МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА»
(РУТ (МИИТ)

Рабочая программа дисциплины (модуля),
как компонент образовательной программы
высшего образования - программы бакалавриата
по направлению подготовки
09.03.01 Информатика и вычислительная техника,
утвержденной первым проректором РУТ (МИИТ)
Тимониным В.С.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Управление жизненным циклом разработки программного обеспечения

Направление подготовки:	09.03.01 Информатика и вычислительная техника
Направленность (профиль):	Технологии разработки программного обеспечения
Форма обучения:	Очная

Рабочая программа дисциплины (модуля) в виде электронного документа выгружена из единой корпоративной информационной системы управления университетом и соответствует оригиналу

Простая электронная подпись, выданная РУТ (МИИТ)
ID подписи: 5665
Подписал: заведующий кафедрой Нутович Вероника Евгеньевна
Дата: 24.05.2022

1. Общие сведения о дисциплине (модуле).

Целями освоения данной дисциплины являются получение базовых, теоретических знаний в области автоматизации процессов жизненного цикла разработки программного обеспечения.

Основной задачей изучения дисциплины является формирование базовых представлений, знаний и умений в области организации инфраструктуры с различными окружениями для непрерывной интеграции, непрерывного развертывания с использованием технологий контейнеризации.

2. Планируемые результаты обучения по дисциплине (модулю).

Перечень формируемых результатов освоения образовательной программы (компетенций) в результате обучения по дисциплине (модулю):

ПК-9 - Способен понимать стандарты и модели жизненного цикла разработки программного обеспечения .

Обучение по дисциплине (модулю) предполагает, что по его результатам обучающийся будет:

Знать:

- типовые среды окружения, их особенности и роль в жизненном цикле разработки программного обеспечения;
- основные принципы и функциональные возможности технологий контейнеризации и кластеризации для обеспечения автоматизации жизненного цикла разработки программного обеспечения.

Уметь:

- применять инструменты мониторинга и визуализации для контроля процессов жизненного цикла разработки программного обеспечения;
- применять технологии контейнеризации для автоматизации процессов непрерывного развертывания.

Владеть:

- навыками конфигурации сред окружения и инфраструктуры с использованием Ansible и Terraform;
- навыками построения инфраструктуры и автоматизации жизненного цикла разработки программного обеспечения с использованием Jenkins.

3. Объем дисциплины (модуля).

3.1. Общая трудоемкость дисциплины (модуля).

Общая трудоемкость дисциплины (модуля) составляет 3 з.е. (108 академических часа(ов).

3.2. Объем дисциплины (модуля) в форме контактной работы обучающихся с педагогическими работниками и (или) лицами, привлекаемыми к реализации образовательной программы на иных условиях, при проведении учебных занятий:

Тип учебных занятий	Количество часов	
	Всего	Сем. №8
Контактная работа при проведении учебных занятий (всего):	64	64
В том числе:		
Занятия лекционного типа	32	32
Занятия семинарского типа	32	32

3.3. Объем дисциплины (модуля) в форме самостоятельной работы обучающихся, а также в форме контактной работы обучающихся с педагогическими работниками и (или) лицами, привлекаемыми к реализации образовательной программы на иных условиях, при проведении промежуточной аттестации составляет 44 академических часа (ов).

3.4. При обучении по индивидуальному учебному плану, в том числе при ускоренном обучении, объем дисциплины (модуля) может быть реализован полностью в форме самостоятельной работы обучающихся, а также в форме контактной работы обучающихся с педагогическими работниками и (или) лицами, привлекаемыми к реализации образовательной программы на иных условиях, при проведении промежуточной аттестации.

4. Содержание дисциплины (модуля).

4.1. Занятия лекционного типа.

№ п/п	Тематика лекционных занятий / краткое содержание
1	Введение в DevOps. Рассматриваемые вопросы: - базовые принципы; - процессы и инструменты; - виртуализация и базовая автоматизация;

№ п/п	Тематика лекционных занятий / краткое содержание
	- системы контроля версий и DevOps.
2	Управление конфигурацией. Рассматриваемые вопросы: - введение в Ansible. - шаблоны, хендлеры, хранилища, роли; - безопасность.
3	Конфигурация инфраструктуры. Рассматриваемые вопросы: - инфраструктура как код; - введение в Terraform; - настройка окружения, бекенды.
4	Типовые окружения. Рассматриваемые вопросы: - локальное окружение; - тестовое окружение; - staging; - production.
5	Контейнеризация. Рассматриваемые вопросы: - введение в Docker; - установка и настройка Docker; - работа с образами и контейнерами.
6	Кластеризация. Рассматриваемые вопросы: - распределенные приложения; - кластеризация и Docker.
7	CI/CD. Рассматриваемые вопросы: - введение в Jenkins; - установка и настройка Jenkins; - построение процесса непрерывной интеграции; - сценарии работы с Jenkins; - инструменты непрерывного развертывания ПО.
8	GitLab. Рассматриваемые вопросы: - основы GitLab; - система контроля версий; - pipeline; - GitLab CI; - инструменты непрерывного развертывания ПО.
9	Мониторинг. Рассматриваемые вопросы: - введение в Prometheus и Grafana; - установка и настройка Prometheus и Grafana; - сценарии работы с Prometheus и Grafana.
10	Elastic Stack. Рассматриваемые вопросы: - введение в Elastic Stack. - мониторинг на базе Elastic Stack;

