

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
« 29 » августа 2025 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)
Б1. О.30 Микросервисная архитектура

Направление подготовки 02.03.01 Математика и компьютерные науки

Профиль Искусственный интеллект и аналитика данных

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 02.03.01 Математика и компьютерные науки

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 № 1 от «26» августа 2025г.

Заведующий кафедрой Подколзин В.В.



Утверждена на заседании учебно-методической комиссии факультета компьютерных технологий и прикладной математики протокол № 01 «28» августа 2025 г.

Председатель УМК факультета Коваленко А.В.



подпись

Рецензенты:

Мостовой Евгений Викторович, генеральный директор ООО «Портал-Юг»,
e-mail: mostovoy@portal-yug.ru

Луценко Евгений Вениаминович, доктор экономических наук, кандидат технических наук, профессор кафедры компьютерных технологий и систем Федерального государственного бюджетное образовательное учреждение высшего образования «Кубанский государственный аграрный университет имени И.Т. Трубилина», e-mail: prof.lutsenko@gmail.com

1 Цели и задачи изучения дисциплины (модуля)

1.1 Цель освоения дисциплины

Цель дисциплины – сформировать у студентов компетенции в области проектирования и реализации микросервисных архитектур с акцентом на применение в системах искусственного интеллекта и аналитики данных. В ходе освоения дисциплины студенты изучат ключевые архитектурные стили и паттерны микросервисов, научатся проектировать и развертывать распределенные системы обработки данных, а также применять полученные знания для построения масштабируемых ML-решений. Это позволит оптимизировать программную архитектуру data-driven приложений, эффективно организовывать ML-пайплайны и управлять жизненным циклом моделей в production-средах.

1.2 Задачи дисциплины

В рамках изучения дисциплины решаются следующие задачи:

- Изучение принципов микросервисной архитектуры и ее преимуществ для систем искусственного интеллекта и аналитики данных
- Приобретение практического опыта проектирования и разработки специализированных микросервисов
- Освоение инструментов контейнеризации и оркестрации для развертывания распределенных аналитических pipeline
- Формирование навыков командной работы в кросс-функциональных командах
- Анализ проблем масштабирования и управления микросервисными системами в контексте обработки больших объемов данных
- Изучение методов мониторинга и отладки data-intensive приложений с учетом специфики ML-моделей
- Применение микросервисного подхода для построения гибких и масштабируемых AI-решений
- Освоение практик обеспечения надежности и безопасности в распределенных системах обработки данных
- Изучение подходов к интеграции микросервисов с современными платформами для машинного обучения и аналитики

1.3 Место дисциплины (модуля) в структуре образовательной программы

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

1.4 Профессиональные роли в структуре образовательной программы

Роль 1: Data Analyst (Аналитик данных)

Извлечение знаний из данных, построение аналитических моделей, использующих МО и ИИ.

Задачи:

- 1. Статистический анализ, визуализация данных, предварительная обработка.*
- 2. Создание прогнозных моделей*
- 3. Построение аналитических моделей для поддержки бизнес-решений.*

Роль 2: MLOps (Специалист по эксплуатации ИИ)

Автоматизация и операционное управление жизненным циклом МО-моделей

Задачи:

- 1. DevOps для ML.*
- 2. Автоматизация, мониторинг ML-систем.*
- 3. Операционное управление жизненным циклом ML-моделей.*

Роль 3: AI PM (Менеджер проектов ИИ)

Управление процессами создания ИИ-решений, включая координацию команды разработки

Задачи:

1. Управление ИИ-проектами от идеи до внедрения
2. Анализ бизнес-требований и постановка задач
3. Оценка эффективности и ROI ИИ-решений

1.5 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-5 Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности

ОПК-5.1 *Применяет современные языки программирования и технологии для решения математических и вычислительных задач*

Знать *Современные языки программирования и фреймворки для разработки микросервисов (Java/Spring Boot, Python/FastAPI, Node.js/NestJS)
Технологии контейнеризации и оркестрации (Docker, Kubernetes)
Принципы проектирования и оптимизации распределенных вычислений*

Уметь *Выбирать и применять подходящие языки программирования и технологии для реализации микросервисов
Разрабатывать эффективные алгоритмы для распределенных систем
Оптимизировать вычислительные процессы в микросервисной архитектуре*

