

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



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

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Бессерверные архитектуры и нативная облачная разработка

Направление подготовки: 09.04.01 Информатика и вычислительная техника

Направленность (профиль): Технологии проектирования 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: 5665

Подписал: заведующий кафедрой Нутович Вероника
Евгеньевна

Дата: 01.09.2026

1. Общие сведения о дисциплине (модуле).

Дисциплина посвящена проектированию программных продуктов для облачных платформ с использованием бессерверных вычислений, управляемых сервисов и средств управления контейнерами. В ходе изучения рассматриваются модели облачных услуг, облачные функции, прикладные шлюзы, очереди сообщений, управляемые базы данных, объектные хранилища, реестры контейнеров, Kubernetes, безопасность, наблюдаемость, стоимость владения, отказоустойчивость и правовые условия применения облачных платформ в Российской Федерации. На практических занятиях обучающиеся последовательно проектируют архитектуру облачного решения для транспортно-логистической задачи и готовят техническую документацию для его реализации.

Целью освоения дисциплины является формирование способности анализировать, проектировать и документировать облачные программные продукты на основе бессерверных вычислений, управляемых сервисов, контейнерной инфраструктуры и требований к безопасности, масштабируемости, стоимости и сопровождаемости в условиях российских облачных платформ.

Для достижения поставленной цели в рамках дисциплины решается комплекс задач, направленных на формирование у обучающихся способности – анализировать требования к облачному продукту, выбирать модель размещения и состав управляемых сервисов, проектировать бессерверные функции и событийное взаимодействие, описывать контейнерную архитектуру, задавать правила доступа и сетевой изоляции, оценивать стоимость и отказоустойчивость, проектировать наблюдаемость, учитывать требования информационной безопасности и готовить техническую документацию.

2. Планируемые результаты обучения по дисциплине (модулю).

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

ПК-1 - Способен проектировать и разрабатывать программные продукты с использованием облачных платформ и технологий виртуализации для корпоративного рынка;

ПК-5 - Способен проектировать и разрабатывать программные продукты с применением перспективных методов исследования на основе мировых тенденций развития вычислительной техники и информационных технологий.

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

Знать:

- роль облачных платформ в цифровизации транспортно-логистической отрасли и разработке масштабируемых программных продуктов
- модели облачных услуг, включая инфраструктуру как услугу, платформу как услугу, программный сервис и бессерверные вычисления
- принципы нативной облачной разработки, включая декомпозицию сервиса, внешнюю настройку, автоматическое масштабирование, устойчивость к отказам и наблюдаемость
- устройство российских облачных платформ и состав типовых сервисов, включая вычисления, сеть, хранилища, базы данных, очереди сообщений, функции и управление контейнерами
- назначение облачной функции, включая событие запуска, среду выполнения, ограничение времени, холодный запуск, параметры окружения и права доступа
- принципы событийной архитектуры, включая событие, очередь, тему, обработчик, повторную доставку, идемпотентность и очередь недоставленных сообщений
- назначение прикладного шлюза и маршрутизации запросов при публикации программных интерфейсов в облачной среде
- модели хранения данных в облаке, включая управляемые реляционные базы, ключ-значение, объектное хранилище и временное хранилище сообщений
- принципы контейнеризации приложения и использования реестра образов при подготовке сервиса к облачному размещению
- назначение Kubernetes как средства управления контейнерными приложениями, включая узел, модуль, службу, развертывание, секрет и настройку
- способы сочетания бессерверных функций, контейнерных сервисов и управляемых ресурсов в единой архитектуре программного продукта
- принципы сетевой архитектуры облачного решения, включая виртуальную сеть, подсеть, публичный доступ, частный доступ, балансировку нагрузки и защищенное соединение
- модели управления доступом, включая учетную запись, роль, политику, ключ доступа, секрет и принцип минимально необходимых прав

- требования к защите данных в облачной среде, включая шифрование, хранение секретов, резервное копирование, журналирование действий и соответствие российским ограничениям

