Минобрнауки России

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Оренбургский государственный университет имени В.А. Бондаренко»

Кафедра математики и цифровых технологий

РАБОЧАЯ ПРОГРАММА

ДИСЦИПЛИНЫ

«Б1.Д.В.Э.4.3 Разработка распределенных корпоративных систем»

Уровень высшего образования

БАКАЛАВРИАТ

Направление подготовки

02.03.02 Фундаментальная информатика и информационные технологии
(код и наименование направления подготовки)

Разработка корпоративных систем и цифровых решений
(наименование направленности (профиля) образовательной программы)

Квалификация

Бакалавр

Форма обучения

Очная

Год набора 2026

Рабочая программа дисциплины «*Б1.Д.В.Э.4.3 Разработка распределенных корпоративных систем*» рассмотрена и утверждена на заседании кафедры

Кафедра математики и цифровых технологий
наименование кафедры

протокол № _____от "___" _____ 20__г.

Заведующий кафедрой
Кафедра математики и цифровых технологий А.Е. Шухман
наименование кафедры *подпись* *расшифровка подписи*

Исполнитель:
Доцент М.В.Ушакова
должность *подпись* *расшифровка подписи*

СОГЛАСОВАНО:
Председатель методической комиссии по направлению подготовки
02.03.02 Фундаментальная информатика и информационные технологии
И.П. Болодурина
код наименование *личная подпись* *расшифровка подписи*

Заведующий отделом формирования фонда и научной обработки документов
гл. библиограф Е.М. Автаева
личная подпись *расшифровка подписи*

Уполномоченный по качеству института С.Н. Морозова
личная подпись *расшифровка подписи*

№ регистрации _____

© Ушакова М.В., 2026
© ОГУ, 2026

2

1 Цели и задачи освоения дисциплины

Цель (цели) освоения дисциплины:

Формирование у обучающихся теоретических знаний и практических навыков проектирования, разработки, оптимизации и долгосрочного сопровождения масштабируемых распределенных информационных систем корпоративного уровня, автоматизирующих сложные бизнес-процессы.

Задачи:

1. Изучить архитектурные стили и модели распределенных корпоративных систем, требования к масштабируемости, отказоустойчивости и согласованности данных, а также паттерны декомпозиции монолитов и интеграции компонентов.
2. Изучить протоколы межсервисного взаимодействия (REST, gRPC, GraphQL, SOAP) и транспортные протоколы (HTTP/HTTPS, TCP/TLS, UDP, QUIC), их назначение, ограничения и критерии выбора для конкретных бизнес-задач.
3. Сформировать навыки разработки web-сервисов и их интеграции с внешними сервисами и учетными системами, включая установку и настройку прикладного программного обеспечения и модулей.
4. Освоить асинхронное взаимодействие компонентов распределенной системы посредством обмена сообщениями, а также управление распределенными транзакциями и обеспечение консистентности данных.
5. Сформировать умения оптимизировать и отлаживать серверное и сетевое программное обеспечение, обеспечивать отказоустойчивость распределенных систем в условиях высоких сетевых нагрузок, включая настройку параметров транспортных и прикладных протоколов.
6. Освоить инструменты тестирования API (Postman, curl) и анализа сетевого трафика (Wireshark, tcpdump) для контроля состояния и безопасности информационных потоков, проведения технического аудита и локализации уязвимостей и «узких мест» производительности.

2 Место дисциплины в структуре образовательной программы

Дисциплина относится к дисциплинам (модулям) по выбору вариативной части блока Д «Дисциплины (модули)»

Пререквизиты дисциплины: *Б1.Д.В.7 Разработка веб-приложений и распределенных систем, Б1.Д.В.8 Интеграция информационных систем, Б1.Д.В.9 Основы промышленной разработки программного обеспечения*

Постреквизиты дисциплины: *Отсутствуют*

3 Требования к результатам обучения по дисциплине

Процесс изучения дисциплины направлен на формирование следующих результатов обучения

