

**МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ**  
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ**  
**УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**  
**«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА»**  
**(РУТ (МИИТ))**



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

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)**

**Автоматизация жизненного цикла ПО**

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: 01.09.2026

## 1. Общие сведения о дисциплине (модуле).

Дисциплина «Автоматизация жизненного цикла ПО» формирует у студентов системное понимание процессов поставки программного обеспечения и данных в условиях современной кросс-функциональной разработки. Курс выступает интегратором знаний, полученных при изучении операционных систем, инфраструктуры и мониторинга, переводя фокус на социотехническое взаимодействие ролей – разработчиков, тестировщиков, дата-сайентистов, ML-инженеров и специалистов по информационной безопасности. В рамках дисциплины рассматриваются передовые индустриальные практики – DevOps, MLOps, DataOps и DevSecOps. Студенты узнают, как проектировать контракты взаимодействия между командами, выстраивать автоматизированные конвейеры непрерывной интеграции и доставки, обеспечивать воспроизводимость экспериментов машинного обучения и встраивать проверки информационной безопасности непосредственно в процесс разработки. Особое внимание уделяется принципам платформенной инженерии и использованию импортозамещенного стека технологий. Дисциплина готовит востребованных специалистов, способных устранять барьеры между отделами и ускорять вывод продуктов на рынок без потери качества и безопасности.

Целью освоения дисциплины является формирование у обучающихся системных компетенций в области проектирования, автоматизации и обеспечения безопасности сквозных процессов жизненного цикла программного обеспечения, моделей машинного обучения и данных в условиях кросс-функционального взаимодействия команд и применения импортозамещенного технологического стека.

Для достижения поставленной цели в рамках дисциплины решается комплекс задач, направленных на формирование у обучающихся способности: анализировать потоки создания ценности и устранять барьеры между ролями разработки, тестирования, эксплуатации и информационной безопасности. Студенты учатся проектировать и настраивать автоматизированные конвейеры непрерывной интеграции и доставки с учетом требований воспроизводимости сред. Обучающиеся интегрируют практики статического и динамического анализа уязвимостей на ранних этапах разработки. Выпускники выстраивают процессы версионирования и контроля качества данных и моделей машинного обучения. Студенты применяют методологии ретроспективного анализа инцидентов для непрерывного улучшения отказоустойчивости автоматизированных систем.

## 2. Планируемые результаты обучения по дисциплине (модулю).

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

**ПК-4** - Способен автоматизировать процессы сборки, тестирования и развёртывания программных продуктов на протяжении их жизненного цикла;

**ПК-10** - Способен обеспечивать информационную безопасность программных продуктов на всех этапах их жизненного цикла.

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

### **Знать:**

- концепция жизненного цикла программного обеспечения и эволюция культурно-организационных подходов к разработке;

- архитектурные принципы, паттерны и этапы построения конвейеров непрерывной интеграции и непрерывной доставки;

- стратегии управления версиями исходного кода и модели кросс-функционального взаимодействия команд в едином репозитории;

- принципы контейнеризации приложений, изоляции зависимостей и стандартизации сред выполнения на всех этапах поставки;

- архитектура систем оркестрации контейнеров и концепция платформенных и бессерверных решений для автоматизированного развертывания;

- парадигма «Инфраструктура как код» и методологии декларативного, идиоматического управления конфигурациями вычислительных сред;

- пирамида автоматизированного тестирования программного обеспечения и методология интеграции контрольных точек качества в конвейер поставки;

- концепция комплексной наблюдаемости распределенных систем и архитектура сбора телеметрических данных;

- культурные и технологические основы DevSecOps, методология сдвига безопасности влево и принципы Security by Design;

- методы и инструменты автоматического анализа уязвимостей исходного кода, открытых зависимостей и образов контейнеров;

- принципы криптографического управления секретами, ротации доступов и безопасной инъекции конфиденциальных данных в среды выполнения;

- специфика жизненного цикла моделей машинного обучения и концептуальные отличия MLOps от классического DevOps;

- технологии версионирования наборов данных, артефактов моделей машинного обучения и трекинга вычислительных экспериментов;
- архитектура пайплайнов обработки данных и методы автоматизированного контроля качества данных в ETL/ELT-процессах;
- принципы контрактного взаимодействия распределенных компонентов и стандарты описания интерфейсов прикладного программирования;
- методология ретроспективного анализа инцидентов и проектирование превентивных мер повышения отказоустойчивости автоматизированных систем.