Владеть *Навыками разработки на современных языках программирования
Технологиями контейнеризации и развертывания приложений
Методами профилирования и оптимизации производительности систем*

SS-2 Способен осуществлять свою трудовую деятельность с учётом необходимости эффективной коммуникации и взаимодействия в рамках коллективной проектной работы в сфере ИИ

SS-2.1 *Эффективно коммуницирует с участниками проектной команды при планировании, реализации и анализе результатов работы.*

Знать *Методологии управления проектами (Agile, Scrum, Kanban)
Принципы распределения ролей в команде (DevOps, Backend, Data Engineer)
Инструменты командной разработки (Git, GitFlow, Merge Requests)*

Уметь *Организовывать командную работу с использованием issue trackers (Jira, YouTrack)*

Владеть *Формулировать технические требования и декомпозировать задачи
Координировать работу с другими разработчиками и аналитиками
Навыками ведения технических обсуждений и документации (Confluence, Notion)*

*Методами проведения code review и конструктивной обратной связи
Навыками презентации результатов (демо-сессии, отчеты)*

SS-2.2 *Учитывает профессиональные и ролевые особенности коллег при совместной разработке технических решений и представлении результатов*

Знать *Особенности работы специалистов по ИИ и аналитике данных в MSA
Принципы кросс-командного взаимодействия в распределенных системах*

	<i>Основы технического письма и визуализации архитектурных решений (UML, С4-модель)</i>
Уметь	<i>Адаптировать техническую документацию под разные роли (разработчик, аналитик) Находить компромиссы при выборе технологий и архитектурных решений Представлять сложные концепции в доступной форме</i>
Владеть	<i>Навыками кросс-функционального взаимодействия Методами разрешения конфликтов в команде Навыками публичных выступлений и защиты проектов</i>
LLM-4	Проектирует, разрабатывает и интегрирует интеллектуальных агентов на базе генеративных моделей
LLM-4.1	Умеет применять и разрабатывать интеллектуальных агентов
Знать	<i>Архитектурные паттерны и подходы к проектированию интеллектуальных агентов (ReAct, Plan-and-Execute, Multi-Agent Systems) Принципы интеграции LLM с внешними инструментами и API (Tool Calling, RAG) Методы оценки эффективности и безопасности агентов</i>
Уметь	<i>Проектировать архитектуру агента для решения конкретной бизнес-задачи Интегрировать LLM с системами и базами знаний предприятия Тестировать и оценивать качество работы агента</i>
Владеть	<i>Навыками работы с фреймворками для создания агентов (LangChain, llamaIndex) Методами оптимизации взаимодействия агента с пользователем и внешними системами Инструментами мониторинга и логирования работы агентов в production-среде</i>

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	СРС
1	2	3	4	5	6	7
1.	Введение в микросервисную архитектуру и её роль в современных ИТ-системах	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 и её роль в современных ИТ-системах	Основные концепции, сравнение с монолитной архитектурой, преимущества и ограничения. Примеры использования в сфере ИИ и аналитики данных.	К
2.	Принципы построения микросервисов	Ключевые принципы (единство ответственности, независимость), популярные паттерны (CQRS, Saga), рекомендации по декомпозиции.	К, Т
3.	Коммуникация между микросервисами	Синхронные (REST, gRPC) и асинхронные (Kafka, RabbitMQ) методы взаимодействия. Event-driven архитектура.	К
4.	Управление конфигурацией и развертыванием микросервисов	Контейнеризация (Docker), оркестрация (Kubernetes), CI/CD-практики. Особенности работы в облачных средах.	К, Т

