

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
Факультет компьютерных технологий и прикладной математики

УТВЕРЖДАЮ

Проректор по учебной работе,
качеству образования – первый
проректор

Хагуров Т.А.

подпись

«30» мая 2025

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Б1.О.33«Микросервисная архитектура»

Направление подготовки 02.03.03 Математическое обеспечение и
администрирование информационных систем

Направленность (профиль) Технологии разработки программных систем

Форма обучения очная

Квалификация бакалавр

Краснодар 2025

Рабочая программа дисциплины «Микросервисная архитектура» составлена в соответствии с федеральным государственным образовательным стандартом высшего образования (ФГОС ВО) по направлению подготовки 02.03.03 Математическое обеспечение и администрирование информационных систем.

Программу составил(и):

Харченко Анна Владимировна, доцент, канд. пед наук

И.О. Фамилия, должность, ученая степень, ученое звание



подпись

Рабочая программа дисциплины «Микросервисная архитектура» утверждена на заседании кафедры информационных технологий протокол №15 от «14» мая 2025г.

Заведующий кафедрой (разработчика)

В. В. Подколзин



подпись

Рабочая программа обсуждена на заседании кафедры информационных технологий протокол №15 от «14» мая 2025г.

Заведующий кафедрой (выпускающей)

В. В. Подколзин



подпись

Утверждена на заседании учебно-методической комиссии факультета компьютерных технологий и прикладной математики протокол №4 от «23» мая 2025 г.

Председатель УМК факультета

А. В. Коваленко



подпись

Рецензенты:

Бегларян М. Е., Проректор по учебной работе, Краснодарский кооперативный институт (филиал) АНО ВО Центросоюза РФ «Российский университет кооперации»

Рубцов Сергей Евгеньевич, кандидат физико-математических наук, доцент кафедры математического моделирования ФГБОУ ВО «КубГУ»

1 Цели и задачи изучения дисциплины (модуля)

1.1 Цель освоения дисциплины

Цель дисциплины – освоить ключевые архитектурные стили и паттерны микросервисов, научиться проектировать и развертывать распределенные системы, а также применять полученные знания для внедрения микросервисных решений в реальных проектах. Это позволит оптимизировать программную архитектуру компании, проводить организационные трансформации, принимать стратегические решения и эффективно управлять командами разработки.

1.2 Задачи дисциплины

В рамках изучения дисциплины решаются следующие задачи:

- Изучение принципов микросервисной архитектуры и ее преимуществ перед монолитной.
- Приобретение практического опыта проектирования, разработки и тестирования микросервисов.
- Освоение инструментов и технологий для создания, развертывания и оркестрации микросервисов.
- Формирование навыков командной работы в условиях микросервисной архитектуры.
- Анализ проблем масштабирования и управления микросервисными системами, изучение способов их решения.
- Изучение методов мониторинга, логирования и отладки распределенных систем.
- Применение микросервисного подхода в реальных проектах для повышения их масштабируемости, гибкости и надежности.

1.3 Место дисциплины (модуля) в структуре образовательной программы

Дисциплина «Микросервисная архитектура» относится к «Обязательная часть» Блока 1 «Дисциплины (модули)» учебного плана.

1.4 Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения образовательной программы

Изучение данной учебной дисциплины направлено на формирование у обучающихся следующих компетенций:

ОПК-3 Способен применять современные информационные технологии, в том числе отечественные, при создании программных продуктов и программных комплексов различного назначения

ИД-1.ОПК-3 Аргументировано применяет современные информационные технологии, в том числе отечественные, при создании программных продуктов и программных комплексов различного назначения

Знать Возможности современных и перспективных средств разработки программных продуктов, технических средств
Методологии разработки программного обеспечения и технологии программирования
Методы и средства проектирования программного обеспечения
Методы и средства проектирования программных интерфейсов

Уметь Проводить оценку и обоснование рекомендуемых решений

Владеть	Проектирование программных интерфейсов Сбор, обработка, анализ и обобщение результатов экспериментов и исследований в соответствующей области знаний Деятельность, направленная на решение задач аналитического характера, предполагающих выбор и многообразие актуальных способов решения задач
ИД-2.ОПК-3	Ориентируется в современных положениях и концепциях прикладного и системного программного обеспечения, архитектуры компьютеров и сетей (в том числе и глобальных), технологии создания и сопровождения программных продуктов и программных комплексов
Знать	Возможности современных и перспективных средств разработки программных продуктов, технических средств Методологии разработки программного обеспечения и технологии программирования Методы и средства проектирования программного обеспечения Методы и средства проектирования программных интерфейсов
Уметь	Вырабатывать варианты реализации требований
Владеть	Оценка времени и трудоемкости реализации требований к программному обеспечению Оценка и согласование сроков выполнения поставленных задач Проектирование программных интерфейсов Оценка и согласование сроков выполнения поставленных задач Сбор, обработка, анализ и обобщение результатов экспериментов и исследований в соответствующей области знаний Деятельность, направленная на решение задач аналитического характера, предполагающих выбор и многообразие актуальных способов решения задач

2. Структура и содержание дисциплины

2.1 Распределение трудоёмкости дисциплины по видам работ

Общая трудоёмкость дисциплины составляет 2 зач. ед. (72 часов), их распределение по видам работ представлено в таблице

Вид учебной работы	Всего часов	Семестры (часы)					
		6					
Контактная работа, в том числе:	50,2	50,2					
Аудиторные занятия (всего):	48	48					
Занятия лекционного типа	16	16					
Лабораторные занятия	32	32					
Занятия семинарского типа (семинары, практические занятия)							
Иная контактная работа:	2,2	2,2					
Контроль самостоятельной работы (КСР)	2	2					
Промежуточная аттестация (ИКР)	0,2	0,2					
Самостоятельная работа, в том числе:	21,8	21,8					
Проработка учебного (теоретического) материала	8	8					
Выполнение индивидуальных заданий	10	10					

Подготовка к текущему контролю	3,8	3,8					
Контроль:							
Подготовка к экзамену							
Общая трудоемкость	час.	72	72				
	в том числе контактная работа	50,2	50,2				
	зач. ед	2	2				

2.2 Структура дисциплины

Распределение видов учебной работы и их трудоемкости по разделам дисциплины.