**Уметь:**

- уметь проектировать и настраивать конвейеры непрерывной интеграции и доставки при помощи систем автоматизации в условиях необходимости обеспечения бесшовной автоматизированной передачи артефактов между ролями разработки, тестирования и эксплуатации;
- уметь автоматизировать процессы сборки, тестирования и развертывания программных продуктов при помощи инструментов контейнеризации и описания инфраструктурных стеков в условиях обеспечения идемпотентности и полной воспроизводимости сред для всех участников жизненного цикла;
- уметь обеспечивать информационную безопасность программных продуктов на этапах разработки и поставки при помощи интеграции инструментов статического анализа кода и централизованного управления секретами в условиях реализации практик DevSecOps и принципа сдвига безопасности влево;
- уметь автоматизировать процессы управления жизненным циклом моделей машинного обучения и данных при помощи специализированных платформ версионирования и трекинга экспериментов в условиях жестких требований к воспроизводимости датасетов и мониторинга дрейфа моделей в продуктивной среде;
- уметь настраивать комплексную наблюдаемость за инфраструктурой, приложениями и бизнес-метриками при помощи стека сбора и визуализации метрик и логов в условиях необходимости своевременного выявления аномалий и автоматической маршрутизации алертов между ответственными ролями;
- уметь проектировать стратегии ветвления кода и контракты взаимодействия между кросс-функциональными командами при помощи систем контроля версий и спецификаций API в условиях устранения ручных рукопожатий и минимизации конфликтов при параллельной разработке;

- уметь разворачивать и конфигурировать вычислительную инфраструктуру как код при помощи декларативных инструментов автоматизации в условиях требований импортозамещения и использования отечественных операционных систем и облачных PaaS-решений;

- уметь проводить ретроспективный анализ инцидентов и предлагать превентивные меры при помощи методологии Blameless Post-Mortem в условиях необходимости непрерывного улучшения автоматизированных процессов и повышения общей отказоустойчивости системы.

### **Владеть:**

- навыки практической работы не формируются в рамках данной дисциплины согласно учебному плану в связи с отсутствием лабораторных работ и курсового проектирования.

### 3. Объем дисциплины (модуля).

#### 3.1. Общая трудоемкость дисциплины (модуля).

Общая трудоемкость дисциплины (модуля) составляет 4 з.е. (144 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	Семестр №7
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 80 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	<p>Эволюция подходов к жизненному циклу ПО и культура автоматизации</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- историческая ретроспектива моделей разработки от каскадных подходов до гибких методологий;</li> <li>- культурные и организационные барьеры при взаимодействии кросс-функциональных команд;</li> <li>- концепция DevOps и Platform Engineering как методология устранения разрозненности процессов.</li> </ul>
2	<p>Архитектура и паттерны конвейеров непрерывной интеграции и доставки</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- базовые принципы и этапы построения конвейеров непрерывной интеграции;</li> <li>- архитектурные паттерны реализации непрерывной доставки и автоматизированного развертывания;</li> <li>- стратегии управления артефактами сборки и механизмы триггеров релизов.</li> </ul>
3	<p>Стратегии ветвления кода и модели кросс-функциональной коллаборации</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- модели управления версиями исходного кода в распределенных системах разработки;</li> <li>- проектирование рабочих процессов для параллельной работы различных ролей в едином репозитории;</li> <li>- методологии автоматизации контроля качества слияний и проведения код-ревью.</li> </ul>
4	<p>Контейнеризация приложений как стандарт унификации сред</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- принципы изоляции процессов и стандартизации сред выполнения на всех этапах поставки;</li> <li>- архитектура контейнерных рантаймов и механизмы оптимизации зависимостей;</li> <li>- методологии создания многоуровневых образов для различных компонентов системы.</li> </ul>
5	<p>Платформенные и бессерверные решения для автоматизированного развертывания</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- концепции оркестрации контейнеров и управления распределенными системами;</li> <li>- архитектура платформенных сервисов и бессерверных вычислительных сред;</li> <li>- стратегии адаптации приложений для развертывания в облачных PaaS-решениях.</li> </ul>
6	<p>Инфраструктура как код и декларативное управление конфигурациями</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- парадигма описания вычислительных сред в виде версионизируемого кода;</li> <li>- принципы идемпотентности и воспроизводимости конфигурационных скриптов;</li> <li>- инструменты автоматизации подготовки и настройки серверной инфраструктуры.</li> </ul>
7	<p>Пирамида автоматизированного тестирования в конвейере поставки</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- пирамида тестирования и распределение контрольных точек качества по этапам сборки;</li> <li>- методологии интеграции юнит-тестов и статических анализаторов в этапы непрерывной интеграции;</li> <li>- стратегии автоматизации приемочных и интеграционных испытаний перед развертыванием.</li> </ul>
8	<p>Концепция комплексной наблюдаемости распределенных систем</p> <p>Рассматриваемые вопросы:</p>

