МИНОБРНАУКИ РОССИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Нижегородский государственный технический университет
им. Р.Е. Алексеева» (НГТУ)

Институт радиоэлектроники и информационных технологий (ИРИТ)
кафедра «Информатика и системы управления» (ИСУ)

ОДОБРЕНО:
на заседании кафедры разработчика
протокол № _1__ от 02.09.2026 г.
Зав. кафедрой
_____ О.П. Тимофеева
«02» _____09_____ 2026 г.

УТВЕРЖДЕН:
на заседании ученого совета ИРИТ
протокол № 9 от «22» ___09___ 2026 г.
Директор института
_____ В.П. Хранилов
«22 » _____09____ 2026 г.

Документ подписан электронной подписью в системе электронного документооборота НГТУ им. Р.Е. Алексеева
СВЕДЕНИЯ О СЕРТИФИКАТЕ ЭП
Сертификат 6e466de771b388c262d15ef91a6e3720
Владелец Иванович Евгений Геннадьевич
Действителен с 04.02.26 по 30.04.27

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Б1.В.ОД.1Сервис-ориентированные системы

(индекс и наименование дисциплины по учебному плану)

для подготовки бакалавров

Направление подготовки:	09.03.01 Информатика и вычислительная техника
Направленность:	Интеллектуальные системы обработки информации и управления
Форма обучения:	очная, заочная
Выпускающая кафедра	ИСУ
Разработчик (и):	Симонов Р.А., ассистент
регистрационный №	09.03.01-и-25

Начальник МО _____
подпись

Заведующая отделом комплектования НТБ _____
подпись

НИЖНИЙ НОВГОРОД, 2026 год

Содержание

1. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ), СООТНЕСЕННЫЕ С ПЛАНИРУЕМЫМИ РЕЗУЛЬТАТАМИ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ 3

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ 6

3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ 7

3.1. Распределение трудоемкости дисциплины по видам работ по семестрам 7

3.2. Содержание дисциплины, структурированное по темам 9

4. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ 17

4.1. Учебная литература 17

4.2. Справочно-библиографическая литература 17

5. ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ 17

5.1. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины (модуля) 18

5.1.1. Собственные образовательные ресурсы. Электронные ресурсы НТБ 18

5.1.2. Сторонние электронные образовательные и информационные ресурсы. Внешниересурсы 18

5.1.3. Перечень современных баз данных и информационных справочных систем 18

5.2. Перечень программного обеспечения и информационных справочных систем 19

6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ, НЕОБХОДИМОЕ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ 19

9. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ОБУЧАЮЩИМСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ 19

9.1. Общие методические рекомендации для обучающихся по освоению дисциплины, образовательные технологии 19

9.2. Методические указания для занятий лекционного типа 20

9.3. Методические указания по освоению дисциплины на лабораторных работах 20

9.5. Методические указания по самостоятельной работе обучающихся 21

10. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ КОНТРОЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ И КРИТЕРИИ ОЦЕНИВАНИЯ 21

11. АДАПТАЦИЯ РАБОЧЕЙ ПРОГРАММЫ ДЛЯ ЛИЦ С ОВЗ И ИНВАЛИДОВ 21

1. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ), СООТНЕСЕННЫЕ С ПЛАНИРУЕМЫМИ РЕЗУЛЬТАТАМИ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Настоящая рабочая программа дисциплины устанавливает планируемые результаты обучения по дисциплине, а также определяет содержание и виды учебных занятий и отчетности.

Программа разработана в соответствии с Федеральным государственным образовательным стандартом высшего образования (ФГОС ВО 3++) по направлению подготовки 09.03.01 «Информатика и вычислительная техника».

Освоение дисциплины вносит вклад в формирование профессиональной компетенции, предусмотренной образовательной программой.

Таблица 1. Компетенции, формируемые при освоении дисциплины

Код компетенции	**Формулировка компетенции**
Профессиональные (ПК)	
ПКС-1	Способен разрабатывать, тестировать и сопровождать ПО систем обработки информации и управления

Планируемые результаты обучения по дисциплине соотнесены с индикатором достижения компетенции ПКС-1 - ИПКС-1.1 и квалификационными требованиями профессионального стандарта 06.001 и 06.028.