- подходы к наблюдаемости облачных приложений, включая метрики, журналы событий, трассировки, проверки доступности и правила оповещения

- методы оценки стоимости облачного решения, включая потребление вычислений, хранение данных, сетевой обмен, число вызовов функций и резервирование ресурсов

- подходы к обеспечению отказоустойчивости, включая резервирование, повторные попытки, ограничение скорости, многоуровневое восстановление и план аварийного восстановления

- требования к технической документации облачного решения, включая схему архитектуры, описание сервисов, потоки данных, права доступа, риски, стоимость и порядок сопровождения

Уметь:

- уметь формализовать требования к облачному продукту при помощи описания бизнес-процесса, пользовательских сценариев и эксплуатационных ограничений в условиях транспортно-логистической задачи

- уметь выбирать модель размещения при помощи сравнения виртуальных машин, контейнеров, облачных функций и управляемых сервисов в условиях требований к стоимости, скорости изменения и устойчивости

- уметь проектировать бессерверную функцию при помощи схемы события, входных данных, результата и правил повторной обработки в условиях ограниченного времени выполнения

- уметь проектировать событийное взаимодействие при помощи очереди сообщений, темы, обработчика и правил идемпотентности в условиях асинхронной обработки транспортных событий

- уметь проектировать прикладной программный интерфейс при помощи прикладного шлюза и маршрутизации запросов в условиях публикации серверного сервиса во внешнюю среду

- уметь выбирать управляемое хранилище при помощи сравнения PostgreSQL, объектного хранилища, Redis и очереди сообщений в условиях разных типов данных и нагрузок

- уметь проектировать контейнерное размещение при помощи Kubernetes, реестра образов и описания ресурсов в условиях сервиса с несколькими компонентами

- уметь проектировать сетевую и защитную модель при помощи виртуальной сети, подсетей, ролей доступа и секретов в условиях разделения публичных и внутренних ресурсов

- уметь оценивать стоимость облачного решения при помощи калькулятора облачной платформы и таблицы потребления в условиях ограниченного бюджета эксплуатации

- уметь проектировать наблюдаемость при помощи метрик, журналов событий, трассировок и правил оповещения в условиях сопровождения распределенного облачного приложения

- уметь анализировать отказоустойчивость при помощи сценариев отказа, повторных попыток, резервирования и плана восстановления в условиях недоступности отдельного управляемого сервиса

- уметь готовить техническую документацию по облачной архитектуре при помощи схемы сервисов, потоков данных, прав доступа, оценки стоимости, рисков и порядка сопровождения

Владеть:

- навыком анализа требований к облачному программному продукту для транспортно-логистической отрасли

- навыком выбора облачных сервисов и модели размещения программных компонентов

- навыком проектирования бессерверных функций и событийного взаимодействия

- навыком проектирования архитектуры с управляемыми базами данных, очередями сообщений и объектным хранилищем

- навыком проектирования контейнерного размещения приложения средствами Kubernetes

- навыком проектирования сетевой изоляции, прав доступа и хранения секретов в облачной среде

- навыком оценки стоимости, отказоустойчивости и наблюдаемости облачного решения

- навыком подготовки технической документации по нативной облачной архитектуре

3. Объем дисциплины (модуля).

3.1. Общая трудоемкость дисциплины (модуля).

Общая трудоемкость дисциплины (модуля) составляет 2 з.е. (72 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 №1
Контактная работа при проведении учебных занятий (всего):	48	48
В том числе:		
Занятия лекционного типа	16	16
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 24 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	Облачные платформы в разработке программных продуктов Рассматриваемые вопросы: - роль облачных платформ в цифровизации транспортно-логистических процессов; - модели облачных услуг и распределение ответственности между поставщиком и разработчиком; - требования к облачному решению в условиях российских ограничений и импортозамещения.
2	Принципы нативной облачной разработки Рассматриваемые вопросы: - декомпозиция приложения, внешняя настройка, устойчивость к отказам и наблюдаемость; - состояние приложения, временные данные и управляемые зависимости; - переносимость архитектурного решения между облачными платформами.
3	Бессерверные вычисления и облачные функции Рассматриваемые вопросы: - облачная функция, событие запуска, среда выполнения и ограничение времени;