№ п/п	Тематика лекционных занятий / краткое содержание
	<ul style="list-style-type: none"> <li>- концептуальные отличия наблюдаемости от классического мониторинга инфраструктуры;</li> <li>- архитектура сбора и корреляции метрик, логов и трейсов в единое телеметрическое окно;</li> <li>- инструменты визуализации данных для предоставления специфичных слоев истины различным ролям.</li> </ul>
9	<b>Культурные и технологические основы DevSecOps</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- методология сдвига безопасности влево на ранних этапах разработки;</li> <li>- принципы Security by Design при проектировании автоматизированных конвейеров;</li> <li>- интеграция процессов информационной безопасности в ежедневные практики команд без потери скорости.</li> </ul>
10	<b>Автоматизированный анализ уязвимостей и зависимостей</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- методы статического и динамического анализа безопасности исходного кода;</li> <li>- технологии сканирования открытых библиотек на наличие лицензионных и security-рисков;</li> <li>- автоматизация проверки образов контейнеров на наличие известных уязвимостей.</li> </ul>
11	<b>Криптографическое управление секретами и доступами</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- принципы централизованного хранения и ротации конфиденциальных данных;</li> <li>- механизмы безопасной инъекции секретов в среды выполнения и автоматизированные пайплайны;</li> <li>- политики разграничения доступов и аудита операций с чувствительной информацией.</li> </ul>
12	<b>Специфика жизненного цикла моделей машинного обучения</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- концептуальные отличия процессов разработки и эксплуатации моделей машинного обучения;</li> <li>- архитектурные паттерны построения MLOps-платформ и их интеграция с классическим DevOps;</li> <li>- проблемы воспроизводимости экспериментов и мониторинга дрейфа данных в продуктивной среде.</li> </ul>
13	<b>Версионирование данных и трекинг вычислительных экспериментов</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- технологии управления версиями наборов данных и артефактов обученных моделей;</li> <li>- методологии фиксации гиперпараметров и метрик качества вычислительных экспериментов;</li> <li>- инструменты каталогизации моделей для обеспечения их прозрачности и переиспользования.</li> </ul>
14	<b>Архитектура пайплайнов данных и методы контроля качества</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- принципы построения автоматизированных конвейеров извлечения, трансформации и загрузки данных;</li> <li>- методологии валидации и контроля качества данных на различных этапах обработки;</li> <li>- стратегии обеспечения согласованности и актуальности данных в аналитических системах.</li> </ul>
15	<b>Контрактное взаимодействие и стандарты описания интерфейсов</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- принципы проектирования интерфейсов прикладного программирования для распределенных систем;</li> <li>- стандарты описания жестких контрактов взаимодействия между микросервисами и компонентами;</li> <li>- методологии версионирования и документирования API для кросс-функциональных команд.</li> </ul>
16	<b>Ретроспективный анализ инцидентов и проектирование отказоустойчивости</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- методология проведения разбора инцидентов без поиска виновных;</li> <li>- проектирование превентивных мер и механизмов автоматического восстановления систем;</li> <li>- практики непрерывного улучшения процессов на основе извлеченных уроков.</li> </ul>