Таблица 2. Индикаторы достижения компетенции и планируемые результаты обучения

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Код и наименование дескриптора достижения компетенции	Код ПС и ТФ	Квалификационные требования к выбранной ТФ	Оценочные материалы (ОМ)
ПКС-1. Способен разрабатывать, тестировать и сопровождать программное обеспечение систем обработки информации и управления	ИПКС-1.1. Разрабатывает программное обеспечение систем обработки информации и управления	**Знать:** - архитектуры сервис-ориентированных систем; - протоколы межсервисного взаимодействия; - спецификации описания сервисов; - требования к IT-инфраструктуре для создания сервис-ориентированных систем. **Уметь:** - проектировать, разрабатывать и отлаживать монолитные и микросервисныесервис-ориентированные системы. **Владеть:** - современными подходами, методами и средствами создания сервис-ориентированных систем.	06.001 D/03.6 «Решает задачу применения методов и средств проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов»	Трудовые действия: – выделение микросервисов и определение границ на основе предметной области и DDD; – проектирование контрактов и способов взаимодействия сервисов (API, события); – развёртывание и масштабирование микросервисов в контейнеризированной среде (Docker, Kubernetes); – настройка наблюдаемости, устойчивости и безопасности системы. Трудовые умения: – проектировать API и описывать спецификации (OpenAPI, Protobuf); – реализовывать паттерны устойчивости и распределённых транзакций (Saga, circuitbreaker, идемпотентность); – настраивать интеграцию через синхронные и асинхронные механизмы (REST, gRPC, Kafka,	1ЛР - Выделение микросервисов и проектирование контрактов; 2ЛР - Реализация синхронного и асинхронного взаимодействия; 3ЛР - Контейнеризация и оркестрация в Kubernetes; 4ЛР - Наблюдаемость: метрики, логи, трейсы; 5ЛР - Устойчивость и отказоустойчивость: паттерны и хаос-тесты; 6ЛР - Итоговый мини-проект: «Оформление заказа» в микросервисах.

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Код и наименование дескриптора достижения компетенции	Код ПС и ТФ	Квалификационные требования к выбранной ТФ	Оценочные материалы (ОМ)
				RabbitMQ); – автоматизировать CI/CD и проводить тесты отказоустойчивости. Трудовые знания: – принципы микросервисной архитектуры и отличия от монолита и SOA; – основы DDD (ограниченные контексты, единый язык, агрегаты); – паттерны взаимодействия, согласованности и устойчивости в распределённых системах; – инструменты контейнеризации, оркестрации, наблюдаемости и безопасности (Docker, Kubernetes, OpenTelemetry, Prometheus, API Gateway).	

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Учебная дисциплина Б1.В.ОД.1 «Сервис ориентированные системы» включена в перечень дисциплин части образовательной программы, формируемой участниками образовательных отношений, и определяет направленность (профиль) «Интеллектуальные системы обработки информации и управления».

Дисциплина изучается после освоения базовых дисциплин программирования, алгоритмов и математической подготовки и непосредственно опирается на результаты дисциплины Б1.В.ОД.5«Интеллектуальный анализ данных».

К началу изучения дисциплины «Сервис Ориентированные Системы» обучающийся должен владеть базовыми навыками программирования на одном из языков (Java, Go, Python, Node.js), понимать принципы клиент-серверного взаимодействия и HTTP-протокола, иметь представление о базах данных и транзакциях, знать основы работы с командной строкой и Git, а также иметь начальные навыки написания скриптов и работы с конфигурационными файлами (JSON/YAML).

В рамках дисциплины эти знания развиваются до уровня проектирования и эксплуатации распределённых микросервисных систем: углубляется понимание DDD и границ сервисов, осваиваются современные подходы к взаимодействию (REST, gRPC, асинхронные события), изучаются паттерны устойчивости и согласованности (Saga, circuitbreaker, идемпотентность), приобретаются практические навыки контейнеризации и оркестрации (Docker, Kubernetes), настройки наблюдаемости (OpenTelemetry, Prometheus, Grafana, Jaeger) и безопасности (API Gateway, mTLS, networkpolicies), а также автоматизации CI/CD и проведения хаос-тестов.

Результаты обучения, полученные при освоении дисциплины, необходимы при изучении дисциплины «Методология разработки DevOps» и выполнении курсовых и выпускных квалификационных работ, связанных с проектированием высоконагруженных и отказоустойчивых систем, а также в практической деятельности при разработке и сопровождении микросервисных приложений.

Особенностью дисциплины является практико-ориентированное освоение технологий микросервисной архитектуры: лабораторные работы построены вокруг реальных сценариев (оформление заказа, система уведомлений, бронирование), выполняются с использованием актуальных инструментов (Kafka/RabbitMQ, Kubernetes, ServiceMesh) и допускают реализацию как в локальной среде (Minikube, kind, k3d), так и в облачных платформах. Акцент сделан на выработке инженерных навыков: проектировании контрактов, обеспечении устойчивости к сбоям, настройке наблюдаемости и автоматизации деплоя.

3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

3.1. Распределение трудоемкости дисциплины по видам работ по семестрам

Общая трудоемкость дисциплины составляет 5 зачетные единицы, 180 часов. Распределение часов по видам работ для очной и заочной форм обучения представлено в таблице 3.