Разделы (темы) дисциплины, изучаемые в 6 семестре

№	Наименование разделов (тем)	Количество часов				
		Всего	Аудиторная работа			Внеаудиторная работа
			Л	ПЗ	ЛР	
1	2	3	4	5	6	7
1.	Введение в микросервисную архитектуру	8	2		4	2
2.	Принципы построения микросервисов	8	2		4	2
3.	Коммуникация между микросервисами	8	2		4	2
4.	Управление конфигурацией и развертыванием микросервисов	8	2		4	2
5.	Мониторинг и логирование в микросервисной архитектуре	8	2		4	2
6.	Тестирование микросервисов	8	2		4	4
7.	Проблемы и решения в микросервисной архитектуре	10	2		4	4
8.	Реализация микросервисной архитектуры в конкретных проектах	9,8	2		4	3,8
ИТОГО по разделам дисциплины		69,8	16		32	21,8
Контроль самостоятельной работы (КСР)		2				
Промежуточная аттестация (ИКР)		0,2				
Подготовка к текущему контролю						
Общая трудоемкость по дисциплине		72				

Примечание: Л – лекции, ПЗ – практические занятия/семинары, ЛР – лабораторные занятия, СРС – самостоятельная работа студента

2.3 Содержание разделов (тем) дисциплины

2.3.1 Занятия лекционного типа

№	Наименование раздела (темы)	Содержание раздела (темы)	Форма текущего контроля
1	2	3	4
1.	Введение в микросервисную архитектуру	Определение, преимущества и недостатки.	К
2.	Принципы построения микросервисов	Основные принципы и рекомендации по разработке микросервисов.	К, Т
3.	Коммуникация между микросервисами	Стратегии разделения функциональности, работа с данными, связывание сервисов.	К
4.	Управление конфигурацией и развертыванием микросервисов	Механизмы обеспечения безопасности и надежности микросервисов: аутентификация, авторизация, логирование, мониторинг и автоматическое восстановление.	К, Т

№	Наименование раздела (темы)	Содержание раздела (темы)	Форма текущего контроля
1	2	3	4
5.	Мониторинг и логирование в микросервисной архитектуре	Инструменты для построения и деплоя микросервисов: контейнеры, оркестрация и автоматизированные средства развертывания.	К
6.	Тестирование микросервисов	Приемы тестирования и отладки микросервисов в условиях большого масштаба и динамичности.	К, Т
7.	Проблемы и решения в микросервисной архитектуре	Монетизация и оптимизация микросервисов: масштабирование, монетизация, оптимизация исполнения.	К
8.	Реализация микросервисной архитектуры в конкретных проектах	Практические примеры использования микросервисов в реальных проектах: архитектурные схемы, задачи и технологии выполнения.	К, Т

Примечание: ЛР – отчет/защита лабораторной работы, КП - выполнение курсового проекта, КР - курсовой работы, РГЗ - расчетно-графического задания, Р - написание реферата, Э - эссе, К - коллоквиум, Т – тестирование, РЗ – решение задач.

2.3.2 Занятия семинарского типа

Не предусмотрены

Примечание: ЛР – отчет/защита лабораторной работы, КП - выполнение курсового проекта, КР - курсовой работы, РГЗ - расчетно-графического задания, Р - написание реферата, Э - эссе, К - коллоквиум, Т – тестирование, РЗ – решение задач.

2.3.3 Лабораторные занятия

№	Наименование раздела (темы)	Наименование лабораторных работ	Форма текущего контроля
1	2	3	4
1.	Введение в микросервисную архитектуру	Разработка простого микросервиса на основе REST API с использованием Spring Boot и Swagger.	РЗ
2.	Введение в микросервисную архитектуру	Разработка микросервисной архитектуры для приложения электронной коммерции с использованием Docker и Kubernetes.	РЗ
3.	Принципы построения микросервисов	Разработка микросервисов с использованием принципа единственной ответственности (Single Responsibility Principle, SRP)	РЗ
4.	Принципы построения микросервисов	Разработка микросервисов с использованием принципа инверсии зависимостей (Dependency Inversion Principle, DIP)	РЗ
5.	Коммуникация между микросервисами	Разработка системы мониторинга и логирования для микросервисной архитектуры с использованием ELK-стека (Elasticsearch, Logstash, Kibana).	РЗ
6.	Коммуникация между микросервисами	Разработка механизма обработки сообщений между микросервисами на основе Apache Kafka.	РЗ
7.	Управление конфигурацией и развертыванием микросервисов	Разработка скрипта автоматического развертывания микросервисов в Kubernetes с использованием Helm.	РЗ
8.	Управление конфигурацией и развертыванием микросервисов	Создание пайплайна CI/CD для микросервисной архитектуры с использованием GitLab CI/CD и Docker.	РЗ
9.	Мониторинг и логирование в микросервисной архитектуре	Настройка системы мониторинга Prometheus для сбора метрик микросервисов и их визуализации в	РЗ

