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 директором ФГАОУ ВО РУТ
(МИИТ) Париновым Д.В.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Операционные среды и корпоративная виртуализация

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 ФГАОУ ВО РУТ (МИИТ)
ID подписи: 937226
Подписал: директор центра Проневич Ольга Борисовна
Дата: 12.11.2026

1. Общие сведения о дисциплине (модуле).

Дисциплина формирует способность понимать и учитывать работу операционной среды, в которой исполняется прикладное программное обеспечение, и отвечать за её эксплуатационную устойчивость. Рынок испытывает устойчивый дефицит специалистов инфраструктурных ролей, а импортозамещение платформ виртуализации обеспечивает многолетний спрос на владение отечественным стеком; в транспортной отрасли расчётные сервисы работают в режиме непрерывной доступности, и цена отказа измеряется реальными потерями грузоотправителей. В ходе освоения студент выступает в роли младшего инженера исследователя на едином учебном стенде расчётного сервиса транспортной компании под управлением РЕД ОС: наблюдает за процессами, памятью, файловой системой и правами, собирает воспроизводимую контейнерную среду с сетями, постоянными данными и ресурсными ограничениями, строит контур наблюдаемости, воспроизводит управляемый отказ и доказательно восстанавливает сервис. Итогом становится курсовая работа – портфель из карты среды, декларативной контейнерной конфигурации, контура наблюдения, протокола отказа и эксплуатационной записки.

Целью освоения дисциплины является формирование способности понимать и учитывать работу процессов, памяти, файловых систем и устройств, создавать воспроизводимую изолированную среду выполнения приложения, настраивать наблюдение за её состоянием и доказательно диагностировать ресурсные и конфигурационные сбои.

Для достижения поставленной цели в рамках дисциплины решается комплекс задач, направленных на формирование у обучающихся способности: объяснять поведение приложения через процессы, потоки, виртуальную память, файловую систему, ввод и вывод и системные вызовы. Создавать воспроизводимую контейнерную среду выполнения приложения с внешней конфигурацией, сетями, постоянными данными и ограничением ресурсов. Настраивать проверки готовности и работоспособности, сбор системных показателей и технологических журналов. Воспроизводить управляемый отказ, локализовать его область и проверять сохранность данных и полезной функции. Анализировать нагрузку на процессор, память, дисковую подсистему и очередь выполнения, различать конкурирующие причины деградации. Безопасно восстанавливать приложение и документировать последовательность диагностики, исправление и возможность отката.

2. Планируемые результаты обучения по дисциплине (модулю).

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

ОПК-3 - Способен использовать современные информационные технологии и программно-аппаратные средства, в том числе отечественного производства, при решении задач профессиональной деятельности;

ППК-Р6 - Способен участвовать в промышленной разработке программного обеспечения.

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

Знать:

- классы современных информационных технологий и программно-аппаратных средств;
- отечественные операционные системы, СУБД и облачные платформы (Единый реестр российского ПО);
- критерии выбора программно-аппаратных средств под профессиональную задачу;
- принципы Continuous Integration and Continuous Delivery (CI/CD);
- системы мониторинга и логирования в продуктивной среде;
- стратегии развертывания промышленного программного обеспечения;
- основы работы с облачными платформами.

Уметь:

- обоснованно выбирать информационные технологии и программно-аппаратные средства;
- применять отечественные программно-аппаратные средства при решении задач;
- оценивать ограничения, совместимость и стоимость владения выбранных средств;
- настраивать потоки работ CI/CD;
- работать с контейнеризацией и оркестрацией;
- настраивать мониторинг в продуктивной среде;
- развертывать приложения в облаке;
- обнаруживать и устранять инциденты с работой продуктивного программного обеспечения.

Владеть:

- навыками работы в отечественной ИТ-инфраструктуре (включая разработку под ОС «Аврора»).

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	Семестр №3
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	Операционная система как операционная среда приложения Рассматриваемые вопросы: - место ОС между аппаратурой и приложением, ядро и системные вызовы; - режимы ядра и пользователя, границы ответственности ОС и прикладного ПО; - классы современных операционных сред – серверные, облачные, встроенные; - РЕД ОС как целевая операционная среда курса: редакции «Сервер» и «Рабочая станция»,