Таблица 3. Распределение трудоемкости дисциплины по видам работ для очной формы обучения

Вид учебной работы	**Трудоемкость в час**	
	Всего час.	**В т.ч. по семестрам 7 сем.**
Формат изучения дисциплины	с использованием элементов электронного обучения	с использованием элементов электронного обучения
Общая трудоемкость дисциплины по учебному плану	180	180
1. Контактная работа:	**74**	**74**
1.1 Аудиторная работа, в том числе:	**68**	**70**
лекции	34	34
лабораторные работы	34	34
практические занятия	–	–
1.2 Контрольно-самостоятельная работа	6	6
2. Самостоятельная работа	**106**	**106**
2.1 самостоятельная работа (самостоятельное изучение разделов, самоподготовка, подготовка к лабораторным занятиям и др.)	70	70
2.2 подготовка к контролю	**36**	**36**
3. Форма контроля	Экзамен	Экзамен

Продолжение таблицы 3. Распределение трудоемкости дисциплины по видам работ для заочной формы обучения

Вид учебной работы	Трудоемкость в час	
	Всего час.	**В т.ч. по семестрам 9 сем.**
Формат изучения дисциплины	с использованием элементов электронного обучения	с использованием элементов электронного обучения
Общая трудоемкость дисциплины по учебному плану	**180**	**180**
1. Контактная работа:	**26**	**26**
1.1 Аудиторная работа, в том числе:	20	20
лекции	10	10
лабораторные работы	10	10
практические занятия	–	–
1.2 Контрольно-самостоятельная работа	**6**	**6**
2. Самостоятельная работа	**154**	**154**
2.1 самостоятельная работа (самостоятельное изучение разделов, самоподготовка, подготовка к лабораторным занятиям и др.)	145	145
2.2 подготовка к контролю	**9**	**9**
3. Форма контроля	Экзамен	Экзамен

3.2. Содержание дисциплины, структурированное по темам

Таблица 4. Содержание дисциплины, структурированное по темам, для очной формы обучения

Планируемые (контролируемые) результаты освоения: код УК; ОПК; ПК и индикаторы достижения компетенций	Наименование разделов, тем	Виды учебной работы				Вид СРС	Наименование используемых активных и интерактивных образовательных технологий
		Контактная работа			Самостоятельная работа студентов (СРС), час		
		Лекции, час	Лабораторные работы, час	Практические занятия, час			
7 семестр							
Раздел 1.Введение и основы							
ПКС-1 - ИПКС-1.1	**Тема 1.1.**Эволюция архитектур: монолит, SOA, микросервисы. Сравнение по критериям: сложность, скорость релизов, масштабирование, стоимость владения.	1			2	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Тема 1.2.**Определение микросервиса: границы, автономность, слабая связанность, независимые релизы.	1			2	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Тема 1.3.** Когда микросервисы оправданы, а когда - нет. Антипаттерны: «распределённый монолит», «гранулярность по фреймворку».	1			2	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Тема 1.4.** Доменные модели и DDD: агрегаты, сущности, ограниченные контексты - как они задают границы микросервисов.	1			2	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
Раздел 2.Проектирование сервисов и контрактов							
ПКС-1 - ИПКС-1.1	**Тема 2.1.**API-first подход. REST vsgRPCvsGraphQL: когда что применять.	2			2	Подготовка к лекции[4.1.1, 4.1.2, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров и их сравнение.
ПКС-1 - ИПКС-1.1	**Тема 2.2.**Версионирование API, обратная совместимость, эволюция контрактов.	2			3	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.

Планируемые (контролируемые) результаты освоения: код УК; ОПК; ПК и индикаторы достижения компетенций	Наименование разделов, тем	Виды учебной работы				Вид СРС	Наименование используемых активных и интерактивных образовательных технологий
		Контактная работа			Самостоятельная работа студентов (СРС), час		
		Лекции, час	Лабораторные работы, час	Практические занятия, час			
ПКС-1 - ИПКС-1.1	**Тема 2.3.** Схемыданных: JSON Schema, OpenAPI, Protobuf.	1			2	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Тема 2.4.** Управление состоянием: statelessvsstateful, распределённые транзакции, Sagapattern.	1			2	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров и их сравнение.
ПКС-1 - ИПКС-1.1	**Тема 2.5.** Компенсация ошибок, идемпотентность, повторные попытки.	1			2	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Лабораторная работа №1.** Выделение микросервисов и проектирование контрактов		6		4	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	
Раздел 3.Взаимодействие и асинхронность							
ПКС-1 - ИПКС-1.1	**Тема 3.1.**Синхронные и асинхронные коммуникации: плюсы/минусы, сценарии.	2			2	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Тема 3.2.**Брокеры сообщений: Kafka, RabbitMQ — когда и что выбирать.	2			3	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Тема 3.3.** Паттерныинтеграции: publish-subscribe, request-reply, event-driven.	2			2	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор паттернов.
ПКС-1 - ИПКС-1.1	**Тема 3.4.** Согласованность: eventual consistency, гарантиидоставки (at-least-once, exactly-once).	1			2	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Тема 3.5.** Дедупликация событий, порядок сообщений, идемпотентность обработчиков.	1			2	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Лабораторная работа №2.** Реализация синхронного и асинхронного взаимодействия		6		4	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	
Раздел 4.Наблюдаемость, устойчивость и безопасность							
ПКС-1 - ИПКС-1.1	**Тема 4.1.**Логирование, метрики, трейсинг: OpenTelemetry, Prometheus, Grafana, Jaeger.	1			2	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.

