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)

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

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):

ДПК-БП-1 - Способен обеспечивать безопасность бизнес-приложения на протяжении жизненного цикла;

ППК-Р3 - Способен выполнять интеграцию программных модулей и компонентов и проверять работоспособность выпусков программного продукта;

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

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, идемпотентного 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 №5
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	Эволюция подходов к жизненному циклу ПО и культура автоматизации Рассматриваемые вопросы: - историческая ретроспектива моделей разработки от каскадных подходов до гибких методологий; - культурные и организационные барьеры при взаимодействии кросс-функциональных команд; - концепция DevOps и Platform Engineering как методология устранения разрозненности процессов.
2	Архитектура и паттерны конвейеров непрерывной интеграции и доставки Рассматриваемые вопросы: - базовые принципы и этапы построения конвейеров непрерывной интеграции; - архитектурные паттерны реализации непрерывной доставки и автоматизированного развертывания; - стратегии управления артефактами сборки и механизмы триггеров релизов.
3	Стратегии ветвления кода и модели кросс-функциональной коллаборации Рассматриваемые вопросы: - модели управления версиями исходного кода в распределенных системах разработки; - проектирование рабочих процессов для параллельной работы различных ролей в едином репозитории; - методологии автоматизации контроля качества слияний и проведения код-ревью.
4	Контейнеризация приложений как стандарт унификации сред Рассматриваемые вопросы: - принципы изоляции процессов и стандартизации сред выполнения на всех этапах поставки; - архитектура контейнерных рантаймов и механизмы оптимизации зависимостей; - методологии создания многоуровневых образов для различных компонентов системы.
5	Платформенные и бессерверные решения для автоматизированного развертывания Рассматриваемые вопросы: - концепции оркестрации контейнеров и управления распределенными системами; - архитектура платформенных сервисов и бессерверных вычислительных сред; - стратегии адаптации приложений для развертывания в облачных PaaS-решениях.
6	Инфраструктура как код и декларативное управление конфигурациями Рассматриваемые вопросы: - парадигма описания вычислительных сред в виде версионируемого кода;

№ п/п	Тематика лекционных занятий / краткое содержание
	- принципы идемпотентности и воспроизводимости конфигурационных скриптов; - инструменты автоматизации подготовки и настройки серверной инфраструктуры.
7	Пирамида автоматизированного тестирования в конвейере поставки Рассматриваемые вопросы: - пирамида тестирования и распределение контрольных точек качества по этапам сборки; - методологии интеграции юнит-тестов и статических анализаторов в этапы непрерывной интеграции; - стратегии автоматизации приемочных и интеграционных испытаний перед развертыванием.
8	Концепция комплексной наблюдаемости распределенных систем Рассматриваемые вопросы: - концептуальные отличия наблюдаемости от классического мониторинга инфраструктуры; - архитектура сбора и корреляции метрик, логов и трейсов в единое телеметрическое окно; - инструменты визуализации данных для предоставления специфичных слоев истины различным ролям.
9	Культурные и технологические основы DevSecOps Рассматриваемые вопросы: - методология сдвига безопасности влево на ранних этапах разработки; - принципы Security by Design при проектировании автоматизированных конвейеров; - интеграция процессов информационной безопасности в ежедневные практики команд без потери скорости.
10	Автоматизированный анализ уязвимостей и зависимостей Рассматриваемые вопросы: - методы статического и динамического анализа безопасности исходного кода; - технологии сканирования открытых библиотек на наличие лицензионных и security-рисков; - автоматизация проверки образов контейнеров на наличие известных уязвимостей.
11	Криптографическое управление секретами и доступами Рассматриваемые вопросы: - принципы централизованного хранения и ротации конфиденциальных данных; - механизмы безопасной инъекции секретов в среды выполнения и автоматизированные пайплайны; - политики разграничения доступов и аудита операций с чувствительной информацией.
12	Специфика жизненного цикла моделей машинного обучения Рассматриваемые вопросы: - концептуальные отличия процессов разработки и эксплуатации моделей машинного обучения; - архитектурные паттерны построения MLOps-платформ и их интеграция с классическим DevOps; - проблемы воспроизводимости экспериментов и мониторинга дрейфа данных в продуктивной среде.
13	Версионирование данных и трекинг вычислительных экспериментов Рассматриваемые вопросы: - технологии управления версиями наборов данных и артефактов обученных моделей; - методологии фиксации гиперпараметров и метрик качества вычислительных экспериментов; - инструменты каталогизации моделей для обеспечения их прозрачности и переиспользования.
14	Архитектура пайплайнов данных и методы контроля качества Рассматриваемые вопросы: - принципы построения автоматизированных конвейеров извлечения, трансформации и загрузки данных; - методологии валидации и контроля качества данных на различных этапах обработки; - стратегии обеспечения согласованности и актуальности данных в аналитических системах.
15	Контрактное взаимодействие и стандарты описания интерфейсов Рассматриваемые вопросы: - принципы проектирования интерфейсов прикладного программирования для распределенных