№	Наименование раздела (темы)	Наименование лабораторных работ	Форма текущего контроля
1	2	3	4
		Grafana.	
10.	Мониторинг и логирование в микросервисной архитектуре	Разработка системы логирования микросервисов на базе ELK-стека (Elasticsearch, Logstash, Kibana) и анализ логов для выявления проблем в работе микросервисов.	РЗ
11.	Тестирование микросервисов	Написание автоматических тестов для микросервиса с использованием фреймворка для тестирования REST API, например, Postman или RestAssured.	РЗ
12.	Тестирование микросервисов	Исследование и применение методов тестирования микросервисов в условиях нагрузки, например, с помощью инструментов для генерации нагрузки, таких как JMeter или Gatling.	РЗ
13.	Проблемы и решения в микросервисной архитектуре	Мониторинг и трассировка в микросервисной архитектуре: разработка и анализ собственной системы мониторинга и трассировки для обнаружения проблем в микросервисной архитектуре и их решения.	РЗ
14.	Проблемы и решения в микросервисной архитектуре	Тестирование микросервисных приложений: разработка и применение инструментария для тестирования микросервисной архитектуры, включая автоматизированное тестирование, интеграционное тестирование и тестирование производительности, для выявления и решения проблем в микросервисной архитектуре.	РЗ
15.	Реализация микросервисной архитектуры в конкретных проектах	Реализация мониторинга микросервисной архитектуры с помощью инструментов Prometheus и Grafana на примере проекта онлайн-магазина	РЗ
16.	Реализация микросервисной архитектуры в конкретных проектах	Оптимизация микросервисной архитектуры с помощью инструментов Kubernetes и Docker на примере проекта облачного хранилища данных	РЗ

Примечание: ЛР – отчет/защита лабораторной работы, КП - выполнение курсового проекта, КР - курсовой работы, РГЗ - расчетно-графического задания, Р - написание реферата, Э - эссе, К - коллоквиум, Т – тестирование, РЗ – решение задач.

2.3.4 Примерная тематика курсовых работ (проектов)

Не предусмотрены

2.4 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)

№	Вид СРС	Перечень учебно-методического обеспечения дисциплины по выполнению самостоятельной работы
1	2	3
1	Изучение теоретического материала	Методические указания по организации самостоятельной работы студентов, утвержденные кафедрой информационных технологий, протокол №1 от 30.08.2019

2	Решение задач	Методические указания по организации самостоятельной работы студентов, утвержденные кафедрой информационных технологий, протокол №1 от 30.08.2019
---	---------------	---

Учебно-методические материалы для самостоятельной работы обучающихся из числа инвалидов и лиц с ограниченными возможностями здоровья (ОВЗ) предоставляются в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа,
- в форме аудиофайла,
- в печатной форме на языке Брайля.

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа,
- в форме аудиофайла.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

3. Образовательные технологии

В соответствии с требованиями ФГОС в программа дисциплины предусматривает использование в учебном процессе следующих образовательные технологии: чтение лекций с использованием мультимедийных технологий; метод малых групп, разбор практических задач и кейсов.

При обучении используются следующие образовательные технологии:

– Технология коммуникативного обучения – направлена на формирование коммуникативной компетентности студентов, которая является базовой, необходимой для адаптации к современным условиям межкультурной коммуникации.

– Технология разноуровневого (дифференцированного) обучения – предполагает осуществление познавательной деятельности студентов с учётом их индивидуальных способностей, возможностей и интересов, поощряя их реализовывать свой творческий потенциал. Создание и использование диагностических тестов является неотъемлемой частью данной технологии.

– Технология модульного обучения – предусматривает деление содержания дисциплины на достаточно автономные разделы (модули), интегрированные в общий курс.

– Информационно-коммуникационные технологии (ИКТ) - расширяют рамки образовательного процесса, повышая его практическую направленность, способствуют интенсификации самостоятельной работы учащихся и повышению познавательной активности. В рамках ИКТ выделяются 2 вида технологий:

– Технология использования компьютерных программ – позволяет эффективно дополнить процесс обучения языку на всех уровнях.

– Интернет-технологии – предоставляют широкие возможности для поиска информации, разработки научных проектов, ведения научных исследований.

– Технология индивидуализации обучения – помогает реализовывать личностно-ориентированный подход, учитывая индивидуальные особенности и потребности учащихся.

– Проектная технология – ориентирована на моделирование социального взаимодействия учащихся с целью решения задачи, которая определяется в рамках профессиональной подготовки, выделяя ту или иную предметную область.

– Технология обучения в сотрудничестве – реализует идею взаимного обучения, осуществляя как индивидуальную, так и коллективную ответственность за решение учебных задач.

– Игровая технология – позволяет развивать навыки рассмотрения ряда возможных способов решения проблем, активизируя мышление студентов и раскрывая личностный потенциал каждого учащегося.

– Технология развития критического мышления – способствует формированию разносторонней личности, способной критически относиться к информации, умению отбирать информацию для решения поставленной задачи.

Комплексное использование в учебном процессе всех вышеперечисленных технологий стимулируют личностную, интеллектуальную активность, развивают познавательные процессы, способствуют формированию компетенций, которыми должен обладать будущий специалист.

Основные виды интерактивных образовательных технологий включают в себя:

– работа в малых группах (команде) - совместная деятельность студентов в группе под руководством лидера, направленная на решение общей задачи путём творческого сложения результатов индивидуальной работы членов команды с делением полномочий и ответственности;

– проектная технология - индивидуальная или коллективная деятельность по отбору, распределению и систематизации материала по определенной теме, в результате которой составляется проект;

– анализ конкретных ситуаций - анализ реальных проблемных ситуаций, имевших место в соответствующей области профессиональной деятельности, и поиск вариантов лучших решений;

– развитие критического мышления – образовательная деятельность, направленная на развитие у студентов разумного, рефлексивного мышления, способного выдвинуть новые идеи и увидеть новые возможности.

