

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



Рабочая программа дисциплины (модуля),
как компонент образовательной программы
высшего образования - программа магистратуры
по направлению подготовки
09.04.03 Прикладная информатика,
утвержденной первым проректором ФГАОУ ВО
РУТ (МИИТ) Тимониним В.С.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Современная микросервисная архитектура

Направление подготовки: 09.04.03 Прикладная информатика

Направленность (профиль): Информационные технологии управления
социально-экономическими системами

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: 564169

Подписал: заведующий кафедрой Каргина Лариса Андреевна

Дата: 23.09.2026

1. Общие сведения о дисциплине (модуле).

Целями освоения дисциплины (модуля) являются:

- формирование у студентов системного понимания эволюции архитектур программного обеспечения и фундаментальных принципов микросервисной архитектуры;
- приобретение практических навыков проектирования, разработки, контейнеризации и развертывания распределенных систем с использованием современных инструментов и платформ (Docker, Kubernetes);
- развитие компетенций в области обеспечения надежности, безопасности и наблюдаемости микросервисных приложений в условиях корпоративной ИТ-инфраструктуры.

Задачами дисциплины (модуля) являются:

- изучение методов декомпозиции бизнес-процессов и выделения сервисов на основе Domain-Driven Design (DDD);
- освоение паттернов распределенных систем: API Gateway, Service Discovery, Circuit Breaker, Saga, а также механизмов синхронного и асинхронного взаимодействия сервисов;
- формирование навыков работы с инструментами контейнеризации и оркестрации (Docker, Kubernetes) для управления жизненным циклом приложений;
- овладение практиками тестирования (контрактное тестирование), обеспечения безопасности (JWT, OAuth2) и наблюдаемости (логирование, метрики, трассировка) распределенных систем.

2. Планируемые результаты обучения по дисциплине (модулю).

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

ОПК-5 - Способен разрабатывать и модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем;

ПК-2 - Способен применять современные методы и инструментальные средства прикладной информатики для автоматизации и информатизации решения прикладных задач различных классов и создания ИС .

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

Знать:

- экономические модели оценки ИТ-решений (TCO, ROI, CAPEX/OPEX).
- показатели качества услуг (SLA, SLO, SLI) и их роль в управлении.
- методы бизнес-ориентированного выделения микросервисов (DDD, Bounded Contexts).
- стратегии управления проектами в распределённых средах (Agile, DevOps).
- подходы к управлению изменениями и миграцией архитектуры.

Уметь:

- проводить экономическое обоснование перехода на микросервисную архитектуру.
- формулировать требования к уровню качества услуг (SLA, SLO).
- анализировать метрики распределённой системы и принимать управленческие решения.
- разрабатывать стратегию поэтапного перехода на микросервисную архитектуру.
- оценивать риски и управлять изменениями при миграции.

Владеть:

- навыками экономического анализа ИТ-решений.
- навыками управления API-контрактами и их версионированием.
- навыками планирования инфраструктурных затрат и управления бюджетом.
- навыками принятия решений в условиях неопределённости при управлении распределёнными системами.

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 №3
Контактная работа при проведении учебных занятий (всего):	32	32
В том числе:		

Занятия лекционного типа	16	16
Занятия семинарского типа	16	16

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 112 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 архитектур программного обеспечения в контексте бизнес-задач. Рассматриваемые вопросы: - монолитная архитектура: определение, преимущества и недостатки; - сервис-ориентированная архитектура (SOA) как предшественник; - микросервисная архитектура: принципы, сравнение с монолитом; - критерии выбора архитектурного решения для различных бизнес-сценариев.
2	Экономические аспекты выбора микросервисной архитектуры Рассматриваемые вопросы: - совокупная стоимость владения (TCO): структура затрат, сравнение монолита и микросервисов; - капитальные и операционные затраты (CAPEX/OPEX); - возврат инвестиций (ROI) как инструмент обоснования архитектурных решений; - расчёт экономической эффективности перехода на микросервисы.
3	Бизнес-ориентированное проектирование микросервисов (DDD) Рассматриваемые вопросы: - предметная область и ограниченные контексты (Bounded Contexts); - паттерны декомпозиции: по бизнес-возможностям и по поддоменам; - выделение сервисов на основе бизнес-процессов; - проблема распределенных транзакций и паттерн Saga.
4	Управление API и контрактами в распределенных системах Рассматриваемые вопросы: - API как продукт и как контракт между сервисами; - принципы проектирования API: согласованность, обратная совместимость, версионирование; - openAPI (Swagger) как инструмент спецификации и документации; - управление жизненным циклом API.