Код и наименование формируемых компетенций	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения по дисциплине, характеризующие этапы формирования компетенций
ПК*-14 Способен выполнять работы по созданию (модификации) и сопровождению	ПК*-14-В-2 Выполняет кодирование на языках web-программирования ПК*-14-В-3 Выполняет установку и настройку прикладного	**Знать:** архитектурные стили и модели распределенных корпоративных систем, требования к масштабируемости, отказоустойчивости и согласованности данных; протоколы межсервисного взаимодействия и транспортные протоколы; принципы

Код и наименование формируемых компетенций	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения по дисциплине, характеризующие этапы формирования компетенций
информационных ресурсов (ИР)	программного обеспечения и модулей ПК*-14-В-4 Выполняет тестирование интеграции ИР с внешними сервисами и учетными системами с использованием взаимодействия компонентов распределенной системы	асинхронного взаимодействия компонентов распределенной системы; принципы управления распределенными транзакциями и обеспечения консистентности; принципы обеспечения отказоустойчивости распределенных систем; синтаксис и стандартные библиотеки языка web-программирования, компоненты программно-технических архитектур ИР; сетевые протоколы и основы web-технологий; основы информационной безопасности web-ресурсов. **Уметь:** разрабатывать web-сервисы и интегрировать их с внешними сервисами и учетными системами; писать программный код клиентской и серверной части ИР; оптимизировать и отлаживать серверное и сетевое программное обеспечение, обеспечивать отказоустойчивость распределенных систем в условиях высоких сетевых нагрузок; устанавливать и настраивать прикладное программное обеспечение и модули; составлять тест-планы и проводить интеграционное тестирование ИР, устранять обнаруженные несоответствия. **Владеть:** инструментами тестирования API и анализа сетевого трафика для контроля состояния и безопасности информационных потоков; навыками разработки web-сервисов и их интеграции с внешними сервисами и учетными системами; приемами организации асинхронного взаимодействия компонентов распределенной системы; методами оптимизации и отладки серверного и сетевого программного обеспечения и обеспечения отказоустойчивости распределенных систем; методами интеграционного тестирования ИР с внешними сервисами и учетными системами.

4 Структура и содержание дисциплины

4.1 Структура дисциплины

Общая трудоемкость дисциплины составляет 4 зачетные единицы (144 академических часа).

Вид работы	Трудоемкость, академических часов	
	7 семестр	всего
Общая трудоёмкость	**144**	**144**
Контактная работа:	**55**	**55**
Лекции (Л)	18	18
Лабораторные работы (ЛР)	34	34
Консультации	1	1
Индивидуальная работа и инновационные формы учебных занятий	1,5	1,5

Вид работы	Трудоемкость, академических часов	
	7 семестр	всего
Промежуточная аттестация (зачет, экзамен)	0,5	0,5
Самостоятельная работа: - выполнение курсового проекта (КП); - проработка и повторение лекционного материала и материала учебников и учебных пособий; - изучение разделов курса в системе электронного обучения; - подготовка к лабораторным занятиям; - подготовка к рубежному контролю.	**89** +	**89**
Вид итогового контроля (зачет, экзамен, дифференцированный зачет)	**экзамен**	

Разделы дисциплины, изучаемые в 7 семестре

№ раздела	Наименование разделов	Количество часов				
		всего	аудиторная работа			внеауд. работа
			Л	ПЗ	ЛР	
1	Архитектура и протоколы распределенных корпоративных систем	38	6		8	24
2	Разработка и интеграция web-сервисов. Асинхронное взаимодействие	36	4		10	22
3	Оптимизация и отказоустойчивость распределенных систем	34	4		8	22
4	Тестирование API и анализ сетевого трафика	16	2		4	10
5	Развертывание и сопровождение распределенных систем	20	2		4	14
	Итого:	144	18		34	92
	Всего:	144	18		34	92

4.2 Содержание разделов дисциплины

1. Архитектура и протоколы распределенных корпоративных систем

Архитектурные стили и модели распределенных корпоративных систем (монолит, микросервисы, SOA, событийно-ориентированная архитектура). Требования к масштабируемости, отказоустойчивости и согласованности данных; модели CAP и PACELC. Паттерны декомпозиции монолитов (по бизнес-возможностям, по поддоменам, strangler fig) и интеграции компонентов (API Gateway, BFF, агрегация, хореография и оркестрация). Протоколы межсервисного взаимодействия (REST, gRPC, GraphQL, SOAP) и критерии их выбора. Транспортные протоколы (HTTP/HTTPS, TCP/TLS, UDP, QUIC) и их параметры. Проектирование контрактов взаимодействия сервисов, версионирование интерфейсов, документирование API (OpenAPI). Обзор паттернов обеспечения консистентности данных (Saga, CQRS, 2PC, eventual consistency).

