

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 РУТ (МИИТ)
Тимониным В.С.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Проектирование распределенных систем

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 РУТ (МИИТ)
ID подписи: 5665
Подписал: заведующий кафедрой Нутович Вероника
Евгеньевна
Дата: 01.09.2026

1. Общие сведения о дисциплине (модуле).

Дисциплина «Проектирование распределенных систем» направлена на формирование у обучающихся компетенций в области архитектуры высоконагруженных корпоративных программных продуктов. В ходе освоения курса студенты изучают фундаментальные ограничения распределенных вычислений, паттерны обеспечения отказоустойчивости, методы предметно-ориентированного проектирования и стратегии оптимизации производительности. Особое внимание уделяется работе в условиях технологического суверенитета с применением сертифицированного отечественного программного обеспечения и открытых стандартов. Практическая часть курса построена вокруг выполнения комплексного курсового проекта, в рамках которого обучающиеся проходят полный жизненный цикл архитектурного проектирования – от декомпозиции бизнес-требований и математического моделирования нагрузок до разработки спецификаций API и планирования емкости инфраструктуры. Выпускники получают навыки системного мышления и архитектурной аргументации, необходимые для решения задач корпоративного сектора.

Целью освоения дисциплины является формирование у обучающихся способности проектировать и разрабатывать распределенные высокопроизводительные программные продукты с применением методов оптимизации программного обеспечения для корпоративного рынка в условиях импортозамещения.

Для достижения поставленной цели в рамках дисциплины решается комплекс задач, направленных на формирование у обучающихся способности – изучать и анализировать архитектурные стили, протоколы взаимодействия и фундаментальные ограничения распределенных систем; применять методы предметно-ориентированного проектирования для декомпозиции сложных корпоративных предметных областей; проектировать топологии микросервисных и событийно-ориентированных кластеров с использованием паттернов отказоустойчивости; разрабатывать спецификации распределенных хранилищ данных и конвейеров асинхронного обмена сообщениями; проводить математическое моделирование нагрузок, выявлять узкие места и планировать емкость высокопроизводительных систем; интегрировать модули предиктивной аналитики и оптимизации бизнес-логики в контур распределенной системы; формировать комплексный пакет архитектурной документации с применением отечественных программных средств.

2. Планируемые результаты обучения по дисциплине (модулю).

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

ПК-2 - Способен проектировать и разрабатывать распределенные высокопроизводительные программные продукты с применением методов оптимизации программного обеспечения для корпоративного рынка.

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

Знать:

- архитектурные стили и паттерны построения распределенных систем, включая фундаментальные ограничения CAP-теоремы и теоремы PACELC;
- методологии предметного моделирования и декомпозиции сложных корпоративных систем на основе Domain-Driven Design;
- нотации, стандарты и лучшие практики визуализации программной архитектуры, включая многоуровневую модель C4 и стандарты описания контрактов API;
- методы выявления, формализации и математического выражения нефункциональных требований к высоконагруженным системам;
- паттерны обеспечения отказоустойчивости и устойчивости распределенных кластеров, такие как Circuit Breaker, Bulkhead, Retry with Backoff и Saga;
- принципы организации распределенных хранилищ данных, включая стратегии репликации, шардирования и партиционирования;
- архитектуру и принципы функционирования систем обмена сообщениями и брокеров событий;
- концепции контейнеризации приложений и оркестрации контейнерных кластеров;
- архитектурные паттерны и стратегии кэширования данных в распределенных средах;
- методы профилирования производительности программного обеспечения и выявления узких мест;
- основы применения методов машинного обучения и предиктивной аналитики для решения задач оптимизации бизнес-логики;
- методологии нагрузочного и стресс-тестирования распределенных систем и подходы к планированию емкости;
- принципы построения наблюдаемости распределенных систем, включая сбор логов, метрик и распределенного трейсинга;

- принципы обеспечения информационной безопасности и конфиденциальности данных на уровне архитектуры;

- специфику построения ИТ-инфраструктуры в условиях технологического суверенитета и этические нормы экологичности вычислений.