Планируемые (контролируемые) результаты освоения: код УК; ОПК; ПК и индикаторы достижения компетенций	Наименование разделов, тем	Виды учебной работы				Вид СРС	Наименование используемых активных и интерактивных образовательных технологий
		Контактная работа			Самостоятельная работа студентов (СРС), час		
		Лекции, час	Лабораторные работы, час	Практические занятия, час			
ПКС-1 - ИПКС-1.1	**Тема 4.2.** Circuit breaker, bulkhead, retry, timeout — паттерныустойчивости.	2			2	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Тема 4.3.** Service Mesh (Istio, Linkerd): маршрутизация, политики, mTLS.	2			2	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Тема 4.4.** Безопасность: аутентификация/авторизация, API Gateway, networkpolicies.	2			2	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Тема 4.5.** CI/CD для микросервисов: стратегии сборки, версионирования, деплоя.	1			2	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Лабораторная работа №3.** Контейнеризация и оркестрация в Kubernetes. **Лабораторная работа №4.** Наблюдаемость: метрики, логи, трейсы.		10		6	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	
Раздел 5.Продвинутые темы и экосистема							
ПКС-1 - ИПКС-1.1	**Тема 5.1.**Domain-Driven Design вмикросервисах: bounded contexts, ubiquitous language.	2			2	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Тема 5.2.** Datamesh: децентрализованные данные как продукт.	1			2	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Тема 5.3.** ChaosEngineering: принципы и инструменты.	2			2	Подготовка к лекции[4.1.1, 4.1.4, 4.1.6, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Тема 5.4.** Costoptimization: управление ресурсами, spot-инстансы, автомасштабирование.	1			2	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Тема 5.5.** Тенденции: serverless-микросервисы, edge-вычисления.	1			2	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.

Планируемые (контролируемые) результаты освоения: код УК; ОПК; ПК и индикаторы достижения компетенций	Наименование разделов, тем	Виды учебной работы				Вид СРС	Наименование используемых активных и интерактивных образовательных технологий
		Контактная работа			Самостоятельная работа студентов (СРС), час		
		Лекции, час	Лабораторные работы, час	Практические занятия, час			
ПКС-1 - ИПКС-1.1	**Лабораторная работа №5.** Устойчивость и отказоустойчивость: паттерны и хаос-тесты. **Лабораторная работа №6.** Итоговый мини-проект в микросервисах.		12		6	Подготовка к лекции[4.1.1, 4.1.4, 4.1.6, 4.2.1, 4.2.2]	
ПКС-1 - ИПКС-1.1	**ИТОГО**	34	34	–	70		

Продолжение таблицы 4. Содержание дисциплины, структурированное по темам, для заочной формы обучения

Планируемые (контролируемые) результаты освоения: код УК; ОПК; ПК и индикаторы достижения компетенций	Наименование разделов, тем	Виды учебной работы				Вид СРС	Наименование используемых активных и интерактивных образовательных технологий
		Контактная работа			Самостоятельная работа студентов (СРС), час		
		Лекции, час	Лабораторные работы, час	Практические занятия, час			
9 семестр							
Раздел 1. Введение и основы							
ПКС-1 - ИПКС-1.1	**Тема 1.1.** Эволюция архитектур: монолит, SOA, микросервисы. Сравнение по критериям: сложность, скорость релизов, масштабирование, стоимость владения.				2	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Тема 1.2.**Определение микросервиса: границы, автономность, слабая связанность, независимые релизы.	2			3	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Тема 1.3.** Когда микросервисы оправданы, а когда - нет. Антипаттерны: «распределённый монолит», «гранулярность по фреймворку».				4	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Тема 1.4.** Доменные модели и DDD: агрегаты, сущности, ограниченные контексты - как они задают границы микросервисов.				4	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
Раздел 2.Проектирование сервисов и контрактов							
ПКС-1 - ИПКС-1.1	**Тема 2.1.**API-first подход. REST vsgRPCvsGraphQL: когда что применять.				5	Подготовка к лекции[4.1.1, 4.1.2, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров и их сравнение.
ПКС-1 - ИПКС-1.1	**Тема 2.2.**Версионирование API, обратная совместимость, эволюция контрактов.	2			7	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.