№	Наименование раздела (темы)	Содержание раздела (темы)	Форма текущего контроля
1	2	3	4
5.	Мониторинг и логирование в микросервисной архитектуре	Централизованное логирование (ELK), метрики (Prometheus), трейсинг (OpenTelemetry). Анализ производительности.	К
6.	Тестирование микросервисов	Стратегии тестирования (unit, интеграционные, e2e). Паттерны устойчивости (Circuit Breaker, Retry). Chaos Engineering.	К, Т
7.	Проблемы и решения в микросервисной архитектуре	Аутентификация (OAuth2), авторизация, защита данных. Оптимизация производительности и масштабируемости.	К
8.	Реализация микросервисной архитектуры в конкретных проектах	Разбор кейсов из реальных проектов. Особенности командной работы. Инструменты коллективной разработки. 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 и её роль в современных ИТ-системах	Создание базового микросервиса с REST API (Spring Boot/Flask + OpenAPI)	РЗ
2.	Введение в микросервисную архитектуру и её роль в современных ИТ-системах	Контейнеризация ML-модели как микросервиса (Docker + Docker Compose)	РЗ
3.	Принципы построения микросервисов	Реализация микросервиса с соблюдением SRP	РЗ
4.	Принципы построения микросервисов	Внедрение зависимостей через DI-контейнер	РЗ
5.	Коммуникация между микросервисами	Настройка синхронного взаимодействия через gRPC.	РЗ
6.	Коммуникация между микросервисами	Реализация event-driven архитектуры с Apache Kafka	РЗ
7.	Управление конфигурацией и развертыванием микросервисов	Оркестрация микросервисов в Kubernetes (Helm charts)	РЗ
8.	Управление конфигурацией и развертыванием микросервисов	Настройка CI/CD пайплайна (GitLab CI/GitHub Actions)	РЗ
9.	Мониторинг и логирование в микросервисной архитектуре	Настройка мониторинга (Prometheus + Grafana)	РЗ
10.	Мониторинг и логирование в микросервисной архитектуре	Логирование качества предсказаний (MLflow Tracking + Elasticsearch)	РЗ
11.	Тестирование микросервисов	Интеграционное тестирование с TestContainers	РЗ

№	Наименование раздела (темы)	Наименование лабораторных работ	Форма текущего контроля
1	2	3	4
12.	Тестирование микросервисов	Нагрузочное тестирование микросервисов (Locust)	РЗ
13.	Проблемы и решения в микросервисной архитектуре	Реализация JWT-аутентификации	РЗ
14.	Проблемы и решения в микросервисной архитектуре	Настройка горизонтального масштабирования	РЗ
15.	Реализация микросервисной архитектуры в конкретных проектах	Разработка системы рекомендаций на микросервисах	РЗ
16.	Реализация микросервисной архитектуры в конкретных проектах	Оптимизация ETL-процессов в микросервисной архитектуре	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 вышеназванных 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 компетенции (или ее части)	Наименование оценочного средства	
			Текущий контроль	Промежуточная аттестация
1	Введение в микросервисную архитектуру и её роль в современных ИТ-системах	ОПК-5.1 SS-2.1 SS-2.2 LLM-4.1	Задания для самостоятельного решения 1-2	Теоретический вопрос 1-4
2	Принципы построения микросервисов	ОПК-5.1 SS-2.1 SS-2.2 LLM-4.1	Задания для самостоятельного решения 3-4	Теоретический вопрос 5-8
3	Коммуникация между микросервисами	ОПК-5.1 SS-2.1 SS-2.2 LLM-4.1	Задания для самостоятельного решения 5-6	Теоретический вопрос 9-12
4	Управление конфигурацией и развертыванием микросервисов	ОПК-5.1 SS-2.1 SS-2.2	Задания для самостоятельного решения 7-8	Теоретический вопрос 13-16

		LLM-4.1		
5	Мониторинг и логирование в микросервисной архитектуре	ОПК-5.1 SS-2.1 SS-2.2 LLM-4.1	Задания для самостоятельного решения 9-10	Теоретический вопрос 17-20
6	Тестирование микросервисов	ОПК-5.1 SS-2.1 SS-2.2 LLM-4.1	Задания для самостоятельного решения 11-12	Теоретический вопрос 21-24
7	Проблемы и решения в микросервисной архитектуре	ОПК-5.1 SS-2.1 SS-2.2 LLM-4.1	Задания для самостоятельного решения 13-14	Теоретический вопрос 25-28
8	Реализация микросервисной архитектуры в конкретных проектах	ОПК-5.1 SS-2.1 SS-2.2 LLM-4.1	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-5 **Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности**

ОПК-5.1 **Применяет современные языки программирования и технологии для решения математических и вычислительных задач**

Знать Основные языки программирования для разработки микросервисов
Базовые принципы контейнеризации приложений
Основные подходы к оптимизации вычислений

Уметь Использовать основные языки программирования для реализации сервисов
Применять технологии контейнеризации для развертывания приложений
Проводить базовую оптимизацию вычислительных процессов

Владеть Навыками программирования на одном из современных языков
Технологиями создания и запуска контейнеров
Методами оценки производительности приложений