#### 4.2. Занятия семинарского типа.

## Практические занятия

№ п/п	Тематика практических занятий/краткое содержание
1	<p><b>Ролевой анализ и картирование потока создания ценности</b></p> <p>Студент анализирует текущий поток создания ценности в гипотетической компании и выявляет узкие места при передаче артефактов между ролями разработки, тестирования, машинного обучения и информационной безопасности. Обучающийся составляет карту потока создания ценности с использованием инструментов офисного пакета. На основе анализа студент проектирует целевую модель кросс-функционального взаимодействия и формализует требования к автоматизации ручных операций.</p>
2	<p><b>Проектирование API-контрактов между кросс-функциональными ролями</b></p> <p>Студент разрабатывает спецификации интерфейсов прикладного программирования для фиксации жестких обязательств между бэкенд-сервисами и ML-компонентами. Обучающийся описывает форматы обмена данными и версии контрактов в текстовом редакторе. Результатом работы становится документация, исключая хаос интеграции при параллельной разработке независимых модулей.</p>
3	<p><b>MLOps – Проектирование стратегии версионирования моделей и данных</b></p> <p>Студент проектирует архитектуру хранения датасетов и артефактов обученных моделей с применением специализированных инструментов. Обучающийся определяет правила именования версий и метаданных экспериментов для синхронизации работы дата-сайентистов и ML-инженеров. Итогом занятия становится регламент версионирования, интегрируемый в общую систему контроля версий проекта.</p>
4	<p><b>Интеграция MLOps практик в CI/CD пайплайн</b></p> <p>Студент модифицирует конфигурационный файл конвейера непрерывной интеграции для автоматизации процессов машинного обучения. Обучающийся настраивает триггеры, которые при обновлении кода модели инициируют проверку воспроизводимости эксперимента и сборку образа сервиса инференса. Студент проверяет корректность выполнения этапов конвейера в тестовой среде.</p>
5	<p><b>DataOps – Внедрение автоматизированного контроля качества данных</b></p> <p>Студент внедряет в этап извлечения и преобразования данных автоматизированные проверки их качества. Обучающийся настраивает правила валидации структурных и содержательных характеристик входящих потоков информации. Результатом работы становится конфигурация, блокирующая передачу некорректных данных в аналитические хранилища.</p>
6	<p><b>DevSecOps – Реализация принципа сдвига безопасности влево</b></p> <p>Студент интегрирует в конвейер поставки инструменты статического анализа кода и сканирования зависимостей. Обучающийся настраивает пороговые значения критичности уязвимостей, при превышении которых процесс слияния кода автоматически блокируется. Студент анализирует отчеты сканеров и предлагает способы устранения выявленных рисков.</p>
7	<p><b>DevSecOps – Управление секретами и сканирование контейнерных образов</b></p> <p>Студент настраивает интеграцию конвейера с централизованным хранилищем конфиденциальных данных для безопасной инъекции секретов. Обучающийся добавляет в процесс сборки этап автоматического сканирования контейнерных образов на наличие известных уязвимостей. Студент проверяет отсутствие хардкода паролей и токенов в конфигурационных файлах проекта.</p>
8	<p><b>MLOps – Проектирование мониторинга дрейфа данных и моделей</b></p> <p>Студент проектирует механизмы отслеживания статистических характеристик входных данных и предсказаний модели в продуктивной среде. Обучающийся настраивает сбор соответствующих метрик и определяет пороговые значения для срабатывания сигналов о деградации качества. Итогом занятия становится спецификация алертов для ML-инженера.</p>
9	<p><b>Автоматизация процессов непрерывного переобучения моделей</b></p> <p>Студент конфигурирует автоматический запуск пайплайна дообучения модели при обнаружении критического дрейфа данных. Обучающийся описывает процесс безопасного выката обновленной</p>

№ п/п	Тематика практических занятий/краткое содержание
	версии модели в продуктивную среду с минимальным влиянием на бизнес-процессы. Студент проверяет корректность работы триггеров и последовательность этапов обновления.
10	<b>Наблюдаемость как сервис для различных ролей</b> Студент проектирует маршрутизацию телеметрических данных и алертов в зависимости от зоны ответственности специалистов. Обучающийся описывает состав метрик и логов, необходимых бизнес-аналитикам, инженерам эксплуатации и специалистам по информационной безопасности. Результатом работы становится матрица доступа к дашбордам системы мониторинга.
11	<b>Проектирование внутренней платформы для разработчиков (IDP)</b> Студент разрабатывает концепцию портала самообслуживания для стандартизации шаблонов проектов и снижения когнитивной нагрузки на команды. Обучающийся описывает пользовательские сценарии инициализации новых сервисов с предустановленными практиками безопасности и наблюдаемости. Студент формирует требования к функционалу платформы для последующей реализации.
12	<b>GitOps как эволюция управления инфраструктурными конфигурациями</b> Студент проектирует переход к декларативному управлению состоянием инфраструктуры через репозиторий системы контроля версий. Обучающийся описывает процессы ревью инфраструктурных изменений и роли участников в утверждении конфигурационных файлов. Итогом занятия становится регламент работы с GitOps-репозиторием.
13	<b>Стратегии безопасного выката и отката изменений</b> Студент разрабатывает регламент и описывает автоматизацию механизмов канареечного развертывания или сине-зеленого деплоя для микросервисов. Обучающийся моделирует сценарии быстрого отката изменений при обнаружении критических ошибок в продуктивной среде. Студент оценивает влияние выбранных стратегий на доступность сервиса для конечных пользователей.
14	<b>Культура инцидент-менеджмента и ретроспективный анализ</b> Студент моделирует критический инцидент, связанный с утечкой данных или падением качества ML-модели. Обучающийся проводит разбор происшествия по методологии без поиска виновных и документирует коренные причины сбоя. Результатом работы становится отчет с предложениями по модификации платформы для предотвращения подобных ситуаций.
15	<b>Аудит и комплаенс автоматизированных процессов поставки</b> Студент проводит комплексную проверку построенного конвейера на соответствие принципам безопасности и воспроизводимости. Обучающийся составляет чек-лист для верификации прослеживаемости всех артефактов от исходного кода до продуктивного развертывания. Студент формирует итоговое заключение о готовности платформы к промышленной эксплуатации.
16	<b>Финальная защита платформы кросс-функционального взаимодействия</b> Студент систематизирует все разработанные артефакты в единое связанное портфолио работ. Обучающийся готовит итоговую презентацию и демонстрирует, как внедренные практики устранили барьеры между кросс-функциональными командами. Студент отвечает на вопросы по архитектурным решениям и защищенности спроектированной системы.