2. Разработка и интеграция web-сервисов. Асинхронное взаимодействие

Разработка web-сервисов на языке Python; клиентская и серверная части ИР. Фреймворк FastAPI: маршрутизация, схемы данных (Pydantic), автоматическая документация OpenAPI, асинхронная обработка запросов. Доступ к данным: SQLAlchemy ORM, миграции, работа с PostgreSQL. Установка и настройка прикладного ПО и модулей; управление зависимостями и конфигурирование. Интеграция ИР с внешними сервисами и учетными системами; обработка ошибок, таймауты, идемпотентность. Аутентификация и авторизация при интеграции (JWT, OAuth2, OpenID Connect), интеграция с LDAP и SSO. Асинхронное взаимодействие компонентов: брокеры сообщений (RabbitMQ, Kafka — обзор), очереди, pub/sub, событийные модели; очереди задач (Celery). Проектирование межсервисного взаимодействия с использованием современных стандартов.

3. Оптимизация и отказоустойчивость распределенных систем

Оптимизация и отладка серверного и сетевого ПО; профилирование производительности серверного кода, выявление «узких мест». Настройка параметров транспортных и прикладных сетевых протоколов (таймауты, размеры буферов, keep-alive, параметры TLS). Управление распределенными транзакциями и обеспечение консистентности данных: Saga (хореография и оркестрация, компенсирующие транзакции), CQRS, 2PC, eventual consistency. Обеспечение отказоустойчивости в условиях высоких сетевых нагрузок: резервирование, кластеризация, балансировка нагрузки; паттерны отказоустойчивости (таймауты, повторные попытки, circuit breaker, деградация функциональности, bulkhead); кэширование (Redis). Распределенное трассирование и логирование, health checks. «Инфраструктура как код» (IaC) и автоматизация развертывания. Защита информационных ресурсов в гетерогенных сетях: шифрование межсервисного трафика (TLS), управление секретами, защита API, аудит безопасности.

4. Тестирование API и анализ сетевого трафика

Инструменты тестирования API (Postman, curl): составление тест-планов, интеграционное тестирование ИР, проверка контрактов, обработка ошибок, автоматизация запросов, устранение несоответствий. Сетевые анализаторы (Wireshark, tcpdump): перехват и анализ трафика, диагностика сетевых сбоев, анализ аномалий трафика, профилирование информационных потоков корпоративного ПО. Контроль состояния и безопасности информационных потоков в гетерогенной среде. Технический аудит распределенной системы: локализация критических уязвимостей и «узких мест» производительности компонентов распределенной системы; анализ конфигураций, выявление небезопасных настроек, подготовка рекомендаций по устранению.

5. Развертывание и сопровождение распределенных систем

Сборка многокомпонентной системы: применение Docker Compose (сервисы, сети, тома, переменные окружения, healthcheck), оптимизация образов прикладных сервисов. CI/CD: этапы конвейера, автотесты, публикация образов, версионирование. Оркестрация развертывания: применение Kubernetes к распределенной системе (обзорно, с опорой на ранее изученное). Мониторинг и наблюдаемость: метрики, журналирование, трассировка, health checks, оповещения. Сопровождение: обновления без простоя (rolling, blue-green, canary), резервное копирование и восстановление системы, регламентные работы и снятие нагрузки.