№ п/п	Тематика лекционных занятий / краткое содержание
	сертифицированная редакция; - обзор отечественных операционных систем и Единый реестр российского ПО.
2	Процессы и потоки: жизненный цикл и планирование Рассматриваемые вопросы: - процесс и поток как единицы исполнения, их создание и завершение; - состояния процесса и очередь выполнения, планировщик и приоритеты; - межпроцессное взаимодействие и его роль в многокомпонентных сервисах; - наблюдение за процессами штатными средствами Linux (ps, top, /proc).
3	Управление памятью в операционной среде Рассматриваемые вопросы: - виртуальная память и страничная организация; - подкачка, кэширование и реальное потребление памяти приложением; - исчерпание памяти и механизм OOM, типовые сценарии отказа по памяти; - интерпретация показателей памяти (free, vmstat).
4	Файловые системы, права и ввод-вывод Рассматриваемые вопросы: - иерархия файловой системы Linux, типы файловых систем; - файловые дескрипторы, права доступа и списки контроля доступа; - дисковая подсистема и диагностика её насыщения (iostat, lsof); - принцип минимальных привилегий для сервисных учётных записей.
5	Изоляция файлового и системного окружения Рассматриваемые вопросы: - изоляция как способ ограничения области отказа и ущерба; - chroot, пространства имён и cgroups как фундамент контейнеризации; - соотнесение механизмов изоляции с требованиями безопасности и эксплуатации; - инструментальная поддержка изоляции в РЕД ОС.
6	Виртуализация аппаратных ресурсов и корпоративные платформы Рассматриваемые вопросы: - концепция виртуализации, типы гипервизоров и аппаратная поддержка; - полная виртуализация и паравиртуализация, их стоимость и ограничения; - корпоративные платформы виртуализации – VMware как исторический эталон; - платформа РЕД Виртуализация на базе KVM: архитектура, управление кластером, программно-определяемые сети, отказоустойчивость и модуль аварийного восстановления; - критерии выбора и стоимость владения виртуализационной инфраструктурой.
7	Контейнерная виртуализация Рассматриваемые вопросы: - контейнер как процесс в изолированном окружении, отличие от виртуальной машины; - образы, слои и реестры образов, жизненный цикл контейнера; - штатные средства контейнеризации РЕД ОС – Docker, containerd, Podman, и их эволюция; - реестр образов registry.red-soft.ru и работа в изолированном контуре; - границы применимости контейнеров в корпоративной среде.
8	Декларативное описание воспроизводимых сред Рассматриваемые вопросы: - файл образа (Dockerfile/Containerfile) как декларативная спецификация среды; - композиция многокомпонентных сервисов (Compose, Quadlet); - принципы воспроизводимости и иммутабельности инфраструктуры; - воспроизводимый подъём среды одной командой как инженерный стандарт.
9	Сети и постоянные данные в контейнерных средах Рассматриваемые вопросы: - типы сетей контейнеров и изоляция сетевого взаимодействия;

№ п/п	Тематика лекционных занятий / краткое содержание
	- постоянные тома и жизненный цикл данных вне контейнера; - внешняя конфигурация приложения и двенадцать факторов как ориентир; - сохранность данных при пересоздании и обновлении компонентов.
10	Управление ресурсами и качество обслуживания Рассматриваемые вопросы: - классы критичности сервисов и требования к доступности; - ресурсные ограничения процессора, памяти и ввода-вывода через cgroups; - конкуренция за ресурсы и деградация полезной функции; - политики поведения при перегрузке и приоритизация критичных компонентов.
11	Наблюдаемость программных систем Рассматриваемые вопросы: - наблюдаемость как свойство среды – метрики, журналы, проверки состояния; - проверки готовности и работоспособности, их роль в автоматике; - мониторинг «чёрного» и «белого» ящика; - принцип локализации отказа без доступа внутрь компонента.
12	Системы мониторинга и журналирования продуктивной среды Рассматриваемые вопросы: - архитектура сбора метрик – Prometheus, Grafana, Zabbix; - технологические журналы, централизация логов и стек ELK; - оповещение и пороги, борьба с ложными срабатываниями; - панель наблюдения как инструмент эксплуатации.
13	Развёртывание и жизненный цикл ПО в корпоративной среде Рассматриваемые вопросы: - стратегии развертывания промышленного ПО и требования к обратимости; - принципы непрерывной интеграции и поставки (CI/CD); - облачные платформы и их место в корпоративной инфраструктуре; - отечественные облачные решения и импортозамещение инфраструктуры.
14	Оркестрация контейнеров: эволюция и границы применимости Рассматриваемые вопросы: - задачи планирования, самовосстановления и масштабирования контейнеров; - эволюция «виртуальная машина – контейнер – оркестратор»; - базовые концепции Kubernetes и его место в корпоративном стеке; - критерии перехода к оркестрации и цена усложнения.
15	Надёжность, отказы и восстановление Рассматриваемые вопросы: - классы отказов – ресурсные, процессные, конфигурационные, инфраструктурные; - изоляция области отказа, сохранность данных и полезной функции; - процедуры восстановления и отката, минимально инвазивные действия; - трудовые функции профстандарта «Системный администратор ИКС» – локализация отказов, восстановление, протоколирование.
16	Диагностика производительности и анализ деградации операционной среды Рассматриваемые вопросы: - методология анализа производительности – утилизация, насыщение, ошибки (подход USE); - таблица конкурирующих гипотез и управляемый эксперимент; - различение симптома, конкурирующей причины и первопричины деградации; - сравнение состояния системы «до – после» исправления как доказательство эффективности решения.