Подход разбора конкретных задач и ситуаций широко используется как преподавателем, так и студентами во время лекций, лабораторных занятий и анализа результатов самостоятельной работы. Это обусловлено тем, что при исследовании и решении каждой конкретной задачи имеется, как правило, несколько методов, а это требует разбора и оценки целой совокупности конкретных ситуаций.

Семестр	Вид занятия	Используемые интерактивные образовательные технологии	количество интерактивных часов
	ЛР	Практические занятия в режимах взаимодействия «преподаватель – студент» и «студент – студент»	6
Итого			6

Примечание: Л – лекции, ПЗ – практические занятия/семинары, ЛР – лабораторные занятия, СРС – самостоятельная работа студента

Темы, задания и вопросы для самостоятельной работы призваны сформировать навыки поиска информации, умения самостоятельно расширять и углублять знания, полученные в ходе лекционных и практических занятий.

Подход разбора конкретных ситуаций широко используется как преподавателем, так и студентами при проведении анализа результатов самостоятельной работы.

Для лиц с ограниченными возможностями здоровья предусмотрена организация консультаций с использованием электронной почты.

Для лиц с нарушениями зрения:

– в печатной форме увеличенным шрифтом,

– в форме электронного документа.

Для лиц с нарушениями слуха:

– в печатной форме,

– в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

– в печатной форме,

– в форме электронного документа.

Для лиц с ограниченными возможностями здоровья предусмотрена организация консультаций с использованием электронной почты.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

4. Оценочные и методические материалы

4.1 Оценочные средства для текущего контроля успеваемости и промежуточной аттестации

Оценочные средства предназначены для контроля и оценки образовательных достижений обучающихся, освоивших программу учебной дисциплины «Микросервисная архитектура».

Оценочные средства включает контрольные материалы для проведения **текущего контроля** в форме тестовых заданий, разноуровневых заданий и **промежуточной аттестации** в форме вопросов и заданий к зачету.

Оценочные средства для инвалидов и лиц с ограниченными возможностями здоровья выбираются с учетом их индивидуальных психофизических особенностей.

– при необходимости инвалидам и лицам с ограниченными возможностями здоровья предоставляется дополнительное время для подготовки ответа на экзамене;

– при проведении процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья предусматривается использование технических средств, необходимых им в связи с их индивидуальными особенностями;

– при необходимости для обучающихся с ограниченными возможностями здоровья и инвалидов процедура оценивания результатов обучения по дисциплине может проводиться в несколько этапов.

Процедура оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья по дисциплине (модулю) предусматривает предоставление информации в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

– в печатной форме увеличенным шрифтом,

– в форме электронного документа.

Для лиц с нарушениями слуха:

– в печатной форме,

– в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

– в печатной форме,

– в форме электронного документа.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

Структура оценочных средств для текущей и промежуточной аттестации

№ п/п	Контролируемые разделы (темы) дисциплины*	Код контролируемой	Наименование оценочного средства
-------	---	--------------------	----------------------------------

		компетенции (или ее части)	Текущий контроль	Промежуточная аттестация
1	Введение в микросервисную архитектуру	ИД-1.ОПК-3 ИД-2.ОПК-3	Задания для самостоятельного решения 1-2	Теоретический вопрос 1-4
2	Принципы построения микросервисов	ИД-1.ОПК-3 ИД-2.ОПК-3	Задания для самостоятельного решения 3-4	Теоретический вопрос 5-8
3	Коммуникация между микросервисами	ИД-1.ОПК-3 ИД-2.ОПК-3	Задания для самостоятельного решения 5-6	Теоретический вопрос 9-12
4	Управление конфигурацией и развертыванием микросервисов	ИД-1.ОПК-3 ИД-2.ОПК-3	Задания для самостоятельного решения 7-8	Теоретический вопрос 13-16
5	Мониторинг и логирование в микросервисной архитектуре	ИД-1.ОПК-3 ИД-2.ОПК-3	Задания для самостоятельного решения 9-10	Теоретический вопрос 17-20
6	Тестирование микросервисов	ИД-1.ОПК-3 ИД-2.ОПК-3	Задания для самостоятельного решения 11-12	Теоретический вопрос 21-24
7	Проблемы и решения в микросервисной архитектуре	ИД-1.ОПК-3 ИД-2.ОПК-3	Задания для самостоятельного решения 13-14	Теоретический вопрос 25-28
8	Реализация микросервисной архитектуры в конкретных проектах	ИД-1.ОПК-3 ИД-2.ОПК-3	Задания для самостоятельного решения 15-16	Теоретический вопрос 29-32

Показатели, критерии и шкала оценки сформированных компетенций

Соответствие **пороговому уровню** освоения компетенций планируемым результатам обучения и критериям их оценивания (оценка: **удовлетворительно /зачтено**):

ОПК-3 Способен применять современные информационные технологии, в том числе отечественные, при создании программных продуктов и программных комплексов различного назначения

ИД-1.ОПК-3 Аргументировано применяет современные информационные технологии, в том числе отечественные, при создании программных продуктов и программных комплексов различного назначения

Знать Современные и перспективные инструменты разработки ПО, включая отечественные платформы и фреймворки для микросервисов (например, Методологии и технологии разработки ПО, включая принципы декомпозиции монолитных систем на микросервисы.

Принципы и инструменты проектирования программного обеспечения, включая шаблоны проектирования микросервисов (Service Discovery, Circuit Breaker, Saga и др.).

Методы проектирования программных интерфейсов (REST/gRPC API, event-driven взаимодействие между микросервисами).

Уметь анализировать и обосновывать выбор технологических решений, включая оценку целесообразности перехода на микросервисную архитектуру.

