантов решения задачи
УК-6 Способен управлять своим временем, выстраивать и реализовывать траекторию саморазвития на основе принципов образования в течение всей жизни	УК-6.2 Способен планировать самостоятельную деятельность в решении профессиональных задач; подвергать критическому анализу проделанную работу; находить и творчески использовать имеющийся опыт в соответствии с задачами саморазвития
ОПК-1 Способен применять естественнонаучные и общетехнические знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности	ОПК-1.2 Способен строить математические модели, производить количественные расчеты и оценки
	ОПК-1.3 Способен определять границы применимости полученных результатов
ОПК-6 Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов	ОПК-6.1 Знает алгоритмы решения типовых задач, области и способы их применения
	ОПК-6.3 Знает методы тестирования программного кода на ошибки и способен проводить тестирование на различных уровнях (модульное, интеграционное, системное)
ПК-2 Способен формализовать и алгоритмизировать поставленную задачу	ПК-2.1 Способен формализовать и алгоритмизировать поставленную задачу
	ПК-2.2 Владеет методами и приемами формализации и алгоритмизации задач

ПК-3 Способен проектировать, разрабатывать, интегрировать, проверять на работоспособность программное обеспечение	ПК-3.3 Умеет излагать основные принципы построения и виды архитектуры программного обеспечения, методы и средства проектирования программного обеспечения, методологии разработки программного обеспечения и технологии программирования
---	--

3. Перечень планируемых результатов обучения по дисциплине (модулю)

В результате освоения дисциплины обучающиеся должны

знать:

- определение и основные принципы DevOps;
- отличие системного администрирования от практик DevOps;
- основные понятия и компоненты микросервисной архитектуры;
- цели и необходимость систем управления конфигурациями;
- отличия инструментов обеспечения ресурсами от систем управления конфигурациями;
- кластеры и его составные части.

уметь:

- применять паттерны и антипаттерны DevOps-культуры;
- применять методы управления микросервисной архитектурой;
- применять в практике разработки ПО инструменты настройки систем управления конфигурациями;
- применять технологии развертывания серверов, балансировщиков нагрузки и проверки соединения;
- применять методы развертывания и настройки кластеров, их масштабирования и обновления сервисов.

владеть:

- навыками управления задачами и API микросервиса;
- навыками настройки систем управления конфигурациями, использования инструментов обеспечения ресурсами, развертывания и настройки кластеров;
- навыками применения практик и инструментов DevOps при разработке программного обеспечения и в проектах машинного обучения.

4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

4.1. Разделы дисциплины (модуля) и трудоемкости по видам учебных занятий

№	Тема (раздел) дисциплины	Трудоемкость по видам учебных занятий, включая самостоятельную работу, час.			
		Лекции	Семинары	Лаборат. работы	Самост. работа
1	Определение и основные принципы DevOps.	2	2		4
2	Микросервисная архитектура.	4	4		6
3	Системы управления конфигурацией.	4	4		6
4	Инструменты обеспечения ресурсами.	6	6		8
5	Кластеры. Развертывание и настройка.	6	6		8
6	Особенности применения практик DevOps в машинном обучении.	4	4		8

7	Инструменты для применения практик DevOps в машинном обучении.	4	4		8
Итого часов		30	30		48
Подготовка к экзамену		0 час.			
Общая трудоёмкость		108 час., 3 зач.ед.			

4.2. Содержание дисциплины (модуля), структурированное по темам (разделам)

Семестр: 4 (Весенний)

1. Определение и основные принципы DevOps.

Определение DevOps в широком и узком смысле. Три принципа DevOps. Отличие системного администрирования от практика DevOps. Паттерны и антипаттерны внедрения DevOps-культуры.

2. Микросервисная архитектура.

Микросервисная архитектура. Понятие балансировщика нагрузки. Автоматический запуск микросервисов: systemd units, supervisor - сходства и различия. Запуск задач по расписанию: crontab, применение запуска задач по расписанию с веб-фреймворками. Задание API микросервиса при помощи конфигурационного документа (OpenAPI).

3. Системы управления конфигурацией.

Системы управления конфигурацией: цели и необходимости. Идемпотентность и сходимост как главные свойства систем управления конфигурацией. Системы типа pull и push для систем управления конфигурации - достоинства и недостатки. Декларативный и процедурный типы конфигурации - различия, достоинства и недостатки. Популярные инструменты настройки систем управления конфигурации и их категоризация.

4. Инструменты обеспечения ресурсами.

Инструменты обеспечения ресурсами: отличие от систем управления конфигурациями в терминах категоризации систем управления конфигурации. Развертывание серверов, балансировщиков нагрузки и проверки состояния соединения.

5. Кластеры. Развертывание и настройка.

Кластер Kubernetes и его составные части: под, развертывание, сервисы, аналогия с концепцией контейнеризации и настройке сетевого соединения. Развертывание и настройка кластера на примере технологии kubespray. Масштабирование, обновление сервисов и развертываний. Пакетный менеджер для настройки kubernetes-кластера. Отличия практик kubernetes в создании ресурсов, сохраняющих и не сохраняющих состояние.