4.2. Занятия семинарского типа.

Лабораторные работы

№ п/п	Наименование лабораторных работ / краткое содержание
1	Развёртывание учебного стенда и первичное наблюдение за сервисом. Студент подключается к стенду под управлением РЕД ОС 8, запускает учебный расчётный сервис и фиксирует его состав – интерфейс на Java, расчётный и служебный обработчики, база данных PostgreSQL. Он находит процессы сервиса, определяет их иерархию, идентификаторы и пользователей, от имени которых они исполняются. Результатом является черновая схема процессов и протокол первичных наблюдений.
2	Исследование жизненного цикла и планирования процессов. Студент наблюдает состояния процессов сервиса под нагрузкой и без неё, изучает очередь выполнения и приоритеты штатными средствами Linux. Он сопоставляет потоки Java приложения с потоками операционной системы и объясняет наблюдаемое распределение процессорного времени. Итогом становится дополненная схема процессов и выводы о поведении планировщика.
3	Анализ потребления памяти и поведения при её исчерпании. Студент измеряет реальное потребление памяти компонентами сервиса, различая резидентную, виртуальную память и кэш. Он моделирует рост потребления памяти расчётным обработчиком и наблюдает реакцию системы вплоть до срабатывания механизма OOM. Результатом является таблица измерений и объяснение поведения сервиса при дефиците памяти.
4	Диагностика файловой системы, прав и дисковой нагрузки. Студент определяет, какие файлы и дескрипторы использует сервис, проверяет права доступа к рабочим каталогам и журналам. Он измеряет дисковую нагрузку при расчётных и служебных операциях и выявляет компонент, создающий наибольшее давление на подсистему ввода вывода. Итогом является раздел карты среды с перечнем файловых ресурсов и прав.
5	Укрепление окружения: пользователи, права и изоляция файлового окружения. Студент создаёт выделенные сервисные учётные записи, назначает минимально необходимые права на каталоги и файлы сервиса и проверяет отказ при попытке доступа за пределы полномочий. Он применяет изоляцию файлового окружения для служебного обработчика и подтверждает ограничение его видимости системы. Результатом является завершённая карта среды и обоснование модели разграничения доступа.
6	Сравнение уровней виртуализации и выбор целевой платформы. Студент сравнивает развёртывание сервиса на виртуальной машине в среде РЕД Виртуализация и в контейнере под РЕД ОС по времени подготовки, накладным расходам и изоляции. Он оценивает платформу РЕД Виртуализация и штатные средства контейнеризации РЕД ОС по критериям применимости и стоимости владения и обосновывает выбор контейнерной схемы для учебного стенда. Итогом становится таблица сравнения и решение о целевой архитектуре среды.
7	Сборка воспроизводимого образа расчётного сервиса. Студент пишет декларативный файл образа для компонентов сервиса на базе образов реестра registry.red-soft.ru, собирает образ штатными средствами контейнеризации РЕД ОС и проверяет его послойную структуру. Он устраняет ручные шаги настройки, добиваясь того, чтобы образ запускался на чистом окружении без доработок. Результатом является версионируемый образ и инструкция его воспроизводимой сборки.
8	Композиция многокомпонентной среды. Студент описывает декларативную композицию из интерфейса, расчётного и служебного обработчиков и базы данных PostgreSQL, задаёт порядок запуска и зависимости компонентов. Он проверяет подъём всей среды одной командой и корректное завершение работы. Итогом является файл композиции, принимаемый как основа среды курсовой работы.
9	Сети контейнеров и постоянные данные. Студент выделяет изолированные сети для внешнего интерфейса и внутреннего взаимодействия с базой данных, проверяя недоступность базы извне. Он подключает постоянные тома, пересоздаёт контейнер базы данных и доказывает сохранность данных и расчётных результатов. Результатом является доработанная композиция с сетевой схемой и томами.