№ п/п	Тематика лекционных занятий / краткое содержание
	систем; - стандарты описания жестких контрактов взаимодействия между микросервисами и компонентами; - методологии версионирования и документирования API для кросс-функциональных команд.
16	Ретроспективный анализ инцидентов и проектирование отказоустойчивости Рассматриваемые вопросы: - методология проведения разбора инцидентов без поиска виновных; - проектирование превентивных мер и механизмов автоматического восстановления систем; - практики непрерывного улучшения процессов на основе извлеченных уроков.

4.2. Занятия семинарского типа.

Лабораторные работы

№ п/п	Наименование лабораторных работ / краткое содержание
1	Ролевой анализ и картирование потока создания ценности 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.
2	Проектирование API-контрактов между кросс-функциональными ролями 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, исключающая хаос интеграции при параллельной разработке независимых модулей.
3	MLOps – Проектирование стратегии версионирования моделей и данных 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.
4	Интеграция MLOps практик в CI/CD пайплайн 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.
5	DataOps – Внедрение автоматизированного контроля качества данных 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.
6	DevSecOps – Реализация принципа сдвига безопасности влево 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.

№ п/п	Наименование лабораторных работ / краткое содержание
7	DevSecOps – Управление секретами и сканирование контейнерных образов 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.
8	MLOps – Проектирование мониторинга дрейфа данных и моделей 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.
9	Автоматизация процессов непрерывного переобучения моделей Студент конфигурирует автоматический запуск пайплайна дообучения модели при обнаружении критического дрейфа данных. Обучающийся описывает процесс безопасного выката обновленной версии модели в продуктивную среду с минимальным влиянием на бизнес-процессы. Студент проверяет корректность работы триггеров и последовательность этапов обновления.
10	Наблюдаемость как сервис для различных ролей 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	Проектирование внутренней платформы для разработчиков (IDP) 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	GitOps как эволюция управления инфраструктурными конфигурациями 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	Стратегии безопасного выката и отката изменений 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	Культура инцидент-менеджмента и ретроспективный анализ 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	Аудит и комплаенс автоматизированных процессов поставки 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	Финальная защита платформы кросс-функционального взаимодействия Студент систематизирует все разработанные артефакты в единое связное портфолио работ. Обучающийся готовит итоговую презентацию и демонстрирует, как внедренные практики

№ п/п	Наименование лабораторных работ / краткое содержание
	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: https://e.lanbook.com/book/509963 (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: https://e.lanbook.com/book/140580 (дата обращения: 09.06.2026)
3	Митрина, О. А. Технологии и инструментарий машинного обучения : учебное пособие / О. А. Митрина. — Текст : электронный	ЭБС Лань. — URL: https://e.lanbook.com/book/368633 (дата обращения: 09.06.2026)
4	Настройка и администрирование сервисного программного обеспечения : учебно-методическое пособие. — Текст : электронный	ЭБС Лань. — URL: https://e.lanbook.com/book/382442 (дата обращения: 09.06.2026)
5	Кульнев, А. В. Машинное обучение и анализ данных : учебное пособие / А. В. Кульнев. — Текст : электронный	ЭБС Лань. — URL: https://e.lanbook.com/book/123456 (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 лабораторных занятий – наличие персональных компьютеров вычислительного класса.

9. Форма промежуточной аттестации:

Зачет в 5 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