Владеть Навыками проектирования программных интерфейсов для взаимодействия микросервисов.

Методами сбора, обработки, анализа и обобщения данных исследований в распределенных системах.

Подходами к решению аналитических задач, включая проблемы масштабируемости, отказоустойчивости и мониторинга микросервисных систем.

ИД-2.ОПК-3 ***Ориентируется в современных положениях и концепциях прикладного и системного программного обеспечения, архитектуры компьютеров и сетей (в том числе и глобальных), технологии создания и сопровождения программных продуктов и программных комплексов***

Знать *Современные и перспективные средства разработки ПО, включая инструменты оркестрации микросервисов (Kubernetes, Nomad, OpenShift). Методологии и технологии программирования, включая подходы к разработке и тестированию микросервисов (Contract Testing, Canary Releases).*

Методы проектирования программного обеспечения, включая стратегии развертывания и управления жизненным циклом микросервисов.

Принципы проектирования программных интерфейсов, включая API-шлюзы и межсервисную асинхронную коммуникацию.

Уметь *Формулировать варианты реализации технических требований, включая выбор между монолитной и микросервисной архитектурой.*

Владеть *Методами оценки трудоемкости и сроков реализации ПО, с учетом особенностей распределенных систем (задержки сети, согласованность данных).*

Навыками согласования сроков выполнения задач в условиях параллельной разработки множества микросервисов.

Техниками проектирования программных интерфейсов для обеспечения слабой связанности микросервисов.

Методами сбора, анализа и обобщения результатов исследований в контексте мониторинга и логирования микросервисных систем (ELK, Prometheus, Grafana).

Подходами к решению аналитических задач, включая оптимизацию производительности и управление распределенными транзакциями в микросервисной архитектуре.

Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

Типовые задачи для самостоятельного решения:

1. Реализация микросервиса для управления пользователями с помощью Node.js и MongoDB. Необходимо разработать микросервис, который будет принимать запросы на создание, чтение, обновление и удаление пользователей из базы данных MongoDB. Реализовать систему аутентификации и авторизации пользователей с помощью JWT-токенов.

2. Разработка сервиса для обработки платежей с помощью Spring Boot и RabbitMQ. Необходимо создать микросервис, который будет принимать платежные запросы и отправлять их в очередь RabbitMQ для дальнейшей обработки. Реализовать

систему проверки платежеспособности клиентов и платежного шлюза для обработки платежей.

3. Создать микросервисы для онлайн-магазина: а именно создать микросервисы для управления корзиной покупателей, обработки заказов и возможностей доступа к каталогу товаров. Необходимо использовать различные инструменты микросервисной архитектуры, включая Docker, Kubernetes и API Gateway.

4. Разработка мониторинга микросервисных систем: создание системы мониторинга, которая будет производить сбор данных о работе каждого сервиса. Необходимо использовать такие инструменты, как Prometheus и Grafana, чтобы проанализировать и визуализировать данные для текущей работы и проактивного управления производительностью системы.

5. Разработка и тестирование микросервисов с использованием контейнеризации: разработать несколько микросервисов на языке программирования по своему выбору; упаковать каждый микросервис в контейнер на базе Docker.; выполнить интеграционное тестирование микросервисной архитектуры в среде Kubernetes; изучить принципы работы контейнеров и оркестратора, а также преимущества использования этой технологии при разработке микросервисов.

6. Применение принципов и практик DevOps в разработке микросервисов: определить процессы CI/CD и настроить их в соответствующих инструментах (например, GitLab, Jenkins, Ansible); создать несколько микросервисов и развернуть их на удаленном сервере при помощи инструментов автоматической доставки (например, Ansible, Terraform); разработать и применить стратегию обнаружения и устранения ошибок в микросервисной архитектуре; изучить основные принципы и практики DevOps и их применение при разработке и эксплуатации микросервисов.

7. Разработка системы мониторинга и анализа сообщений между микросервисами на основе Kafka и ELK-стека.

8. Реализация механизма маршрутизации запросов между микросервисами на основе Service Mesh (например, Istio) с использованием механизма транспортной авторизации и предотвращения утечек данных между сервисами.

9. "Автоматизация развертывания микросервисов на платформе Kubernetes". Необходимо изучить основы работы с кластером Kubernetes и создать микросервисы, которые будут автоматически разворачиваться и масштабироваться на этой платформе.

10. Управление конфигурацией микросервисов с помощью инструментов Ansible и Terraform: изучить принципы инфраструктуры кода; использовать инструменты Ansible и Terraform для автоматического развертывания и управления конфигурацией микросервисов в облаке.

11. Разработка комплексной системы мониторинга и логирования для микросервисов на основе ELK-стека.

12. Создание проактивной системы мониторинга и анализа логов для непрерывной оптимизации микросервисной архитектуры.

13. Тестирование микросервисов с использованием инструментов автоматизации тестирования: изучить различные инструменты автоматизации тестирования, такие как Selenium, JUnit, TestNG и т.д.; создать тестовый сценарий для микросервисов и протестировать его, используя один из этих инструментов.

14. Использование контейнерной виртуализации для тестирования микросервисов: изучить Docker, Kubernetes, Swarm и другие технологии контейнерной виртуализации; создать контейнер для микросервиса и запустить его в контейнерной среде; протестировать микросервис, используя созданный контейнер и убедиться, что микросервис работает корректно в контейнерной среде.

15. Разработка микросервисного приложения с использованием общего механизма управления состоянием (например, Redux) и архитектурного шаблона

проектирования (например, MVC): спроектировать и разработать микросервисные приложения, используя современные инструменты и подходы.