Уметь:

- декомпозировать предметную область и формулировать нефункциональные требования к распределенной системе при помощи методов Domain-Driven Design и нотаций архитектурного моделирования при условиях жестких корпоративных ограничений по SLA и требований технологического суверенитета;

- проектировать топологию микросервисного или событийно-ориентированного кластера при помощи паттернов устойчивости и инструментов визуализации архитектуры при условиях необходимости обеспечения отказоустойчивости и предотвращения каскадных сбоев;

- проектировать схемы распределенных хранилищ данных и конвейеры асинхронного обмена сообщениями при помощи DDL-скриптов для СУБД и спецификаций брокеров сообщений при условиях обеспечения требуемого уровня консистентности данных и высокой пропускной способности;

- разрабатывать и интегрировать модули предиктивной аналитики и оптимизации бизнес-логики при помощи стека Python и методов профилирования кода при условиях ограниченных вычислительных ресурсов и критической необходимости снижения latency;

- проводить математическое моделирование нагрузок и планирование емкости распределенной системы при помощи методов стресс-тестирования и анализа метрик производительности при условиях подготовки архитектуры к пиковым корпоративным нагрузкам;

- формировать комплексный пакет архитектурной и проектной документации при помощи сертифицированных отечественных офисных пакетов и стандартов описания API при условиях строгого соблюдения корпоративных стандартов качества и норм лицензирования.

Владеть:

- методами и инструментами визуализации архитектуры, включая многоуровневую модель С4 и стандарты описания контрактов API, применительно к корпоративным распределенным системам;

- технологиями контейнеризации, оркестрации и наблюдаемости распределенных кластеров с использованием отечественных и open-source решений;

- методиками стресс-тестирования, профилирования производительности и математического планирования емкости высоконагруженных программных продуктов.

3. Объем дисциплины (модуля).

3.1. Общая трудоемкость дисциплины (модуля).

Общая трудоемкость дисциплины (модуля) составляет 5 з.е. (180 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 №2
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 132 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>Архитектурные стили и фундаментальные ограничения распределенных систем Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - эволюция архитектурных стилей и фундаментальные ограничения, включая CAP-теорему; - методы выявления и математического выражения нефункциональных требований и метрик SLA; - протоколы межсервисного взаимодействия и концепции API Gateway.
2	<p>Предметное моделирование и стандартизация визуализации архитектуры Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - методологии предметного моделирования и декомпозиции систем на основе Domain-Driven Design; - выделение ограниченных контекстов, агрегатов и сущностей в корпоративной предметной области; - многоуровневая модель C4 и стандарты описания контрактов API.
3	<p>Протоколы взаимодействия и паттерны обеспечения отказоустойчивости Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - паттерны обеспечения отказоустойчивости и устойчивости кластеров по принципу Design for Failure; - механизмы компенсации распределенных транзакций и паттерны Circuit Breaker, Bulkhead и Saga; - архитектура систем обмена сообщениями и гарантии доставки событий в брокерах.
4	<p>Организация распределенных хранилищ данных и брокеров сообщений Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - принципы организации распределенных хранилищ данных, включая стратегии репликации и шардирования; - обеспечение консистентности данных в реляционных и NoSQL базах данных с учетом CAP-теоремы; - архитектура и паттерны топиков в Apache Kafka и RabbitMQ.
5	<p>Контейнеризация, оркестрация и наблюдаемость распределенных систем Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - концепции контейнеризации приложений и оркестрации кластеров с использованием Docker и Kubernetes; - управление состояниями сервисов, сервис-дискаверинг и конфигурирование сетевых политик; - принципы построения наблюдаемости, включая сбор логов, метрик и распределенного трейсинга.
6	<p>Стратегии кэширования и профилирование производительности Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - архитектурные паттерны и стратегии многоуровневого кэширования данных в распределенных средах; - проблемы инвалидации кэша и предотвращение эффекта thundering herd; - методы профилирования производительности, выявления узких мест и оптимизации ресурсов.
7	<p>Математическое моделирование нагрузок и применение методов машинного обучения Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - методологии нагрузочного и стресс-тестирования, подходы к математическому моделированию очередей; - планирование емкости и прогнозирование масштабируемости распределенных систем; - применение методов машинного обучения и градиентного бустинга для оптимизации бизнес-логики.
8	<p>Информационная безопасность и технологический суверенитет в корпоративных системах Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - принципы обеспечения информационной безопасности и конфиденциальности данных на уровне архитектуры;