SS-2 **Способен осуществлять свою трудовую деятельность с учётом необходимости эффективной коммуникации и взаимодействия в рамках коллективной проектной работы в сфере ИИ**

SS-2.1 **Эффективно коммуницирует с участниками проектной команды при планировании, реализации и анализе результатов работы.**

Знать Особенности Agile-методологий в data science проектах
Ролевую модель в командах разработки AI-решений
Инструменты совместной работы над ML-проектами

Уметь Координировать работу data scientists и разработчиков микросервисов
Формулировать требования к ML-моделям как сервисам
Организовывать кросс-функциональное взаимодействие

Владеть	<i>Методами документирования ML-экспериментов Практиками проведения code review для аналитического кода Навыками презентации результатов data science проектов</i>
SS-2.2	Учитывает профессиональные и ролевые особенности коллег при совместной разработке технических решений и представлении результатов
Знать	<i>Специфику работы с данными в распределенных системах Принципы визуализации архитектур data-intensive приложений Особенности коммуникации между техническими специалистами и аналитиками</i>
Уметь	<i>Адаптировать технические решения для задач анализа данных Находить баланс между исследовательскими и production-требованиями Объяснять сложные концепции машинного обучения нетехническим специалистам</i>
Владеть	<i>Методами визуализации data pipeline Навыками кросс-дисциплинарного взаимодействия Техниками презентации аналитических решений</i>
LLM-4	Проектирует, разрабатывает и интегрирует интеллектуальных агентов на базе генеративных моделей
LLM-4.1	Умеет применять и разрабатывать интеллектуальных агентов
Знать	<i>Базовые архитектурные паттерны построения интеллектуальных агентов (ReAct, Plan-and-Execute) Способы интеграции LLM с внешними инструментами и API Критерии оценки работоспособности агента</i>
Уметь	<i>Разрабатывать простых интеллектуальных агентов для типовых задач Интегрировать агента с одним внешним инструментом или API Проводить базовое тестирование функциональности агента</i>
Владеть	<i>Навыками работы с одним из фреймворков для создания агентов (LangChain) Методами подключения агента к источнику данных Инструментами логирования ключевых этапов работы агента</i>

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. Разработка сервиса управления пользовательскими данными с реализацией механизмов аутентификации на основе токенов и разграничением прав доступа в соответствии с ролевой моделью
2. Создание распределённой системы обработки финансовых транзакций с обеспечением надежности доставки сообщений и механизмами восстановления после сбоев
3. Проектирование и реализация модульной архитектуры электронной торговой платформы с выделением сервисов каталога, корзины и обработки заказов

4. Разработка системы мониторинга производительности распределённых приложений с возможностью сбора, агрегации и визуализации ключевых метрик работоспособности
5. Исследование методов контейнеризации программных компонентов и их оркестрации в кластерной среде с оценкой эффективности различных стратегий развертывания
6. Автоматизация процессов непрерывной интеграции и поставки программного обеспечения с реализацией pipeline для тестирования, сборки и развертывания
7. Создание системы анализа потоковых данных между компонентами распределённой архитектуры с возможностью выявления аномалий взаимодействия
8. Реализация механизмов безопасного взаимодействия сервисов с контролем доступа на уровне сетевых вызовов и валидацией передаваемых данных
9. Изучение методов автоматизированного развёртывания и горизонтального масштабирования приложений в облачной среде с балансировкой нагрузки
10. Разработка системы управления конфигурациями в распределённых средах с поддержкой различных окружений и механизмом централизованного обновления параметров
11. Проектирование комплексного решения для сбора, хранения и анализа операционных данных с возможностью генерации аналитических отчетов
12. Создание системы проактивного мониторинга и оптимизации производительности с механизмами прогнозирования критических состояний
13. Исследование методов автоматизированного тестирования распределённых систем с оценкой покрытия функциональных сценариев и нагрузочных характеристик
14. Применение технологий виртуализации для обеспечения изолированного тестирования компонентов с анализом воспроизводимости результатов в различных средах
15. Разработка специализированного сервиса обработки структурированных данных с реализацией pipeline для преобразования признаков и интеграцией алгоритмов машинного обучения
16. Проектирование безопасной архитектуры для системы управления медицинскими данными с учетом требований к конфиденциальности и реализацией механизмов аудита доступа