№ п/п	Наименование лабораторных работ / краткое содержание
10	Ресурсные ограничения и защита критичной функции. Студент задаёт лимиты процессора и памяти для каждого компонента в соответствии с классами критичности, отдавая приоритет интерфейсу и базе данных перед служебными работами. Он запускает регламентную фоновую обработку и проверяет, что прикладная функция остаётся доступной. Итогом становится финальная конфигурация среды с обоснованными ограничениями.
11	Проверки готовности и работоспособности компонентов. Студент проектирует и настраивает проверки готовности и работоспособности для каждого компонента композиции, отличая состояния «запущен», «готов принимать работу» и «деградировал». Он проверяет реакцию среды на остановку отдельного компонента. Результатом является набор проверок состояния, встроенный в конфигурацию среды.
12	Сбор метрик и панель наблюдения. Студент развёртывает сбор метрик компонентов и хоста при помощи Prometheus и строит панель наблюдения в Grafana с показателями процессора, памяти, диска и состояния сервиса. Он подбирает пороги оповещений и проверяет их срабатывание на контролируемой нагрузке. Итогом является действующий контур наблюдения за средой.
13	Централизация технологических журналов. Студент настраивает сбор журналов всех контейнеров в единую точку, обеспечивая поиск событий по компоненту и времени. Он проверяет, что по метрикам и журналам можно локализовать отказ компонента без входа внутрь контейнера. Результатом является завершённый контур наблюдаемости.
14	Управляемый отказ: исчерпание памяти компонента. Студент по заданному сценарию воспроизводит отказ памяти в служебном обработчике и наблюдает реакцию среды через построенный контур наблюдения. Он доказывает, что интерфейс и база данных сохранили полезную функцию, а данные не утрачены. Итогом является протокол отказа с измерениями и границей области отказа.
15	Диагностика перегрузки процессора по кейсу индустриального партнёра. Студент воспроизводит ситуацию, в которой регламентная работа создаёт процессорную конкуренцию и недоступность прикладной функции. Он строит временную шкалу и таблицу гипотез, интерпретирует загрузку процессора и очередь выполнения, выбирает минимально инвазивное действие и исправляет неверную конфигурацию запуска. Результатом является исправленная конфигурация и журнал команд диагностики.
16	Восстановление, сравнение «до – после» и эксплуатационная записка. Студент повторяет единый нагрузочный сценарий для исходной и исправленной конфигураций, получает сопоставимые измерения и формулирует вывод о первопричине инцидента. Он оформляет эксплуатационную записку с процедурой восстановления и планом отката по ГОСТ в Р7-Офис. Итогом цепочки является полный портфель курсовой работы, готовый к защите.

4.3. Самостоятельная работа обучающихся.

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

4.4. Примерный перечень тем курсовых работ

Проектирование, развёртывание и доказательная эксплуатация изолированной воспроизводимой среды выполнения сервиса тарификации перевозок

Проектирование, развёртывание и доказательная эксплуатация изолированной воспроизводимой среды выполнения сервиса планирования маршрутов транспортных заданий

Проектирование, развёртывание и доказательная эксплуатация изолированной воспроизводимой среды выполнения сервиса диспетчеризации и распределения заданий между исполнителями

Проектирование, развёртывание и доказательная эксплуатация изолированной воспроизводимой среды выполнения сервиса приёма и обработки телематических событий

Проектирование, развёртывание и доказательная эксплуатация изолированной воспроизводимой среды выполнения сервиса формирования электронных перевозочных документов

Проектирование, развёртывание и доказательная эксплуатация изолированной воспроизводимой среды выполнения сервиса складского обеспечения рейсов

Проектирование, развёртывание и доказательная эксплуатация изолированной воспроизводимой среды выполнения сервиса начислений и взаиморасчётов с перевозчиками

Проектирование, развёртывание и доказательная эксплуатация изолированной воспроизводимой среды выполнения сервиса нормативно-справочной информации транспортных расчётов

Проектирование, развёртывание и доказательная эксплуатация изолированной воспроизводимой среды выполнения сервиса пакетной загрузки внешних реестров перевозок

Проектирование, развёртывание и доказательная эксплуатация изолированной воспроизводимой среды выполнения сервиса событийного обмена результатами расчётов

Проектирование, развёртывание и доказательная эксплуатация изолированной воспроизводимой среды выполнения сервиса контроля SLA выполнения транспортных заданий

Проектирование, развёртывание и доказательная эксплуатация изолированной воспроизводимой среды выполнения мультитенантного сервиса обработки заданий нескольких перевозчиков