16. Реализация микросервисной архитектуры для медицинской системы управления данными: изучить методы и способы реализации микросервисной архитектуры для медицинской системы управления данными; использовать технологии, такие как Docker, Kubernetes и Istio, для создания микросервисов, которые позволят создать гибкую и масштабируемую инфраструктуру для управления большими объемами медицинских данных.

Зачетно-экзаменационные материалы для промежуточной аттестации (экзамен/зачет)

Вопросы для подготовки к зачету

1. Что такое микросервисная архитектура и как она отличается от монолитной?
2. Какие основные принципы лежат в основе микросервисной архитектуры?
3. Какие преимущества имеет микросервисная архитектура?
4. Какие недостатки можно выделить у микросервисной архитектуры?
5. Какие технологии используются в микросервисной архитектуре?
6. Какую роль играет контейнеризация в микросервисной архитектуре?
7. Что такое API Gateway и какова его роль в микросервисной архитектуре?
8. Как происходит мониторинг и логирование в микросервисной архитектуре?
9. Как обеспечивается безопасность в микросервисной архитектуре?
10. Какие подходы используются для организации коммуникации между микросервисами?
11. Как осуществляется управление транзакциями в микросервисной архитектуре?
12. Какую роль играют серверные события (Server-Sent Events) в микросервисной архитектуре?
13. Какую роль играют команды DevOps в микросервисной архитектуре?
14. Как решаются проблемы масштабирования в микросервисной архитектуре?
15. Что такое микросервисный шаблон Circuit Breaker и какова его роль в микросервисной архитектуре?
16. Как решаются проблемы взаимодействия микросервисов в условиях ненадежной сети?
17. Какую роль играет Service Registry в микросервисной архитектуре?
18. Какую роль играют команды Agile в микросервисной архитектуре?
19. Как обеспечивается согласованность данных в микросервисной архитектуре?
20. Что такое тестирование среды (Chaos Engineering) и как оно связано с микросервисной архитектурой?
21. Какие ограничения на количество микросервисов существуют в микросервисной архитектуре?
22. Как можно организовать автоматическое масштабирование в микросервисной архитектуре?
23. Какие подходы используются для версионирования микросервисов?
24. Как проводится мониторинг производительности микросервисов?
25. Как осуществляется управление конфигурацией микросервисов?
26. Какие подходы используются для управления персистентными данными в микросервисной архитектуре?
27. Какие принципы должны учитываться при проектировании микросервисной архитектуры?
28. Какие подходы используются для организации маршрутизации HTTP-запросов в микросервисной архитектуре?

29. Какие возможности предоставляют современные платформы для реализации микросервисной архитектуры?

30. Как осуществляется обеспечение высокой доступности в микросервисной архитектуре?

31. Какие подходы используются для управления состоянием микросервисов?

32. Какие роли выполняют микросервисы в современных приложениях и сервисах?

4.2 Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Методические рекомендации, определяющие процедуры оценивания тестов:

Тест проводится онлайн в системе Moodle или Google Docs и ограничен по времени. На сдачу теста дается две попытки. Тест считается успешно пройденным если студент правильно ответил на 70% вопросов.

Методические рекомендации, определяющие процедуры оценивания выполнения контрольных заданий:

Задание считается выполненным при выполнении следующих условий:

- предоставлен исходный код на Python в среде PyCharm, Google Collab
- продемонстрирована работоспособность программы
- студент понимает исходный код и отвечает на вопросы по его организации.

Методические рекомендации к сдаче зачета

Процедура промежуточной аттестации проходит в соответствии с Положением о текущем контроле и промежуточной аттестации обучающихся ФГБОУ ВО «КубГУ».

Текущий контроль успеваемости студентов проводится в целях совершенствования и непрерывного контроля качества образовательного процесса, проверки усвоения учебного материала, активизации самостоятельной работы студентов, стимулирования их учебной работы, обеспечения эффективности образовательного процесса, предупреждения рисков отчисления студентов.

Текущий контроль знаний студентов осуществляется постоянно в течение всего семестра.

Виды текущего контроля: устный (письменный) опрос на занятиях; проверка выполнения домашних заданий; проведение контрольных работ; оценка активности студента на занятии.

Студенты очной формы обучения обязаны сдать зачет до начала экзаменационной сессии.

По результатам сдачи зачета выставляется оценка «зачтено»/ «не зачтено».

Выставление зачетов для студентов очной формы обучения проводятся в период до экзаменационной сессии.

При отсутствии зачетной книжки у студента экзаменатор не имеет права принимать у него зачет/экзамен. Такой студент считается не явившимся на зачет/экзамен. В исключительных случаях, на основании распоряжения декана преподаватель может допустить студента к зачету/экзамену при наличии документа, удостоверяющего личность.

В целях объективного оценивания знаний во время проведения зачетов и экзаменов не допускается наличие у студентов посторонних предметов и технических устройств.

Студенту, использующему в ходе зачета неразрешенные источники и средства получения информации, выставляется неудовлетворительная оценка, и он удаляется из аудитории.

Во время зачета студенты могут пользоваться утвержденной рабочей программой учебной дисциплины, которая должна быть в наличии на экзамене, а также с разрешения экзаменатора справочной литературой и другими пособиями.

Студенты, нарушающие правила поведения при проведении зачетов и экзаменов, могут быть незамедлительно удалены из аудитории, к ним могут быть применены меры дисциплинарного воздействия.

На зачете/экзамене могут присутствовать ректор, проректор по учебной работе, декан факультета, заведующий кафедрой, которая обеспечивает учебный процесс по данной дисциплине. Присутствие на экзаменах и зачетах посторонних лиц без разрешения ректора или проректора по учебной работе не допускается.