Планируемые (контролируемые) результаты освоения: код УК; ОПК; ПК и индикаторы достижения компетенций	Наименование разделов, тем	Виды учебной работы				Вид СРС	Наименование используемых активных и интерактивных образовательных технологий
		Контактная работа			Самостоятельная работа студентов (СРС), час		
		Лекции, час	Лабораторные работы, час	Практические занятия, час			
ПКС-1 - ИПКС-1.1	**Тема 2.3.** Схемыданных: JSON Schema, OpenAPI, Protobuf.				5	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Тема 2.4.** Управление состоянием: statelessvsstateful, распределённые транзакции, Sagapattern.				5	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров и их сравнение.
ПКС-1 - ИПКС-1.1	**Тема 2.5.** Компенсация ошибок, идемпотентность, повторные попытки.				5	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Лабораторная работа №1.** Выделение микросервисов и проектирование контрактов		2		5	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	
Раздел 3.Взаимодействие и асинхронность							
ПКС-1 - ИПКС-1.1	**Тема 3.1.**Синхронные и асинхронные коммуникации: плюсы/минусы, сценарии.				5	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Тема 3.2.**Брокеры сообщений: Kafka, RabbitMQ — когда и что выбирать.	2			5	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Тема 3.3.** Паттерныинтеграции: publish-subscribe, request-reply, event-driven.				5	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор паттернов.
ПКС-1 - ИПКС-1.1	**Тема 3.4.** Согласованность: eventual consistency, гарантиидоставки (at-least-once, exactly-once).				5	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Тема 3.5.** Дедупликация событий, порядок сообщений, идемпотентность обработчиков.				5	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Лабораторная работа №2.** Реализация синхронного и асинхронного взаимодействия		2		5	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	
Раздел 4.Наблюдаемость, устойчивость и безопасность							
ПКС-1 - ИПКС-1.1	**Тема 4.1.**Логирование, метрики, трейсинг: OpenTelemetry, Prometheus, Grafana, Jaeger.				5	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.

Планируемые (контролируемые) результаты освоения: код УК; ОПК; ПК и индикаторы достижения компетенций	Наименование разделов, тем	Виды учебной работы				Вид СРС	Наименование используемых активных и интерактивных образовательных технологий
		Контактная работа			Самостоятельная работа студентов (СРС), час		
		Лекции, час	Лабораторные работы, час	Практические занятия, час			
ПКС-1 - ИПКС-1.1	**Тема 4.2.** Circuit breaker, bulkhead, retry, timeout — паттерныустойчивости.				5	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Тема 4.3.** Service Mesh (Istio, Linkerd): маршрутизация, политики, mTLS.				5	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Тема 4.4.** Безопасность: аутентификация/авторизация, API Gateway, networkpolicies.	2			5	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Тема 4.5.** CI/CD для микросервисов: стратегии сборки, версионирования, деплоя.				5	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Лабораторная работа №3.** Контейнеризация и оркестрация в Kubernetes. **Лабораторная работа №4.** Наблюдаемость: метрики, логи, трейсы.		2		10	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	
Раздел 5.Продвинутые темы и экосистема							
ПКС-1 - ИПКС-1.1	**Тема 5.1.**Domain-Driven Design вмикросервисах: bounded contexts, ubiquitous language.				5	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Тема 5.2.** Datamesh: децентрализованные данные как продукт.				4	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Тема 5.3.** ChaosEngineering: принципы и инструменты.	2			7	Подготовка к лекции[4.1.1, 4.1.4, 4.1.6, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Тема 5.4.** Costoptimization: управление ресурсами, spot-инстансы, автомасштабирование.				4	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.
ПКС-1 - ИПКС-1.1	**Тема 5.5.** Тенденции: serverless-микросервисы, edge-вычисления.				5	Подготовка к лекции[4.1.1, 4.1.4, 4.2.1, 4.2.2]	Интерактивная лекция; разбор примеров.