4.3 Лабораторные работы

№ ЛР	№ раздела	Наименование лабораторных работ	Кол-во часов
1	1	Анализ архитектурных стилей и моделей распределенных корпоративных систем; обоснование выбора архитектуры и паттернов декомпозиции для заданного бизнес-процесса	4
2	1	Исследование межсервисного взаимодействия: сравнение REST, gRPC, GraphQL; настройка параметров транспортных протоколов (HTTP/HTTPS, TCP/TLS, QUIC)	4
3	2	Разработка web-сервиса на FastAPI: маршрутизация, схемы Pydantic, документация OpenAPI, работа с PostgreSQL через SQLAlchemy	4
4	2	Интеграция ИР с внешним сервисом и учетной системой: аутентификация и авторизация (JWT, OAuth2/OIDC), обработка ошибок, идемпотентность	4
5	2	Реализация асинхронного взаимодействия компонентов: RabbitMQ (очереди, pub/sub), очереди задач Celery	2
6	3	Оптимизация и отладка серверного кода; настройка параметров сетевых протоколов и выявление «узких мест» производительности	4
7	3	Управление распределенными транзакциями и обеспечение консистентности: Saga (хореография/оркестрация), компенсирующие транзакции	2
8	3	Обеспечение отказоустойчивости в условиях высоких нагрузок: таймауты, retries, circuit breaker, деградация, кэширование (Redis)	2

№ ЛР	№ раздела	Наименование лабораторных работ	Кол-во часов
9	4	Тестирование API (Postman, curl) и анализ сетевого трафика (Wireshark, tcpdump) для контроля безопасности информационных потоков	2
10	4	Технический аудит распределенной системы: локализация уязвимостей и «узких мест», анализ конфигураций, рекомендации по устранению	2
11	5	Сборка многокомпонентной системы в Docker Compose (web-сервис, PostgreSQL, RabbitMQ); сети, тома, переменные окружения	2
12	5	Развертывание и сопровождение: конвейер CI/CD (сборка, автотесты, публикация образов), мониторинг, журналирование, health checks	2
		Итого:	34

4.4 Курсовой проект (7 семестр)

Примерные темы курсового проекта:

1. Система управления заказами интернет-магазина.
2. Система бронирования ресурсов (переговорные, оборудование, транспорт).
3. Система управления складом.
4. Микросервисная система технической поддержки (help desk) с SLA
5. Сервис электронного документооборота с согласованием.
6. Платформа обмена данными с внешними сервисами и учётными системами (интеграционная шина).
7. Система бронирования и оплаты билетов (мероприятия, транспорт).
8. Система управления заявками на закупку (procurement).
9. Корпоративный портал с единой аутентификацией и бизнес-процессами.
10. Система обработки заказов с потоковой аналитикой и мониторингом.

5 Учебно-методическое обеспечение дисциплины

5.1 Основная литература

1. Щелоков, С. А. Проектирование распределенных информационных систем [Текст] : курс лекций: учебное пособие для студентов, обучающихся по программам высшего профессионального образования по направлению подготовки 230100.68 Информатика и вычислительная техника / С. А. Щелоков, Е. Н. Чернопрудова; М-во образования и науки Рос. Федерации, Федер. гос. бюджет. образоват. учреждение высш. проф. образования "Оренбург. гос. ун-т", Каф. прогр. обеспечения вычисл. техники и автоматизир. систем. - Оренбург : Университет, 2013. - 195 с. : ил. - Библиогр.: с. 194-195. - ISBN 978-5-4417-0332-1.

5.2 Дополнительная литература

1 Елисеев, А. И. Разработка веб-приложений с использованием фреймворка Flask : учебное пособие : в 2 частях / А. И. Елисеев, Ю. В. Минин, В. А. Гриднев ; Тамбовский государственный технический университет. – Тамбов : Тамбовский государственный технический университет (ТГТУ), 2020. – Часть 1. – 82 с. : ил., табл. – Режим доступа: по подписке. – URL: https://biblioclub.ru/index.php?page=book&id=720618 (дата обращения: 21.10.2026). – Библиогр. в кн. – ISBN 978-5-8265-2188-5 (ч. 1). – ISBN 978-5-8265-2187-8. – Текст : электронный.