№ п/п	Тематика лекционных занятий / краткое содержание
	- механизмы аутентификации, авторизации и шифрования трафика в распределенных кластерах; - специфика построения ИТ-инфраструктуры в условиях технологического суверенитета и этические нормы.

4.2. Занятия семинарского типа.

Практические занятия

№ п/п	Тематика практических занятий/краткое содержание
1	Анализ бизнес-кейса и выявление нефункциональных требований Студент изучает исходный бизнес-кейс корпоративного заказчика и выявляет ключевые функциональные ограничения модернизируемой системы. Он проводит интервью с условными стейкхолдерами для сбора требований к доступности и скорости отклика. Результаты анализа фиксируются в виде структурированного отчета, оформленного в сертифицированном офисном пакете.
2	Математическое выражение метрик SLA и расчет базовых нагрузок Обучающийся формализует собранные бизнес-требования в строгие математические метрики, включая целевые показатели RPS и Latency. Он рассчитывает допустимое время простоя системы в год на основе заданных параметров SLA. Итоговые расчеты сводятся в таблицу базовых нагрузок для дальнейшего планирования емкости.
3	Декомпозиция предметной области методами Domain-Driven Design Студент применяет методы предметного моделирования для разбиения сложной корпоративной системы на изолированные ограниченные контексты. Он выделяет ключевые агрегаты, сущности и доменные события, определяя границы ответственности будущих сервисов. Схема декомпозиции визуализируется с использованием стандартных нотаций архитектурного моделирования.
4	Визуализация архитектуры с применением модели C4 Обучающийся формирует многоуровневую карту проектируемой системы, последовательно создавая диаграммы контекста и контейнеров. Он определяет внешние интеграции и внутренние микросервисы, фиксируя их взаимодействие. Готовые диаграммы встраиваются в единый архитектурный реестр проекта.
5	Проектирование контрактов межсервисного взаимодействия Студент разрабатывает спецификации API для синхронного взаимодействия между выделенными микросервисами. Он описывает форматы запросов и ответов, определяя оптимальные протоколы передачи данных для каждого узла. Спецификации оформляются в машиночитаемом формате для последующей генерации документации.
6	Выбор и обоснование паттернов отказоустойчивости Обучающийся анализирует спроектированную топологию на предмет точек каскадного отказа. Он подбирает и математически обосновывает применение паттернов устойчивости для критических участков системы. Результаты обоснования компромиссов фиксируются в матрице архитектурных решений.
7	Проектирование топологии распределенных хранилищ данных Студент выбирает стратегии хранения данных для каждого ограниченного контекста с учетом требований к консистентности. Он проектирует схему шардирования и репликации для реляционной СУБД, оптимизируя ее под корпоративные нагрузки. Итоговая схема данных описывается в виде DDL-скриптов и диаграмм связей.
8	Моделирование асинхронного обмена событиями через брокеры сообщений Обучающийся проектирует конвейеры асинхронного взаимодействия между сервисами с использованием топиков брокера событий. Он определяет гарантии доставки сообщений и