Планируемые (контролируемые) результаты освоения: код УК; ОПК; ПК и индикаторы достижения компетенций	Наименование разделов, тем	Виды учебной работы				Вид СРС	Наименование используемых активных и интерактивных образовательных технологий
		Контактная работа			Самостоятельная работа студентов (СРС), час		
		Лекции, час	Лабораторные работы, час	Практические занятия, час			
ПКС-1 - ИПКС-1.1	**Лабораторная работа №5.** Устойчивость и отказоустойчивость: паттерны и хаос-тесты. **Лабораторная работа №6.** Итоговый мини-проект в микросервисах.		4		10	Подготовка к лекции[4.1.1, 4.1.4, 4.1.6, 4.2.1, 4.2.2]	
ПКС-1 - ИПКС-1.1	**ИТОГО**	10	10	–	145		

4. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

4.1. Учебная литература

4.1.1. Алпатов, А. Н. Архитектура клиент-серверных приложений : учебное пособие / А. Н. Алпатов, М. Ю. Волков. — Москва: РТУ МИРЭА, 2026. — 107 с. — ISBN 978-5-7339-2911-8. — Текст : электронный // Лань: электронно-библиотечная система. — URL: https://e.lanbook.com/book/523277

4.1.2. Алпатов, А. Н. Интерфейсы прикладного программирования : учебное пособие / А. Н. Алпатов. — Москва: РТУ МИРЭА, 2024. — 157 с. — ISBN 978-5-7339-2342-0. — Текст : электронный // Лань: электронно-библиотечная система. — URL: https://e.lanbook.com/book/457043

4.1.3. Баланов, А. Н. DevOps: интеграция и автоматизация : учебное пособие для вузов / А. Н. Баланов. — 3-е изд., стер. — Санкт-Петербург : Лань, 2026. — 240 с. — ISBN 978-5-507-54640-4. — Текст : электронный // Лань: электронно-библиотечная система. — URL: https://e.lanbook.com/book/509963

4.1.4.Баланов, А. Н. Построение микросервисной архитектуры и разработка высоконагруженных приложений : учебное пособие для вузов / А. Н. Баланов. — 2-е изд., стер. — Санкт-Петербург: Лань, 2025. — 244 с. — ISBN 978-5-507-52652-9. — Текст : электронный // Лань: электронно-библиотечная система. — URL: https://e.lanbook.com/book/456920

4.1.5. 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

4.1.6. Розенталь, К. Хаос-инжиниринг / К. Розенталь, Н. Джонс ; перевод с английского В. С. Яценкова. — Москва: ДМК Пресс, 2021. — 284 с. — ISBN 978-5-97060-796-1. — Текст : электронный // Лань: электронно-библиотечная система. — URL: https://e.lanbook.com/book/190728

4.2. Справочно-библиографическая литература

4.2.1. Баланов, А. Н. Управление жизненным циклом веб-разработки : учебное пособие / А. Н. Баланов. — Москва:Интермедиатор, 2025. — 148 с. — ISBN 978-5-91349-117-6. — Текст : электронный // Лань: электронно-библиотечная система. — URL: https://e.lanbook.com/book/518271

4.2.2. Баланов, А. Н. Секреты бэкенд-разработки : учебное пособие / А. Н. Баланов. — Москва:Интермедиатор, 2025. — 120 с. — ISBN 978-5-91349-114-5. — Текст: электронный // Лань: электронно-библиотечная система. — URL: https://e.lanbook.com/book/518268

5. ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Учебный процесс по дисциплине обеспечивается свободно распространяемым программным обеспечением и электронными образовательными ресурсами. Состав

программного обеспечения может обновляться с сохранением функциональной эквивалентности, необходимой для достижения планируемых результатов обучения.

5.1. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины (модуля)

5.1.1. Собственные образовательные ресурсы. Электронные ресурсы НТБ

–Научно-техническая библиотека НГТУ – https://www.nntu.ru/structure/view/podrazdeleniya/nauchno-tehnicheskaya-biblioteka/resursy

– Библиотека электронных учебников НГТУ – https://fdp.nntu.ru/knizhnaya-polka/

– Электронный каталог книг и периодических изданий (АИБС «МегаПро») – https://library.nntu.ru/megapro/web

5.1.2. Сторонние электронные образовательные и информационные ресурсы. Внешниересурсы

– gRPC Documentation - https://grpc.io/docs/

– Docker Documentation – https://docs.docker.com

– Kubernetes Documentation – https://kubernetes.io/docs/home/

– Apache Kafka Documentation –https://kafka.apache.org/44/getting-started/introduction/

–RabbitMQ Documentation –https://www.rabbitmq.com/docs

–RabbitMQ Simulator –https://tryrabbitmq.com/

5.1.3. Перечень современных баз данных и информационных справочных систем

Таблица 5. Перечень современных профессиональных баз данных и информационно-справочных систем