2 Ушаков, Ю. А. Администрирование в информационных системах [Электронный ресурс] : лабораторный практикум для обучающихся по образовательным программам высшего образования по направлениям подготовки 27.03.03 и 27.04.03 Системный анализ и управление, 27.03.04 Управление в технических системах, 02.04.02 Фундаментальная информатика и информационные технологии, 02.00.00 Компьютерные и информационные науки, 10.00.00 Информационная безопасность,

09.00.00 Информатика и вычислительная техника / Ю. А. Ушаков, М. В. Ушакова, А. Л. Коннов; М-во образования и науки Рос. Федерации, Федер. гос. бюджет. образоват. учреждение высш. образования "Оренбург. гос. ун-т". - Ч. 1. - Электрон. текстовые дан. (1 файл: 9.84 Мб). - Оренбург : ОГУ, 2018. - 143 с. - Загл. с тит. экрана. - Adobe Acrobat Reader 6.0. - Режим доступа: http://artlib.osu.ru/web/books/metod_all/63264_20180227.pdf - ISBN 978-5-7410-1947-4.

3 Эндрюс, Г. Р. Основы многопоточного, параллельного и распределенного программирования = Foundations of Multithreaded, Parallel, and Distributed Programming [Текст] / Г. Р. Эндрюс. - М. : Вильямс, 2003. - 512 с. - Парал. тит. л. на англ. яз.. - Предм. указ.: с. 496-505. - ISBN 5-8459-0388-2

5.3 Периодические издания

Не используются

5.4 Интернет-ресурсы

https://fastapi.tiangolo.com/ru/ - «FastAPI», официальная документация фреймворка на русском языке: руководство, руководства по разработке web-сервисов. - Режим доступа: https://fastapi.tiangolo.com/ru/

https://www.rabbitmq.com/docs - «RabbitMQ», официальная документация брокера сообщений: учебные руководства по обмену сообщениями. - Режим доступа: https://www.rabbitmq.com/docs

https://docs.celeryq.dev - «Celery», официальная документация системы очередей задач: руководства по фоновым и периодическим операциям. - Режим доступа: https://docs.celeryq.dev

https://docs.docker.com - «Docker Docs», официальная документация по Docker и Docker Compose: руководства по контейнеризации приложений. - Режим доступа: https://docs.docker.com

https://microservices.io/patterns/ — «Microservice Architecture», каталог паттернов микросервисной архитектуры (декомпозиция, интеграция, отказоустойчивость, данные). — Режим доступа: https://microservices.io/patterns/

https://www.grpc.io/docs/ — «gRPC Documentation», официальная документация по протоколу gRPC: руководства по межсервисному взаимодействию. — Режим доступа: https://www.grpc.io/docs/

https://openedu.ru/ — «Открытое образование», каталог массовых открытых онлайн-курсов, в т.ч. по распределенным системам и микросервисной архитектуре. — Режим доступа: https://openedu.ru/

https://moodle.osu.ru - «Электронные курсы ОГУ», учебные материалы, тесты и методические указания по дисциплине. - Режим доступа: http://moodle.osu.ru

https://aist.osu.ru/cgi-bin/auth.cgi - Автоматизированная интерактивная система сетевого тестирования ОГУ (АИССТ). - Режим доступа: https://aist.osu.ru/

5.5 Программное обеспечение, профессиональные базы данных и информационные справочные системы

– Язык программирования Python (версии 3.11 и выше) — свободно распространяемое программное обеспечение; используется при разработке компонентов распределенной системы, выполнении лабораторных работ и курсового проекта (разделы 2–5). — Режим доступа: https://www.python.org/

– Фреймворк FastAPI — свободно распространяемый фреймворк разработки web-сервисов; используется при разработке REST API, интеграции и выполнении лабораторных работ (раздел 2, ЛР 3–4). — Режим доступа: https://fastapi.tiangolo.com/

– PostgreSQL — свободно распространяемая система управления базами данных; используется для хранения данных компонентов распределенной системы (раздел 2, ЛР 3; раздел 5, ЛР 11). — Режим доступа: https://www.postgresql.org/

– RabbitMQ — свободно распространяемый брокер сообщений; используется при асинхронном взаимодействии компонентов (раздел 2, ЛР 5; раздел 5, ЛР 11). — Режим доступа: https://www.rabbitmq.com/