Проектирование, развёртывание и доказательная эксплуатация изолированной воспроизводимой среды выполнения сервиса расчёта сроков доставки и котировок

Проектирование, развёртывание и доказательная эксплуатация изолированной воспроизводимой среды выполнения сервиса претензионной работы и перерасчётов по отклонениям

Проектирование, развёртывание и доказательная эксплуатация изолированной воспроизводимой среды выполнения сервиса отчётности по выполненным транспортным заданиям

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

№ п/п	Библиографическое описание	Место доступа
1	Гостев, И. М. Операционные системы : учебник и практикум для среднего профессионального образования / И. М. Гостев. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2024. — 164 с. — (Профессиональное образование). — ISBN 978-5-534-04951-0. — Текст : электронный	Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/539078 (дата обращения: 11.11.2026).
2	Староверова, Н. А. Операционные системы : учебник / Н. А. Староверова. — Санкт-Петербург : Лань, 2022. — 308 с. — ISBN 978-5-8114-4000-9. — Текст : электронный	Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/207089 (дата обращения: 11.11.2026).
3	Бобровский, В. И. Расширенное администрирование сетевой операционной системы GNU/Linux. Локальное системное администрирование : учебное пособие / В. И. Бобровский, А. В. Дагаев, Е. П. Журавель. — Санкт-Петербург : СПбГУТ им. М.А. Бонч-Бруевича, 2022. — 138 с. — ISBN 978-5-89160-252-6. — Текст : электронный	Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/279176 (дата обращения: 11.11.2026).
4	Романов, С. Л. Работа в операционной среде Linux: практикум для вузов : учебное пособие / С. Л. Романов. — Санкт-Петербург : БГТУ "Военмех" им. Д.Ф. Устинова, 2017. — 74 с. — Текст : электронный	Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/121866 (дата обращения: 11.11.2026).

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

ЭБС «Лань» – https://e.lanbook.com/ (учебники Староверовой Н. А., Кобылянского В. Г., Журавлева А. Е. и др., практикум по работе в операционной среде Linux).

Образовательная платформа «Юрайт» – https://urait.ru/ (учебник и практикум Гостева И. М.).

Официальная документация РЕД ОС – https://redos.red-soft.ru/support/docs/; База знаний РЕД ОС – https://redos.red-soft.ru/base/; реестр контейнерных образов – https://registry.red-soft.ru.

Официальный сайт и документация платформы РЕД Виртуализация – https://redvirt.red-soft.ru.

Официальная документация PostgreSQL / Postgres Pro – https://postgrespro.ru/docs.

Официальная документация Docker – https://docs.docker.com; Podman – https://docs.podman.io.

Официальная документация Prometheus – https://prometheus.io/docs; Grafana – https://grafana.com/docs; Zabbix – https://www.zabbix.com/documentation.

Официальная документация Kubernetes – https://kubernetes.io/ru/docs.

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

Операционная система: РЕД ОС 8 в редакциях «Сервер» и «Рабочая станция», сертифицированная редакция (ФСТЭК России).

Платформа корпоративной виртуализации: РЕД Виртуализация 8.0 на базе KVM – управление кластером и сетями (SDN), отказоустойчивость, модуль аварийного восстановления (DRM).

Контейнеризация и оркестрация: штатные сертифицированные средства РЕД ОС – Docker 28.1, containerd 1.7, Kubernetes 1.32; Podman + Buildah; реестр образов registry.red-soft.ru и локальный registry-зеркало (pull-through cache) для автономной работы класса.

СУБД и среда исполнения: PostgreSQL / Postgres Pro (в реестре ПО РФ), OpenJDK / Axiom JDK.

Наблюдаемость: Prometheus, Grafana, Zabbix, Loki / стек ELK.

Диагностика и нагрузочные эксперименты: штатные средства Linux (ps, top/htop, vmstat, iostat, lsof, strace, journalctl), stress-ng, cgroups v2 / systemd.

Управление кодом и CI/CD: Git, GitFlic (реестр ПО РФ) или GitVerse.

Офисные пакеты: Р7-Офис / МойОфис (для подготовки протоколов, эксплуатационной записки и курсовой работы по ГОСТ).

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 лабораторных занятий – наличие персональных компьютеров вычислительного класса.

9. Форма промежуточной аттестации:

Зачет в 3 семестре.

Курсовая работа в 3 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»
Института железнодорожного
транспорта И.С. Разживайкин

Согласовано:

Заведующий кафедрой ЦТУТП В.Е. Нутович

Директор ЦС IT - программ ВИШ О.Б. Проневич

Председатель учебно-методической
комиссии Н.А. Андриянова