№ п/п	Тематика лекционных занятий / краткое содержание
5	Наблюдаемость (Observability) как инструмент управления распределенными системами Рассматриваемые вопросы: - уровни качества услуг (SLA, SLO, SLI); - Error Budget (бюджет ошибок); - мониторинг и анализ показателей; - сбор и обработка журналов регистрации, метрик и трассировок; - принятие управленческих решений на основе данных наблюдаемости.
6	Обзор технологического стека и инструментов микросервисной архитектуры Рассматриваемые вопросы: - контейнеризация (Docker) и оркестрация (Kubernetes) как стандарты развертывания; - брокеры сообщений (RabbitMQ, Kafka) для асинхронного взаимодействия; - шлюзы API и сервис-дискавери; - критерии выбора технологического стека.
7	Организационные аспекты внедрения микросервисной архитектуры Рассматриваемые вопросы: - взаимосвязь структуры команд и архитектуры системы (закон Конвея); - роль организационной структуры в успешной реализации распределенных систем; - типология команд в контексте микросервисов: команды потоков создания ценности, платформенные команды, команды поддержки и команды сложных подсистем; - влияние выбранной модели командной работы на границы сервисов и эффективность разработки; - принципы организации взаимодействия между командами для минимизации кросс-командных зависимостей.
8	Стратегии миграции с монолитной архитектуры на микросервисную Рассматриваемые вопросы: - инкрементальный подход к миграции; - стратегия Strangler Fig (метод «удушения»); - миграция данных: стратегии и риски; - управление изменениями и коммуникация с заинтересованными сторонами; - оценка рисков и планирование миграции.

4.2. Занятия семинарского типа.

Практические занятия

№ п/п	Тематика практических занятий/краткое содержание
1	Анализ кейсов и развертывание простого веб-сервиса В результате практического занятия студент: - анализирует реальные кейсы перехода на микросервисы (Netflix, Amazon, российские компании); - разворачивает простой REST-сервис в контейнере с использованием Docker.
2	Расчёт экономической эффективности и проектирование границ сервисов В результате практического занятия студент: - сравнивает TCO для монолита и микросервисов на основе реальных данных; - выделяет сервисы на основе ограниченных контекстов DDD для гипотетической компании.
3	Практика выделения ограниченных контекстов (DDD) В результате практического занятия студент: - анализирует предметную область интернет-магазина; - выделяет ограниченные контексты и строит карту контекстов; - определяет границы сервисов.

№ п/п	Тематика практических занятий/краткое содержание
4	Проектирование API и согласование контрактов В результате практического занятия студент: - разрабатывает OpenAPI-спецификацию для выделенных сервисов; - согласовывает интерфейсы между командами; - управляет версиями API.
5	Анализ метрик и дашбордов наблюдаемости В результате практического занятия студент: - работает с готовыми дашбордами (Grafana); - анализирует метрики доступности и времени ответа; - определяет нарушения SLO и принимает управленческие решения.
6	Развертывание приложения в локальном кластере Kubernetes В результате практического занятия студент: - создает манифесты для развертывания сервисов в Kubernetes; - выполняет деплой приложения в Minikube с использованием готовых инструкций; - проверяет работоспособность развернутого приложения.
7	Деловая игра «Совет архитекторов»: проектирование организационной структуры для миграции на микросервисы В результате практического занятия студент: - анализирует гипотетическую компанию, переходящую с монолитной архитектуры на микросервисную; - разрабатывает предложения по реорганизации ИТ-команды на основе принципов командных топологий и ограниченных контекстов DDD; - формирует структуру подразделений, распределяет роли и ответственность; - определяет механизмы взаимодействия между командами; - оценивает риски и потенциальные проблемы, связанные с изменением организационной структуры.
8	Защита индивидуального проекта В результате практического занятия студент: - представляет и защищает индивидуальный проект, включающий экономическое обоснование перехода на микросервисы, план миграции и организационную структуру команд; - готовится к экзамену.

4.3. Самостоятельная работа обучающихся.

№ п/п	Вид самостоятельной работы
1	Работа с литературой
2	Работа с лекционным материалом
3	Подготовка к практическим занятиям
4	Подготовка к промежуточной аттестации.
5	Подготовка к текущему контролю.

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

№ п/п	Библиографическое описание	Место доступа
1	Баланов, А. Н. Бэкенд-разработка веб-приложений: архитектура, проектирование и управление проектами : учебное пособие для вузов / А. Н. Баланов. — 2-е изд., стер. — Санкт-Петербург : Лань, 2025. — 312 с. — ISBN 978-5-507-52472-3.	— Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/451820 (дата обращения: 23.09.2026).
2	Туманова, М. Б. Проектирование программных систем : учебное пособие / М. Б. Туманова, Е. К. Михайлова, Е. А. Муравьева. — Москва : РТУ МИРЭА, 2023. — 138 с. — ISBN 978-5-7339-2050-4.	— Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/398273 (дата обращения: 23.09.2026).

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

Научно-техническая библиотека РУТ (МИИТ)

Образовательная платформа «Юрайт»

Электронно-библиотечная система издательства «Лань»

Справочная система «Консультант Плюс»

Справочная система «Гарант»

Официальная документация Docker

Официальная документация Kubernetes

Официальная документация Spring

Официальная документация OpenTelemetry

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

1. Windows

2. Офисный пакет приложений Microsoft Office

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

Для проведения лекционных занятий необходима аудитория с мультимедиа аппаратурой. Для проведения практических занятий требуется аудитория, оснащенная мультимедиа аппаратурой и ПК с необходимым программным обеспечением, и подключением к сети интернет.

9. Форма промежуточной аттестации:

Экзамен в 3 семестре.

10. Оценочные материалы.

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

Авторы:

Е.И. Баранов

Согласовано:

Заведующий кафедрой ИСЦЭ

Л.А. Каргина

Председатель учебно-методической
комиссии

М.В. Ишханян