№ п/п	Тематика лекционных занятий / краткое содержание
	- логирование, агрегация, визуализация и анализ данных.

4.2. Занятия семинарского типа.

Практические занятия

№ п/п	Тематика практических занятий/краткое содержание
1	Управление конфигурацией. В результате выполнения практической работы студент получает навык конфигурации окружения с помощью Ansible.
2	Контейнеризация. В результате выполнения практической работы студент получает навык работы с Docker контейнеризацией в различных окружениях.
3	Кластеризация. В результате выполнения практической работы студент получает навык работы с Docker кластеризацией и при разработке распределенных приложений.
4	CI и Jenkins. В результате выполнения практической работы студент на практических кейсах осваивает принципы непрерывной интеграции на базе Jenkins.
5	CD и Jenkins. В результате выполнения практической работы студент на практических кейсах осваивает принципы непрерывного развертывания ПО на базе Jenkins и Docker контейнеров.
6	Мониторинг. В результате выполнения практической работы студент на практических кейсах осваивает принципы логирования на базе Prometheus и визуализации на базе Grafana.

4.3. Самостоятельная работа обучающихся.

№ п/п	Вид самостоятельной работы
1	Изучение рекомендованной литературы.
2	Подготовка к практическим работам.
3	Подготовка к промежуточной аттестации.
4	Подготовка к текущему контролю.

5. Перечень изданий, которые рекомендуется использовать при освоении дисциплины (модуля).

№ п/п	Библиографическое описание	Место доступа
1	Б. Ластер Jenkins 2. Приступаем к работе : руководство. Москва : ДМК Пресс, 2019. - 652 с. - ISBN 978-5-97060-711-4.	https://e.lanbook.com/book/131697
2	Э. Моуэт Использование Docker. Москва : ДМК	https://e.lanbook.com/book/93576

	Пресс, 2017. - 354 с. - ISBN 978-5-97060-426-7.	
3	Э. Х. Сейерс, А. Милл Docker на практике. Москва : ДМК Пресс, 2020. - 516 с. - ISBN 978-5-97060-772-5.	https://e.lanbook.com/book/131719
4	М. Ленц Python Непрерывная интеграция и доставка : руководство. Москва : ДМК Пресс, 2020. - 168 с. - ISBN 978-5-97060-797-8.	https://e.lanbook.com/book/179462
5	П. С. Кочер Микросервисы и контейнеры Docker : руководство. Москва : ДМК Пресс, 2019. - 240 с. - ISBN 978-5-97060-739-8.	https://e.lanbook.com/book/123710

6. Перечень современных профессиональных баз данных и информационных справочных систем, которые могут использоваться при освоении дисциплины (модуля).

Электронно-библиотечная система издательства «Лань» (http://e.lanbook.com/)

Электронно-библиотечная система ibooks.ru (http://ibooks.ru/)

Открытые лекции (https://sphere.mail.ru/materials/video/#16)

Учебные курсы Microsoft (https://www.microsoft.com/ru-ru/learning/training.aspx)

7. Перечень лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства, необходимого для освоения дисциплины (модуля).

Microsoft Office
.NET 6
Java 17
Jenkins
Docker
Prometheus
Grafana
Ubuntu Server 20.04
Virtual Box
Браузер с выходом в интернет
Microsoft Visual Studio CE
JetBrains IntelliJ IDEA Community Edition

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

Учебные аудитории для проведения учебных занятий, оснащенные

компьютерной техникой и наборами демонстрационного оборудования.

Для практических занятий – наличие персональных компьютеров вычислительного класса.

9. Форма промежуточной аттестации:

Экзамен в 8 семестре.

10. Оценочные материалы.

Оценочные материалы, применяемые при проведении промежуточной аттестации, разрабатываются в соответствии с локальным нормативным актом РУТ (МИИТ).

Авторы:

старший преподаватель кафедры
«Цифровые технологии управления
транспортными процессами» Е.А. Заманов

Согласовано:

Заведующий кафедрой ЦТУТП В.Е. Нутович

Председатель учебно-методической
комиссии Н.А.Клычева