№ п/п	Тематика практических занятий/краткое содержание
	стратегии обработки дублей на стороне потребителей. Архитектура потоков данных визуализируется и описывается в виде последовательности событий.
9	Контейнеризация сервисов и проектирование сетевой топологии Студент описывает конфигурации контейнеров для каждого микросервиса, определяя требования к вычислительным ресурсам. Он проектирует сетевые политики и механизмы сервис-дискаверинга для изолированного взаимодействия компонентов. Итоговая инфраструктурная спецификация подготавливается для развертывания в оркестраторе.
10	Архитектура наблюдаемости и распределенного трейсинга Обучающийся формирует план сбора телеметрии, определяя ключевые бизнес-метрики и технические индикаторы здоровья системы. Он проектирует схему распределенного трейсинга для отслеживания запросов сквозь цепочки микросервисов. Спецификация наблюдаемости включает правила агрегации логов и настройки алертинга.
11	Проектирование многоуровневой стратегии кэширования Студент выявляет в архитектуре узлы с высокой частотой чтений и проектирует для них уровни кэширования. Он разрабатывает алгоритмы инвалидации кэша, предотвращающие рассинхронизацию данных и эффект thundering herd. Стратегия кэширования математически обосновывается с точки зрения снижения нагрузки на СУБД.
12	Выявление узких мест и планирование оптимизации производительности Обучающийся проводит теоретическое профилирование спроектированных алгоритмов и запросов к базе данных. Он выявляет потенциальные bottlenecks и формирует план оптимизации, включая индексацию и рефакторинг кода. План оптимизации приоритизируется на основе оценки влияния на итоговую задержку системы.
13	Разработка спецификации модуля предиктивной аналитики Студент проектирует архитектуру интеграции ML-модели в контур распределенной системы для оптимизации бизнес-логики. Он определяет формат входных признаков, частоту переобучения модели и механизмы доставки предсказаний в сервисы. Спецификация включает требования к вычислительным ресурсам для инференса.
14	Математическое моделирование нагрузок и планирование емкости Обучающийся применяет теорию массового обслуживания для моделирования поведения системы при пиковых корпоративных нагрузках. Он рассчитывает необходимое количество инстансов микросервисов и узлов базы данных для обеспечения целевых метрик SLA. Результаты моделирования оформляются в виде плана горизонтального масштабирования.
15	Интеграция механизмов информационной безопасности в архитектуру Студент внедряет в спроектированную топологию шлюзы аутентификации и авторизации, применяя стандарты OAuth2 и JWT. Он определяет границы шифрования трафика и стратегии маскирования персональных данных в логах. Архитектурный реестр дополняется матрицей угроз и картой контроля доступа.
16	Формирование финального пакета документации и подготовка к защите Обучающийся консолидирует все спроектированные артефакты в единый пакет архитектурной документации. Он проверяет документацию на соответствие корпоративным стандартам и лицензионной чистоте используемого стека. Студент подготавливает презентацию и аргументацию для защиты принятых компромиссных решений перед комиссией.

4.3. Самостоятельная работа обучающихся.

№ п/п	Вид самостоятельной работы
1	Изучение рекомендованной литературы.
2	Выполнение курсового проекта.

3	Подготовка к промежуточной аттестации.
4	Подготовка к текущему контролю.

4.4. Примерный перечень тем курсовых проектов

Проектирование распределенной системы процессинга транзакций розничного банка с обеспечением строгой консистентности и отказоустойчивостью на уровне кластера.

Архитектура системы кредитного скоринга в реальном времени с интеграцией моделей градиентного бустинга и событийным конвейером обмена данными.

Проектирование платформы электронного биллинга и учета трафика мобильного оператора с шардированием хранилища данных и гарантированной доставкой событий.

Архитектура высоконагруженного маркетплейса с событийно-ориентированной микросервисной топологией и многоуровневым кэшированием каталога.

Проектирование агрегатора объектов недвижимости с поисковой выдачей, интеграцией ML-рекомендаций и асинхронным обменом событиями.

Архитектура распределенной системы управления цепями поставок с интеграцией модуля предиктивной аналитики спроса и математическим планированием емкости.

Проектирование системы диспетчеризации и динамического ценообразования сервиса такси с обработкой потоков событий в реальном времени.

Архитектура платформы промышленного интернета вещей (IIoT) для сбора и аналитики телеметрии с использованием оркестрации контейнеров.