После прослушивания лекции рекомендуется выполнить упражнения, приводимые в лекции для самостоятельной работы, а также выполнить на компьютере с использованием среды Python задачи, приводимые в лекции в качестве примеров.

При самостоятельной работе студентов необходимо изучить литературу, приведенную в перечнях, для осмысления вводимых понятий, анализа предложенных подходов и методов разработки параллельных программ. Разрабатывая решение новой задачи, студент должен уметь выбрать методы решения задачи с учетом целевой аппаратной платформы, проводить отладку и профилирование программы на языке Python.

В качестве систем программирования для решения задач и изучения методов и алгоритмов, приведенных в лекциях, рекомендуется использовать на практических занятиях и при самостоятельной работе стандартную реализацию языка Python в связке со средой разработки PyCharm, Google Collab.

Оценка	
Не зачтено	Зачтено
• если студент правильно решил менее 70 % задач и/или не имеет представление как решать остальные задачи • не знает значительной части теоретического материала, допускает существенные ошибки.	• если студент правильно решил 70 % задач, имеет представление как решать остальные задачи • на теоретические вопросы дан развернутый ответ. Материал изложен в целом последовательно. Имеются логичные и аргументированные выводы

Оценочные средства для инвалидов и лиц с ограниченными возможностями здоровья выбираются с учетом их индивидуальных психофизических особенностей.

– при необходимости инвалидам и лицам с ограниченными возможностями здоровья предоставляется дополнительное время для подготовки ответа на экзамене;

– при проведении процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья предусматривается использование технических средств, необходимых им в связи с их индивидуальными особенностями;

– при необходимости для обучающихся с ограниченными возможностями здоровья и инвалидов процедура оценивания результатов обучения по дисциплине может проводиться в несколько этапов.

Процедура оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья по дисциплине предусматривает предоставление информации в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа.

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

5. Перечень учебной литературы, информационных ресурсов и технологий

5.1 Основная литература:

1. Лаврищева, Е. М. Программная инженерия и технологии программирования сложных систем : учебник для вузов / Е. М. Лаврищева. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 432 с. — (Высшее образование). — ISBN 978-5-534-07604-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/513067> (дата обращения: 08.06.2025).

2. Чернышев, С. А. Принципы, паттерны и методологии разработки программного обеспечения : учебное пособие для вузов / С. А. Чернышев. — Москва : Издательство Юрайт, 2023. — 176 с. — (Высшее образование). — ISBN 978-5-534-14383-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/520097> (дата обращения: 08.06.2025).

3. «Архитектурные решения информационных систем : учебник для вузов / А. И. Водяхо, Л. С. Выговский, В. А. Дубенецкий, В. В. Цехановский. — 3-е изд., стер. — Санкт-Петербург : Лань, 2022. — 356 с. — ISBN 978-5-507-44710-7. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/254624> (дата обращения: 08.06.2025). — Режим доступа: для авториз. пользователей.»

Для освоения дисциплины инвалидами и лицами с ограниченными возможностями здоровья имеются издания в электронном виде в электронно-библиотечных системах *«Лань»* и *«Юрайт»*.

5.2 Дополнительная литература:

1. Кафка Д., Мурти Д. Нигам С. Микросервисная архитектура. Реализация и проектирование: Пер. с англ. — М.: ДМК-Пресс, 2017. — 432 с.

2. Фаулер М., Палмер Р., Льюис Джеймс. Микросервисы. Паттерны разработки и лучшие практики: Пер. с англ. — М.: ДМК-Пресс, 2018. — 510 с.
3. МакКроски Р. Микросервисы. Создание расширяемых и отказоустойчивых приложений: Пер. с англ. — М.: ДМК-Пресс, 2018. — 460 с.
4. Саммервиль Дж., Жерарди Р. Микросервисы в действии: Пер. с англ. — СПб.: Питер, 2018. — 304 с.
5. Ганри Ш. Микросервисы. Надежные и масштабируемые системы: Пер. с англ. — М.: ООО «ДМК Пресс», 2019. — 340 с.
6. Баджуска Д., Москетто Р. Микросервисы изнутри. Основы, инструменты и практические рекомендации: Пер. с англ. — М.: ДМК-Пресс, 2020. — 480 с.2.

5.3. Периодические издания:

1. Базы данных компании «Ист Вью» <http://dlib.eastview.com>
2. Электронная библиотека GREBENNICON.RU <https://grebennikon.ru/>

5.4. Интернет-ресурсы, в том числе современные профессиональные базы данных и информационные справочные системы

Электронно-библиотечные системы (ЭБС):

1. ЭБС «ЮРАЙТ» <https://urait.ru/>
2. ЭБС «УНИВЕРСИТЕТСКАЯ БИБЛИОТЕКА ОНЛАЙН» <http://www.biblioclub.ru/>
3. ЭБС «BOOK.ru» <https://www.book.ru>
4. ЭБС «ZNANIUM.COM» www.znanium.com
5. ЭБС «ЛАНЬ» <https://e.lanbook.com>

Профессиональные базы данных

1. Scopus <http://www.scopus.com/>
2. ScienceDirect <https://www.sciencedirect.com/>
3. Журналы издательства Wiley <https://onlinelibrary.wiley.com/>
4. Научная электронная библиотека (НЭБ) <http://www.elibrary.ru/>
5. Полнотекстовые архивы ведущих западных научных журналов на Российской платформе научных журналов НЭИКОН <http://archive.neicon.ru>
6. Национальная электронная библиотека (доступ к Электронной библиотеке диссертаций Российской государственной библиотеки (РГБ) <https://rusneb.ru/>
7. Президентская библиотека им. Б.Н. Ельцина <https://www.prilib.ru/>
8. База данных CSD Кембриджского центра кристаллографических данных (CCDC) <https://www.ccdc.cam.ac.uk/structures/>
9. Springer Journals: <https://link.springer.com/>
10. Springer Journals Archive: <https://link.springer.com/>
11. Nature Journals: <https://www.nature.com/>
12. Springer Nature **Protocols and Methods:** <https://experiments.springernature.com/sources/springer-protocols>
13. Springer Materials: <http://materials.springer.com/>
14. Nano Database: <https://nano.nature.com/>
15. Springer eBooks (i.e. 2020 eBook collections): <https://link.springer.com/>
16. "Лекториум ТВ" <http://www.lektorium.tv/>
17. Университетская информационная система РОССИЯ <http://uisrussia.msu.ru>