№ п/п	Тематика лекционных занятий / краткое содержание
	- холодный запуск, масштабирование, параметры окружения и права доступа; - сценарии применения функций в обработке заявок, уведомлений и транспортных событий.
4	Событийная архитектура облачного приложения Рассматриваемые вопросы: - событие, очередь, тема, обработчик и асинхронная обработка; - повторная доставка, идемпотентность и очередь недоставленных сообщений; - связь событийной модели с бессерверными функциями и управляемыми сервисами.
5	Прикладные шлюзы и публикация программных интерфейсов Рассматриваемые вопросы: - назначение прикладного шлюза при публикации серверных функций и сервисов; - маршрутизация запросов, проверка входных данных и ограничение частоты обращений; - согласование внешнего программного интерфейса с внутренними облачными ресурсами.
6	Управляемые хранилища данных в облачной архитектуре Рассматриваемые вопросы: - управляемые реляционные базы данных, объектное хранилище, Redis и очереди сообщений; - выбор хранилища по типу данных, скорости доступа, надежности и стоимости; - резервное копирование, срок хранения и восстановление данных.
7	Контейнеризация и реестр образов Рассматриваемые вопросы: - контейнер как единица упаковки приложения и его зависимостей; - реестр образов, версии, уязвимости и правила публикации; - сочетание контейнерного сервиса с бессерверными компонентами.
8	Управление контейнерными приложениями средствами Kubernetes Рассматриваемые вопросы: - узел, модуль, служба, развертывание, настройка и секрет; - масштабирование, обновление версии и проверка готовности приложения; - границы применения Kubernetes в сравнении с облачными функциями.
9	Сетевая архитектура облачного решения Рассматриваемые вопросы: - виртуальная сеть, подсеть, публичный доступ, частный доступ и балансировка нагрузки; - защищенное соединение между компонентами и внешними системами; - разделение контуров разработки, проверки и эксплуатации.
10	Управление доступом и защита секретов Рассматриваемые вопросы: - учетные записи, роли, политики доступа и принцип минимально необходимых прав; - ключи доступа, секреты, сертификаты и их жизненный цикл; - журналирование действий и контроль административных операций.
11	Безопасность и правовые условия применения облаков в Российской Федерации Рассматриваемые вопросы: - требования к персональным данным, критичной информации и выбору поставщика облачных услуг; - шифрование, резервное копирование, хранение журналов и контроль местоположения данных; - риски зависимости от поставщика и требования к переносимости решения.
12	Наблюдаемость облачных приложений Рассматриваемые вопросы: - метрики, журналы событий, трассировки и проверки доступности; - правила оповещения, контроль качества сервиса и разбор отказа; - связь наблюдаемости с эксплуатационной устойчивостью облачного продукта.
13	Стоимость облачного решения Рассматриваемые вопросы:

№ п/п	Тематика лекционных занятий / краткое содержание
	- потребление вычислений, хранение данных, сетевой обмен и число вызовов функций; - оценка стоимости первой версии и прогноз роста нагрузки; - технические решения для ограничения расходов и выявления лишнего потребления.
14	Отказоустойчивость и восстановление облачного продукта Рассматриваемые вопросы: - резервирование, повторные попытки, ограничение скорости и безопасная деградация; - отказ управляемого сервиса, потеря сообщения и недоступность внешней зависимости; - план восстановления и проверка готовности к аварийному событию.
15	Инфраструктура как описание архитектуры Рассматриваемые вопросы: - декларативное описание облачных ресурсов и контроль изменений; - применение Terraform, OpenTofu и GitOps для воспроизводимого управления инфраструктурой; - согласование инфраструктурного описания с требованиями безопасности и стоимости.
16	Документирование нативного облачного решения Рассматриваемые вопросы: - схема сервисов, потоки данных, права доступа и зависимости; - описание стоимости, рисков, отказных сценариев и порядка сопровождения; - критерии готовности архитектуры к реализации и эксплуатации.