6. Особенности применения практик DevOps в машинном обучении.

Особенности применения практик DevOps в машинном обучении. Три стадии эволюции инфраструктуры машинного обучения: отличие, лучшие практики для достижения каждого уровня. Отличие практик непрерывного развертывания и непрерывной интеграции для проектов машинного обучения. Практика непрерывного обучения моделей в машинном обучении: причины появления, преимущества.

7. Инструменты для применения практик DevOps в машинном обучении.

Инструменты для применения практик DevOps в машинном обучении. Система контроля данных: регистрация файлов при помощи системы контроля версий, физическая модель хранения данных, импорт данных из сторонних репозиториях. Совместная модель разработки при использовании системы контроля данных. Сборка графа обучения моделей, активация переобучения моделей при помощи системы контроля данных. Универсальные форматы хранения обученных моделей. Использование инструментов логирования и мониторинга для создания журнала экспериментов.

5. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

Учебная аудитория, оснащенная компьютерной техникой с подключением к сети «Интернет» и мультимедийным оборудованием (проектор, звуковая система).

6. Перечень рекомендуемой литературы

Основная литература

1. Введение в прикладное дискретное программирование: модели и вычислительные алгоритмы [Электронный ресурс], учеб. пособие / И. Х. Сигал, А. П. Иванова. — М., Физматлит, 2007.— URL: <https://e.lanbook.com/book/2312> (дата обращения: 15.01.2021). - Полный текст (Режим доступа : из сети МФТИ / Удаленный доступ)
2. Практика и теория программирования [Текст] : в 2 кн. : учеб. пособие для вузов / Н. А. Винокуров, А. В. Ворожцов .— М. : Физматкнига, 2008 .— (Серия "Информатика"). - ISBN 978-5-89155-182-4 (в пер.) .— Кн.2, Ч. 3-4. - 2008. - 288 с.

-

Дополнительная литература

Литература, рекомендованная для самостоятельного изучения:

Грувер Гэри. Запуск и масштабирование DevOps на предприятии. - Москва : ДМК Пресс, 2018. - 80 с.

7. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины (модуля)

Не используются

8. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень необходимого программного обеспечения и информационных справочных систем (при необходимости)

На лекционных занятиях используются мультимедийные технологии, включая демонстрацию презентаций. Для контроля и коррекции знаний, обучающиеся могут использовать компьютерное тестирование.

9. Методические указания для обучающихся по освоению дисциплины (модуля)

Программа курса в разделе «самостоятельная работа» обозначает минимально необходимое время для работы студента над темой. Самостоятельная работа включает в себя:

- проработку учебного материала (по конспектам лекций, учебной и научной литературе), подготовку ответов на вопросы, предназначенных для самостоятельного изучения, доказательство отдельных утверждений, свойств;
- выполнение домашних заданий.

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

по направлению:	Программная инженерия
профиль подготовки:	Разработка программно-информационных систем высшая школа программной инженерии МФТИ - Яндекс высшая школа программной инженерии
курс:	2
квалификация:	бакалавр

Семестр, формы промежуточной аттестации: 4 (весенний) - Дифференцированный зачет

Разработчик: А.В. Малеев, директор

1. Компетенции, формируемые в процессе изучения дисциплины

Код и наименование компетенции	Индикаторы достижения компетенции
УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.2 Находит, критически анализирует и выбирает информацию, необходимую для решения поставленной задачи
	УК-1.3 Рассматривает различные варианты решения задачи, оценивает их преимущества и недостатки
	УК-1.4 Грамотно, логично, аргументированно формирует собственные суждения и оценки
	УК-1.5 Определяет и оценивает практические последствия возможных вариантов решения задачи
УК-6 Способен управлять своим временем, выстраивать и реализовывать траекторию саморазвития на основе принципов образования в течение всей жизни	УК-6.2 Способен планировать самостоятельную деятельность в решении профессиональных задач; подвергать критическому анализу проделанную работу; находить и творчески использовать имеющийся опыт в соответствии с задачами саморазвития
ОПК-1 Способен применять естественнонаучные и общетехнические знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности	ОПК-1.2 Способен строить математические модели, производить количественные расчеты и оценки
	ОПК-1.3 Способен определять границы применимости полученных результатов
ОПК-6 Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов	ОПК-6.1 Знает алгоритмы решения типовых задач, области и способы их применения
	ОПК-6.3 Знает методы тестирования программного кода на ошибки и способен проводить тестирование на различных уровнях (модульное, интеграционное, системное)
ПК-2 Способен формализовать и алгоритмизировать поставленную задачу	ПК-2.1 Способен формализовать и алгоритмизировать поставленную задачу
	ПК-2.2 Владеет методами и приемами формализации и алгоритмизации задач
ПК-3 Способен проектировать, разрабатывать, интегрировать, проверять на работоспособность программное обеспечение	ПК-3.3 Умеет излагать основные принципы построения и виды архитектуры программного обеспечения, методы и средства проектирования программного обеспечения, методологии разработки программного обеспечения и технологии программирования

2. Показатели оценивания компетенций

В результате изучения дисциплины «Введение в технологии DevOps» обучающийся должен:

знать:

- определение и основные принципы DevOps;
- отличие системного администрирования от практик DevOps;
- основные понятия и компоненты микросервисной архитектуры;
- цели и необходимость систем управления конфигурациями;
- отличия инструментов обеспечения ресурсами от систем управления конфигурациями;
- кластеры и его составные части.