Проектирование распределенной системы умного учета потребления электроэнергии с прогнозной аналитикой пиковых нагрузок и стратегиями репликации данных.

Проектирование межведомственной платформы электронного документооборота в условиях технологического суверенитета с применением сертифицированных средств криптозащиты.

Архитектура распределенной системы ведения электронных медицинских карт с обеспечением конфиденциальности данных и строгим SLA доступности.

Архитектура платформы потокового видео по запросу с адаптивным распределением нагрузок и стратегиями кэширования медиаконтента.

Проектирование масштабируемой платформы онлайн-обучения с поддержкой массовых трансляций и распределенной системой проверки заданий.

Архитектура микросервисной системы автоматического андеррайтинга и урегулирования убытков с применением паттернов Circuit Breaker и Bulkhead.

Проектирование системы бронирования авиабилетов с высоким SLA с механизмами компенсации распределенных транзакций и математическим моделированием очередей.

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

№ п/п	Библиографическое описание	Место доступа
1	Стин, в. М. Распределенные системы : сборник научных трудов / в. М. Стин, Э. С. Таненбаум ; перевод с английского В. А. Яроцкого. — Москва : ДМК Пресс, 2021. — 584 с. — ISBN 978-5-97060-708-4. — Текст : электронный	Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/240992 (дата обращения: 23.06.2026)
2	Карнелл, Д. Микросервисы Spring / Д. Карнелл, И. У. Санчес ; перевод с английского А. Н. Киселева. — Москва : ДМК Пресс, 2022. — 490 с. — ISBN 978-5-97060-971-2. — Текст : электронный	Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/241172 (дата обращения: 23.06.2026)
3	ймен, 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: 23.06.2026)
4	Кочер, П. С. Микросервисы и контейнеры Docker : руководство / П. С. Кочер ; перевод с английского А. Н. Киселева. — Москва : ДМК Пресс, 2019. — 240 с. — ISBN 978-5-97060-739-8. — Текст : электронный	Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/123710 (дата обращения: 23.06.2026)

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

Электронно-библиотечная система «Лань» – официальный сайт. URL: <https://www.lanbook.com> (дата обращения: 23.06.2026).

Электронно-библиотечная система «Юрайт» – официальный сайт. URL: <https://urait.ru> (дата обращения: 23.06.2026).

Postgres Professional – официальная русскоязычная документация. URL: <https://postgrespro.ru/docs> (дата обращения: 23.06.2026).

Apache Kafka – официальная документация платформы. URL: <https://kafka.apache.org/documentation/> (дата обращения: 23.06.2026).

CatBoost – документация библиотеки градиентного бустинга. URL: <https://catboost.ai/docs/en/> (дата обращения: 23.06.2026).

C4 Model – официальный сайт модели визуализации архитектуры. URL: <https://c4model.com/> (дата обращения: 23.06.2026).

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

Операционные системы – Astra Linux Special Edition / РЕД ОС / Альт Сервер.

Офисные пакеты – Р7-Офис / МойОфис (для подготовки архитектурных реестров, отчетов и презентаций по ГОСТ).

Среды разработки и профилирования – VS Code Community Edition / PyCharm Community Edition / JupyterLab / cProfile / py-spy.

Моделирование архитектуры – Draw.io / PlantUML / Structurizr.

Технологический стек ИИ и Data Science (Open Source / РФ) – Python 3.10+, Pandas, NumPy, Scikit-learn, CatBoost, Streamlit / Dash.

Распределенные хранилища данных и брокеры сообщений – Postgres Pro / Tarantool / Redis / Apache Kafka / RabbitMQ.

Контейнеризация и оркестрация – Docker Engine / Podman / Deckhouse Kubernetes Platform.

Инструменты стресс-тестирования и наблюдаемости – Yandex Tank / kb / Prometheus / Grafana / Jaeger / OpenTelemetry.

Работа с API и сетевыми протоколами – Postman / Hoppscotch / curl / gRPCurl.

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:

Курсовой проект во 2 семестре.

Экзамен во 2 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