4.2. Занятия семинарского типа.

Практические занятия

№ п/п	Тематика практических занятий/краткое содержание
1	Анализ облачного сценария транспортно-логистического продукта Студент выбирает прикладной процесс и описывает пользователей, внешние системы, данные, ограничения и ожидаемый результат. Формируется карта компонентов будущего облачного решения. Для каждого компонента фиксируются требования к доступности, времени ответа и защите данных.
2	Выбор модели размещения программных компонентов Студент сравнивает виртуальные машины, контейнерные сервисы, облачные функции и управляемые сервисы для выбранного сценария. Для каждого компонента указываются причины выбора модели размещения. Итоговая схема разделяет вычисления, хранение данных, обмен сообщениями и внешний доступ.
3	Проектирование облачных функций Студент описывает функции обработки событий, их входные данные, результат, ограничения времени и условия повторного запуска. Для функций задаются параметры окружения, права доступа и ожидаемые ошибки. Проектируется связь функций с очередью сообщений или прикладным шлюзом.
4	Проектирование событийного взаимодействия Студент строит схему событий, очередей, обработчиков и недоставленных сообщений. Для каждого события задаются источник, структура, идентификатор, правило идемпотентности и действие при ошибке. Схема проверяется на сценариях повторной доставки и потери внешней зависимости.
5	Проектирование программного интерфейса через прикладной шлюз Студент описывает маршруты внешнего программного интерфейса, методы, параметры, коды ответа и ограничения частоты обращений. Для каждого маршрута определяется вызываемая функция или внутренний сервис. Отдельно фиксируются правила проверки входных данных и обработки ошибок.

№ п/п	Тематика практических занятий/краткое содержание
6	Выбор управляемых хранилищ данных Студент сопоставляет типы данных сценария с PostgreSQL, объектным хранилищем, Redis и очередью сообщений. Для каждого хранилища задаются срок хранения, резервное копирование, требования к скорости доступа и стоимость. Итоговая схема показывает движение данных между компонентами.
7	Проектирование контейнерной части облачного решения Студент выделяет компоненты, которые целесообразно разместить в контейнерах. Описываются образ, реестр, переменные настройки, секреты и требования к ресурсам. Для каждого контейнерного сервиса задаются условия масштабирования и обновления версии.
8	Проектирование размещения в Kubernetes Студент описывает модули, службы, настройки, секреты и правила доступа для контейнерной части решения. На схеме показываются внутренние и внешние соединения сервисов. Отдельно фиксируются проверки готовности и допустимые действия при сбое модуля.
9	Проектирование сетевой изоляции облачного решения Студент определяет виртуальные сети, подсети, публичные и внутренние ресурсы. Для каждого соединения задаются источник, назначение, протокол и обоснование необходимости доступа. Схема проверяется на отсутствие лишних публичных ресурсов.
10	Проектирование прав доступа и защиты секретов Студент задает учетные записи сервисов, роли, политики доступа и перечень секретов. Для каждого секрета определяется место хранения, порядок обновления и круг компонентов, которым он доступен. Проверяется соблюдение принципа минимально необходимых прав.
11	Анализ требований безопасности и правовых ограничений Студент определяет категории данных, требования к их хранению, шифрованию, резервному копированию и журналированию. Для выбранного поставщика облачных услуг фиксируются ограничения применения в российском правовом контексте. Результат оформляется как карта рисков и мер снижения.
12	Проектирование наблюдаемости облачного приложения Студент выбирает метрики, журналы событий, трассировки и проверки доступности для каждого компонента решения. Для критичных нарушений задаются правила оповещения и порядок первичного разбора. Схема наблюдаемости связывается с требованиями доступности и времени ответа.
13	Оценка стоимости облачного решения Студент составляет таблицу потребления вычислений, хранилищ, сетевого обмена и вызовов функций. Для первой версии продукта рассчитывается приблизительная стоимость эксплуатации и выделяются основные источники расходов. Предлагаются меры ограничения лишнего потребления.
14	Анализ отказоустойчивости и восстановления Студент моделирует отказ функции, очереди, базы данных, контейнерного сервиса или внешней зависимости на уровне архитектурной схемы. Для каждого отказа задается ожидаемая реакция, повторная попытка, безопасная деградация или ручное восстановление. Итогом становится таблица отказных сценариев и мер восстановления.
15	Описание инфраструктуры как воспроизводимой модели Студент проектирует структуру декларативного описания ресурсов для Terraform или OpenTofu. Ресурсы группируются по сети, вычислениям, хранилищам, доступам и наблюдаемости. Отдельно фиксируются параметры, которые должны отличаться между средами разработки, проверки и эксплуатации.
16	Подготовка технической документации облачной архитектуры Студент собирает итоговую архитектурную схему, описание сервисов, потоки данных, права доступа, стоимость, риски, отказные сценарии и порядок сопровождения. Документация