– Apache Kafka — свободно распространяемая платформа потоковой обработки и обмена сообщениями; используется в обзорном режиме при изучении брокеров сообщений и событийных моделей (раздел 2). — Режим доступа: https://kafka.apache.org/

– Celery — свободно распространяемая система очередей задач; используется для фоновой и периодической обработки в распределенных компонентах (раздел 2, ЛР 5). — Режим доступа: https://docs.celeryq.dev/

– Redis — свободно распространяемое хранилище данных в оперативной памяти; используется для кэширования и снижения нагрузки при обеспечении отказоустойчивости (раздел 3, ЛР 8). — Режим доступа: https://redis.io/

– Docker Engine и Docker Compose — свободно распространяемая платформа контейнеризации; используется при сборке и развертывании многокомпонентной системы (раздел 5, ЛР 11). — Режим доступа: https://docs.docker.com/

– Kubernetes (kind / minikube) — свободно распространяемая система оркестрации контейнеров; используется в обзорном режиме при изучении развертывания распределенной системы (раздел 5, ЛР 12). — Режим доступа: https://kubernetes.io/ , https://kind.sigs.k8s.io/

– pytest (с плагинами pytest-asyncio, httpx) — свободно распространяемый фреймворк тестирования на Python; используется для автотестов в CI/CD-конвейере и интеграционного тестирования API (раздел 4, ЛР 12). — Режим доступа: https://docs.pytest.org/

– Postman — свободно распространяемый инструмент тестирования API; используется при интеграционном тестировании web-сервисов и проверке контрактов (раздел 4, ЛР 9). — Режим доступа: https://www.postman.com/

– curl — свободно распространяемая утилита командной строки для тестирования API и работы с сетевыми запросами; используется при тестировании API (раздел 4, ЛР 9). — Режим доступа: https://curl.se/

– Wireshark — свободно распространяемый сетевой анализатор; используется для анализа трафика и диагностики сетевых сбоев (раздел 4, ЛР 9). — Режим доступа: https://www.wireshark.org/

– tcpdump — свободно распространяемая утилита командной строки для перехвата и анализа сетевого трафика; используется для анализа трафика (раздел 4, ЛР 9). — Режим доступа: https://www.tcpdump.org/

– Visual Studio Code — свободно распространяемый редактор кода со средствами разработки на языке Python; используется при выполнении лабораторных работ и курсового проекта. — Режим доступа: https://code.visualstudio.com/

– СУБД-клиент DBeaver — свободно распространяемый инструмент работы с базами данных; используется для работы с PostgreSQL (раздел 2, ЛР 3). — Режим доступа: https://dbeaver.io/

– Яндекс.Браузер — свободно распространяемая программа для просмотра сайтов, входит в реестр отечественного ПО; используется для проверки работоспособности web-сервисов. — Режим доступа: https://yandex.ru/

– Университетская платформа электронного обучения «Электронные курсы ОГУ в системе обучения Moodle» — используется для доступа к учебным материалам, тестам и методическим указаниям по дисциплине. — Режим доступа: http://moodle.osu.ru

– Автоматизированная интерактивная система сетевого тестирования (АИССТ) — используется для рубежного контроля и тестирования. — Режим доступа: https://aist.osu.ru/cgi-bin/auth.cgi

6 Материально-техническое обеспечение дисциплины

Учебные аудитории для проведения занятий лекционного типа, для проведения групповых и индивидуальных консультаций, курсового проектирования, текущего контроля и промежуточной аттестации.

Аудитории оснащены комплектами ученической мебели, техническими средствами обучения, служащими для представления учебной информации большой аудитории.

Для проведения лабораторных занятий используется компьютерный класс, оснащенный компьютерной техникой с подключением к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду ОГУ; на компьютерах установлены средства разработки на языке Python, система управления базами данных PostgreSQL, платформа контейнеризации Docker, брокер сообщений RabbitMQ, система кэширования Redis, средства тестирования API (Postman, curl), сетевые анализаторы (Wireshark, tcpdump), фреймворк автотестов pytest.

Помещение для самостоятельной работы обучающихся оснащены компьютерной техникой, подключенной к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду ОГУ.