№	Наименование профессиональной базы данных, информационно-справочной системы	Доступ к ресурсу
1	Научная электронная библиотека eLIBRARY.RU	https://www.elibrary.ru/
2	Научная электронная библиотека открытого доступа «КиберЛенинка»	https://cyberleninka.ru/
3	arXiv – открытый архив научных публикаций	https://arxiv.org/
4	DBLP ComputerScienceBibliography	https://dblp.org/
5	SemanticScholar	https://www.semanticscholar.org/

5.2. Перечень программного обеспечения и информационных справочных систем

Таблица 6. Перечень программного обеспечения

Программное обеспечение, используемое в университете на договорной основе	Программное обеспечение свободного распространения и веб-сервисы
Специализированное проприетарное программное обеспечение для освоения дисциплины не требуется. При проведении занятий в компьютерных аудиториях используется базовое системное и офисное программное обеспечение, установленное в НГТУ.	Java, Go, Python, Node.js, Docker, Kubernetes, ApacheKafka, RabbitMQ, gRPC, REST, GraphQL, OpenAPI, Prometheus, Grafana, Jaeger, Istio, GitHubActions, GitLabCI, Git, VSCode

6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ, НЕОБХОДИМОЕ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ

Для проведения лекционных занятий используются учебные аудитории, оснащенные рабочим местом преподавателя, мультимедийным проектором или интерактивной панелью, экраном и доступом к информационно-телекоммуникационной сети «Интернет».

Лабораторные работы проводятся в компьютерной аудитории либо с использованием персональных компьютеров обучающихся, обеспечивающих работу с технологическим стеком и доступ к информационно-телекоммуникационной сети «Интернет».

9. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ОБУЧАЮЩИМСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

9.1. Общие методические рекомендации для обучающихся по освоению дисциплины, образовательные технологии

Дисциплина реализуется посредством контактной работы с обучающимися, самостоятельной работы и промежуточной аттестации. Контактная работа проводится в аудиториях и при необходимости, с использованием электронной информационно-образовательной среды университета.

Изучение дисциплины строится по принципу последовательного перехода от основ микросервисной архитектуры и выделения границ сервисов к проектированию взаимодействия, контейнеризации и оркестрации, а затем - к обеспечению устойчивости, наблюдаемости и автоматизации жизненного цикла микросервисов. Основными образовательными технологиями являются разбор архитектурных паттернов и реальных кейсов, проектирование контрактов API, практическая реализация сценариев взаимодействия (синхронных и асинхронных), развертывание в Kubernetes, внедрение инструментов наблюдаемости и устойчивости, проведение хаос-тестов, а также элементы проектного обучения на сквозном сценарии (например, «оформление заказа»).

При освоении дисциплины рекомендуется уделять основное внимание не воспроизведению готовых манифестов или шаблонов кода, а обоснованию архитектурных решений: выбору границ сервисов на базе DDD, подбору протокола взаимодействия под

бизнес-сценарий, проектированию отказоустойчивых сценариев (Saga, circuitbreaker, идемпотентность), настройке наблюдаемости для поиска узких мест, оценке компромиссов между доступностью, задержкой, сложностью эксплуатации и стоимостью инфраструктуры. Особое внимание стоит уделять воспроизводимости окружения (через Docker и Kubernetes-манифесты), автоматизации CI/CD и документированию принятых решений - именно эти навыки критичны для реальной инженерной практики.

9.2. Методические указания для занятий лекционного типа

Лекционный курс предполагает систематизированное изложение основных вопросов тематического плана: от эволюции архитектур и принципов DDD до паттернов взаимодействия, контейнеризации, наблюдаемости и автоматизации. Перед занятием обучающемуся рекомендуется ознакомиться с материалами предыдущей темы и указанными преподавателем первичными источниками (книги, официальная документация ключевых технологий). В ходе лекции необходимо фиксировать основные определения (микросервис, ограниченный контекст, eventualconsistency и др.), схемы архитектур (монолит vsмикросервисы, топологии взаимодействия), ключевые паттерны (Saga, circuitbreaker, bulkhead) и связи между рассматриваемыми подходами (например, как выбор протокола влияет на гарантии доставки и устойчивость).

При изучении тем, связанных с проектированием границ сервисов, асинхронным взаимодействием (Kafka/RabbitMQ), оркестрацией в Kubernetes, внедрением наблюдаемости (OpenTelemetry, Prometheus, Jaeger) и паттернами устойчивости, рекомендуется сопоставлять теоретические положения с документацией используемых инструментов и результатами последующих лабораторных работ: проверять, как описанные паттерны реализуются в коде и манифестах, как метрики и трейсы отражают поведение системы, а хаос-тесты - её реальную отказоустойчивость. Особое внимание стоит уделять фиксации принятых проектных решений и их обоснованию: это напрямую ложится в итоговую документацию проекта.