#### 4.3. Самостоятельная работа обучающихся.

№ п/п	Вид самостоятельной работы
1	Изучение рекомендованной литературы.
2	Подготовка к лабораторным 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	Баланов, А. Н. DevOps – интеграция и автоматизация : учебное пособие для вузов / А. Н. Баланов. — 3-е изд., стер. — Санкт-Петербург : Лань, 2026. — 240 с. — Текст : электронный	ЭБС Лань. — URL: <a href="https://e.lanbook.com/book/509963">https://e.lanbook.com/book/509963</a> (дата обращения: 09.06.2026).
2	Херинг, М. DevOps для современного предприятия : учебное пособие / М. Херинг ; перевод с английского М. А. Райтмана. — Москва : ДМК Пресс, 2020. — 232 с. — Текст : электронный	ЭБС Лань. — URL: <a href="https://e.lanbook.com/book/140580">https://e.lanbook.com/book/140580</a> (дата обращения: 09.06.2026)
3	Митрина, О. А. Технологии и инструментарий машинного обучения : учебное пособие / О. А. Митрина. — Текст : электронный	ЭБС Лань. — URL: <a href="https://e.lanbook.com/book/368633">https://e.lanbook.com/book/368633</a> (дата обращения: 09.06.2026)
4	Настройка и администрирование сервисного программного обеспечения : учебно-методическое пособие. — Текст : электронный	ЭБС Лань. — URL: <a href="https://e.lanbook.com/book/382442">https://e.lanbook.com/book/382442</a> (дата обращения: 09.06.2026)
5	Кульнев, А. В. Машинное обучение и анализ данных : учебное пособие / А. В. Кульнев. — Текст : электронный	ЭБС Лань. — URL: <a href="https://e.lanbook.com/book/123456">https://e.lanbook.com/book/123456</a> (дата обращения: 09.06.2026)

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

официальная документация платформ непрерывной интеграции и доставки;

официальная документация систем версионирования данных и трекинга ML-экспериментов;

официальная документация средств криптографического управления секретами;

официальная документация стеков наблюдаемости и телеметрии;

официальная документация инструментов декларативного описания инфраструктуры;

официальная документация платформ контейнеризации и изоляции сред.

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

операционные системы: отечественные дистрибутивы Linux, включенные в единый реестр ПО;

офисные пакеты: отечественные решения для подготовки текстовой и презентационной документации;

среды разработки: открытые редакторы кода и среды выполнения скриптов;

инструменты контейнеризации и IaC: платформы для изоляции сред и декларативного описания инфраструктуры;

платформы CI/CD: системы автоматизации конвейеров сборки, тестирования и поставки;

средства информационной безопасности: централизованные хранилища секретов и сканеры уязвимостей;

инструменты MLOps и DataOps: платформы версионирования датасетов и трекинга экспериментов;

системы наблюдаемости: стеки для сбора телеметрии, логирования и визуализации метрик;

системы управления базами данных: реляционные СУБД из реестра отечественного ПО.

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 7 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»

И.С. Разживайкин

Согласовано:

Заведующий кафедрой ЦТУТП

В.Е. Нутович

Председатель учебно-методической  
комиссии

Н.А. Андриянова