№ п/п	Тематика практических занятий/краткое содержание
	проверяется на полноту для передачи команде реализации. Итоговый материал оформляется как связанное портфолио проектных решений.

4.3. Самостоятельная работа обучающихся.

№ п/п	Вид самостоятельной работы
1	Изучение рекомендованной литературы.
2	Подготовка к промежуточной аттестации.

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

№ п/п	Библиографическое описание	Место доступа
1	Аймен, Э. А. Облачные микросервисы в Kubernetes : руководство / Э. А. Аймен ; пер. с англ. В. С. Яценкова. — Москва : ДМК Пресс, 2025. — 276 с. — ISBN 978-5-93700-324-9. — Текст : электронный	Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/514873 (дата обращения: 18.06.2026)
2	Маркелов, А. А. Введение в технологию контейнеров и Kubernetes / А. А. Маркелов. — Москва : ДМК Пресс, 2019. — 194 с. — ISBN 978-5-97060-775-6. — Текст : электронный	Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/131702 (дата обращения: 18.06.2026)
3	Морган, Д. Linkerd на практике. Работа с приложениями в Kubernetes : руководство / Д. Морган, Флинн ; пер. с англ. В. И. Бахура. — Москва : ДМК Пресс, 2025. — 230 с. — ISBN 978-6-01123-649-2. — Текст : электронный	Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/514966 (дата обращения: 18.06.2026)
4	Трофименко, Е. В. Введение в облачные вычисления. Основные понятия : учебное пособие / Е. В. Трофименко. — Воронеж : ВГУ, 2020. — 59 с. — Текст : электронный	Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/432866 (дата обращения: 18.06.2026)

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

ЭБС Лань – <https://e.lanbook.com/>.

Образовательная платформа Юрайт – <https://urait.ru/>.

Единый реестр российских программ для ЭВМ и баз данных – <https://reestr.digital.gov.ru/reestr/>.

https://www.consultant.ru/document/cons_doc_LAW_157436/.

Документация Yandex Cloud – <https://yandex.cloud/ru/docs/>.

Документация VK Cloud – <https://cloud.vk.com/docs/>.

Документация Selectel – <https://docs.selectel.ru/>.

Документация Kubernetes – <https://kubernetes.io/docs/>.

Документация Terraform – <https://developer.hashicorp.com/terraform/docs>.

Документация OpenTofu – <https://opentofu.org/docs/>.

Документация Podman – <https://docs.podman.io/>.

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

Операционные системы – Astra Linux, ALT Linux, РЕД ОС, Debian GNU/Linux.

Облачные платформы – Yandex Cloud, VK Cloud, Selectel, MWS Cloud.

Контейнеры и управление ресурсами – Podman, Kubernetes.

Описание инфраструктуры – Terraform, OpenTofu.

Сопровождение проекта – Git, Markdown.

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:

Зачет в 1 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. Андриянова