уметь:

- применять паттерны и антипаттерны DevOps-культуры;
- применять методы управления микросервисной архитектурой;
- применять в практике разработки ПО инструменты настройки систем управления конфигурациями;
- применять технологии развертывания серверов, балансировщиков нагрузки и проверки соединения;
- применять методы развертывания и настройки кластеров, их масштабирования и обновления сервисов.

владеть:

- навыками управления задачами и API микросервиса;
- навыками настройки систем управления конфигурациями, использования инструментов обеспечения ресурсами, развертывания и настройки кластеров;
- навыками применения практик и инструментов DevOps при разработке программного обеспечения и в проектах машинного обучения.

3. Перечень типовых (примерных) вопросов, заданий, тем для подготовки к текущему контролю

С целью контроля освоения обучающимися учебного материала проводится устный опрос в начале занятия по теме прошлого занятия.

4. Перечень типовых (примерных) вопросов и тем для проведения промежуточной аттестации обучающихся

Примерный перечень контрольных вопросов:

1. Дайте определение DevOps в широком и узком смысле
2. Опишите основные принципы DevOps.
3. Объясните, в чем отличие системного администрирования от практики DevOps?
4. Приведите преимущества и недостатки методологии DevOps.
5. Перечислите наиболее распространенные паттерны внедрения DevOps-культуры.
6. Как осуществляется задание API микросервиса при помощи конфигурационного документа?
7. Объясните различия, достоинства и недостатки декларативного и процедурного типа конфигурации.
8. Приведите и опишите основные компоненты системы контроля данных.
9. Опишите цели и необходимость систем управления конфигурациями.
10. Опишите совместную модель разработки при использовании системы контроля данных.

Критерии оценивания

- оценка «отлично (10)» выставляется студенту, показавшему всесторонние, систематизированные, глубокие знания учебной программы дисциплины и умение уверенно применять их на практике при решении конкретных задач, свободное и правильное обоснование принятых решений
- оценка «отлично (9)» выставляется студенту, показавшему всесторонние, систематизированные, глубокие знания учебной программы дисциплины и умение применять их на практике при решении конкретных задач, свободное и правильное обоснование принятых решений
- оценка «отлично (8)» выставляется студенту, показавшему всесторонние систематизированные, глубокие знания учебной программы дисциплины и умение применять их на практике при решении конкретных задач, и правильное обоснование принятых решений
- оценка «хорошо (7)» выставляется студенту, если он твердо знает материал, грамотно и по существу излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач некоторые неточности;
- оценка «хорошо (6)» выставляется студенту, если он знает материал, грамотно и по существу излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач некоторые неточности;

- оценка «хорошо (5)» выставляется студенту, если он знает материал, и по существу излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач некоторые неточности;
- оценка «удовлетворительно (4)» выставляется студенту, показавшему фрагментарный, разрозненный характер знаний, недостаточно правильные формулировки базовых понятий, нарушения логической последовательности в изложении программного материала, но при этом он владеет основными разделами учебной программы, необходимыми для дальнейшего обучения и может применять полученные знания по образцу в стандартной ситуации;
- оценка «удовлетворительно (3)» выставляется студенту, показавшему фрагментарный, разрозненный характер знаний, недостаточно правильные формулировки базовых понятий, нарушения логической последовательности в изложении программного материала, но при этом он владеет фрагментарно основными разделами учебной программы, необходимыми для дальнейшего обучения и может применять полученные знания по образцу в стандартной ситуации;
- оценка «неудовлетворительно (2)» выставляется студенту, который не знает большей части основного содержания учебной программы дисциплины, допускает грубые ошибки в формулировках основных понятий дисциплины и не умеет использовать полученные знания при решении типовых практических задач
- оценка «неудовлетворительно (1)» выставляется студенту, который не знает формулировок основных понятий дисциплины

5. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

Дифференцированный зачет проводится по итогам текущей успеваемости и сдачи заданий и других видов работ, предусмотренных программой дисциплины и (или) путем организации специального опроса, проводимого в устной и (или) письменной форме. Во время проведения дифференцированного зачета обучающиеся могут пользоваться программой дисциплины, а также справочной литературой, конспектами лекций или другими материалами.