9.3. Методические указания по освоению дисциплины на лабораторных работах

Лабораторные работы выполняются преимущественно в формате репозиторияGit с набором файлов (Dockerfile, манифесты Kubernetes, код сервисов, скрипты тестирования) либо в среде контейнеризации (DockerCompose / Minikube / kind). До начала работы обучающийся должен ознакомиться с постановкой задачи, подготовить программное окружение (версии языков, библиотек, инструментов CLI) и исходные артефакты (спецификации API, схемы данных, базовые манифесты). Ресурсоёмкие операции (запуск кластера Kubernetes, нагрузочное тестирование, хаос-тесты) допускается выполнять в облачной среде или на удалённом ресурсе, предоставленном преподавателем.

Отчёт по лабораторной работе должен содержать постановку задачи, краткое описание проектируемых сервисов и выбранных архитектурных решений с обоснованием, перечень ключевых артефактов (фрагменты OpenAPI-спецификаций, фрагменты кода с реализацией паттернов, Dockerfile, Kubernetes-манифесты), параметры развёртывания и настройки инструментов (версии брокеров, параметры HPA, пороги circuitbreaker), результаты проверки сценариев (логи, метрики, трейсы, скриншоты дашбордов), сравнительные данные (время отклика, количество ошибок, поведение при сбоях) и содержательные выводы о соответствии результатов поставленным целям. При наличии

стохастических элементов (случайное отключение подов в хаос-тестах, вероятностные стратегии повторов) следует обеспечивать воспроизводимость результатов путём фиксации конфигурации тестов, сценариев сбоев, пороговых значений и существенных параметров среды (версии компонентов, лимиты ресурсов, политики масштабирования).

При работе с системами запрещается включать в отчеты и общедоступные репозитории реальные API-ключи, пароли и иные секреты. Инструменты должны запускаться с минимально необходимыми разрешениями, а эксперимент – выполняться в тестовой директории или учебном репозитории.

9.5. Методические указания по самостоятельной работе обучающихся

Самостоятельная работа обеспечивает подготовку к лекциям, лабораторным работам, текущему контролю и промежуточной аттестации. Она включает изучение рекомендуемой литературы и официальной технической документации, подготовку данных и программной среды, выполнение части вычислительных экспериментов, оформление отчетов и подготовку к защите лабораторных работ.

Для обучающихся заочной формы самостоятельная работа является основным видом учебной деятельности.

10. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ КОНТРОЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ И КРИТЕРИИ ОЦЕНИВАНИЯ

Оценочные материалы текущей и промежуточной аттестации по дисциплине оформляются отдельным комплектом в соответствии с макетом оценочных материалов НГТУ. Комплект обеспечивает контроль достижения планируемых результатов обучения, указанных в разделе 1 настоящей РПД.

В состав текущего контроля по дисциплине входят:

- Лабораторная работа №1. Выделение микросервисов и проектирование контрактов
- Лабораторная работа №2. Реализация синхронного и асинхронного взаимодействия
- Лабораторная работа №3. Контейнеризация и оркестрация в Kubernetes.
- Лабораторная работа №4. Наблюдаемость: метрики, логи, трейсы.
- Лабораторная работа №5. Устойчивость и отказоустойчивость: паттерны и хаос-тесты.
- Лабораторная работа №6. Итоговый мини-проект в микросервисах.

Промежуточная аттестация проводится в форме экзамена. К промежуточной аттестации допускаются обучающиеся, выполнившие обязательные лабораторные работы. Конкретные задания, показатели, критерии и шкалы оценивания приведены в комплекте оценочных материалов текущей и промежуточной аттестации.

11. АДАПТАЦИЯ РАБОЧЕЙ ПРОГРАММЫ ДЛЯ ЛИЦ С ОВЗ И ИНВАЛИДОВ

В соответствии с Федеральным законом от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации» профессиональное обучение и профессиональное образование обучающихся с ограниченными возможностями здоровья осуществляются на основе образовательных программ, адаптированных при необходимости для обучения указанных обучающихся.

Адаптация рабочей программы осуществляется при наличии заявления обучающегося и документов, определяющих необходимые специальные условия

обучения. Для инвалидов специальные условия определяются также с учетом индивидуальной программы реабилитации или абилитации.

При освоении дисциплины могут применяться электронные учебные материалы в доступных форматах, увеличенное время на выполнение отдельных видов работ, альтернативные способы представления результатов и консультации с использованием ЭИОС. Конкретные формы адаптации выбираются с учетом индивидуальных потребностей обучающегося и технических возможностей университета без снижения требований к планируемым результатам обучения.