Информационные справочные системы

1. Консультант Плюс - справочная правовая система (доступ по локальной сети с компьютеров библиотеки)

Ресурсы свободного доступа

1. КиберЛенинка <http://cyberleninka.ru/>;
2. Американская патентная база данных <http://www.uspto.gov/patft/>
3. Министерство науки и высшего образования Российской Федерации <https://www.minobrnauki.gov.ru/>;
4. Федеральный портал "Российское образование" <http://www.edu.ru/>;
5. Информационная система "Единое окно доступа к образовательным ресурсам" <http://window.edu.ru/>;
6. Единая коллекция цифровых образовательных ресурсов <http://school-collection.edu.ru/> .
7. Проект Государственного института русского языка имени А.С. Пушкина "Образование на русском" <https://pushkininstitute.ru/>;
8. Справочно-информационный портал "Русский язык" <http://gramota.ru/>;
9. Служба тематических толковых словарей <http://www.glossary.ru/>;
10. Словари и энциклопедии <http://dic.academic.ru/>;
11. Образовательный портал "Учеба" <http://www.ucheba.com/>;
12. Законопроект "Об образовании в Российской Федерации". Вопросы и ответы http://xn--273--84d1f.xn--p1ai/voprosy_i_otvety

Собственные электронные образовательные и информационные ресурсы КубГУ

1. Электронный каталог Научной библиотеки КубГУ <http://megapro.kubsu.ru/MegaPro/Web>
2. Электронная библиотека трудов ученых КубГУ <http://megapro.kubsu.ru/MegaPro/UserEntry?Action=ToDb&idb=6>
3. Среда модульного динамического обучения <http://moodle.kubsu.ru>
4. База учебных планов, учебно-методических комплексов, публикаций и конференций <http://infoneeds.kubsu.ru/>
5. Библиотека информационных ресурсов кафедры информационных образовательных технологий <http://mschool.kubsu.ru;>
6. Электронный архив документов КубГУ <http://docspace.kubsu.ru/>
7. Электронные образовательные ресурсы кафедры информационных систем и технологий в образовании КубГУ и научно-методического журнала "ШКОЛЬНЫЕ ГОДЫ" <http://icdau.kubsu.ru/>

5.5 Перечень информационно-коммуникационных технологий

- Проверка домашних заданий и консультирование посредством электронной почты.
- Использование электронных презентаций при проведении лекционных занятий
- Система MOODLE
- Проверка домашних заданий и консультирование посредством ЭОИС КубГУ

5.6 Перечень лицензионного и свободно распространяемого программного обеспечения

OpenOffice
 Компилятор C++
 Oracle VirtualBox 6
 VMware Workstation 16
 Putty 0.76 или Kitty 0.76
 FileZilla 3.57.0
 WinSCP 5.19
 Advanced port scanner 2.5
 Python 3 (3.7 И 3.9)
 numpy 1.22.0
 opencv 4.5.5

Keras 2.7.0
 Tensor flow 2.7.0
 matplotlib 3.5.1
 PyCharm 2021
 Cuda Toolkit 11.6
 Фреймворк Django
 Firefox, любая версия
 Putty, любая версия
 Visual Studio Code, версия 1.52+
 Eclipse PHP Development Tools, версия 2020-06+
 Плагин Remote System Explorer (RSE) для Eclipse PDT
 JetBrains PHP Storm
 GIT
 Java Version 8 Update 311
 Clojure 1.10.3.1029.ps1
 SWI Prolog 8.4
 IntelliJ Idea IDE 2021
 Mozilla Firefox 96
 Google Chrome 97
 GitHub Desktop 2.9
 PHP Storm 2021
 FileZilla 3.57.0
 Putty 0.76

6. Методические указания для обучающихся по освоению дисциплины (модуля)

В освоении дисциплины инвалидами и лицами с ограниченными возможностями здоровья большое значение имеет индивидуальная учебная работа (консультации) – дополнительное разъяснение учебного материала.

Индивидуальные консультации по предмету являются важным фактором, способствующим индивидуализации обучения и установлению воспитательного контакта между преподавателем и обучающимся инвалидом или лицом с ограниченными возможностями здоровья.

7. Материально-техническое обеспечение по дисциплине (модулю)

№	Вид работ	Наименование учебной аудитории, ее оснащенность оборудованием и техническими средствами обучения
1.	Лекционные занятия	Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения
2.	Лабораторные занятия	Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения, компьютерами, проектором, программным обеспечением
3.	Групповые (индивидуальные) консультации	Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения, компьютерами, программным обеспечением
4.	Текущий контроль, промежуточная аттестация	Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения, компьютерами, программным обеспечением

5.	Самостоятельная работа	Кабинет для самостоятельной работы, оснащенный компьютерной техникой с возможностью подключения к сети «Интернет», программой экранного увеличения и обеспеченный доступом в электронную информационно-образовательную среду университета.
----	------------------------	--

Примечание: Конкретизация аудиторий и их оснащение определяется ОПОП.

нащение определяется ОПОП.