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

1. Дайте определение микросервисной архитектуры и проведите сравнительный анализ с монолитным подходом к разработке программного обеспечения
2. Опишите ключевые принципы проектирования систем на основе микросервисной архитектуры
3. Проанализируйте преимущества микросервисного подхода при построении сложных распределенных систем
4. Обсудите основные проблемы и ограничения, связанные с реализацией микросервисной архитектуры
5. Классифицируйте и охарактеризуйте основные технологии, используемые при реализации микросервисных решений
6. Объясните значение технологии контейнеризации в контексте микросервисной архитектуры
7. Опишите назначение и архитектурную роль API Gateway в системах, построенных на микросервисах

8. Представьте методы и инструменты мониторинга и логирования в распределенных микросервисных системах
9. Проанализируйте подходы к обеспечению информационной безопасности в микросервисных архитектурах
10. Сравните различные методы организации межсервисного взаимодействия в распределенных системах
11. Опишите стратегии управления распределенными транзакциями в условиях микросервисной архитектуры
12. Объясните роль асинхронной коммуникации на основе событий в микросервисных системах
13. Проанализируйте значение методологии DevOps при разработке и эксплуатации микросервисов
14. Представьте методы горизонтального и вертикального масштабирования микросервисных систем
15. Опишите паттерн Circuit Breaker и его значение для обеспечения отказоустойчивости
16. Проанализируйте подходы к обеспечению надежности взаимодействия сервисов в условиях нестабильной сети
17. Объясните назначение и принципы работы Service Discovery в микросервисных архитектурах
18. Обсудите преимущества Agile-методологий при разработке микросервисных систем
19. Представьте методы обеспечения согласованности данных в распределенных микросервисных системах
20. Опишите принципы Chaos Engineering и их применение для тестирования отказоустойчивости
21. Проанализируйте критерии определения оптимального уровня гранулярности микросервисов
22. Объясните принципы автоматического масштабирования в облачных микросервисных решениях
23. Сравните различные стратегии управления версиями API в микросервисных системах
24. Опишите методы и метрики для оценки производительности микросервисных решений
25. Представьте подходы к централизованному управлению конфигурациями распределенных сервисов
26. Проанализируйте стратегии работы с персистентными данными в микросервисной архитектуре
27. Сформулируйте основные принципы декомпозиции предметной области при проектировании микросервисов
28. Опишите механизмы маршрутизации запросов в сложных микросервисных экосистемах
29. Сравните возможности современных платформ для развертывания микросервисных решений
30. Представьте архитектурные подходы к обеспечению высокой доступности сервисов
31. Проанализируйте методы управления состоянием в stateless и stateful микросервисах
32. Классифицируйте типовые роли и функциональные обязанности микросервисов в современных ИТ-системах

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

Методические рекомендации, определяющие процедуры оценивания выполнения контрольных заданий:

Задание считается выполненным при выполнении следующих условий:

- предоставлен исходный код
- 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 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.

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 различных языков и сред программирования.

Оценка	
Не зачтено	Зачтено
<ul style="list-style-type: none">• 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.	<ul style="list-style-type: none">• 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.

4.3. Методические указания по организации вычислительной инфраструктуры

Условия применения:

- Курс рассчитан на студентов 3 курса (6 семестр).
- Наличие доступа к вычислительным ресурсам (Docker, Kubernetes, GitLab CI/GitHub Actions, облачные платформы Yandex Cloud/SberCloud).
- Использование технологий: Spring Boot/FastAPI, Kafka, Redis, Prometheus, Grafana.

Цели, задачи и ожидаемые результаты:

Цели:

- Обеспечить студентов инструментами для развертывания и оркестрации микросервисов.
- Научить работать с системами мониторинга, логирования и CI/CD.
- Сформировать навыки командной работы с использованием Git и issue trackers.

Задачи преподавателя:

- Разработка шаблонного репозитория с предустановленной структурой микросервисного приложения.
- Настройка пайплайнов CI/CD для автоматического тестирования, сборки и развертывания.
- Подготовка инструкций по работе с Docker, Kubernetes и облачными платформами.

Ожидаемые результаты студентов:

- Умение работать с Docker и Kubernetes (развертывание, масштабирование, обновление).
- Навыки настройки CI/CD для микросервисов.
- Понимание принципов мониторинга и отладки распределенных систем.

Порядок реализации:

Настройка CI/CD:

- Для GitLab: настройка GitLab Runner, создание .gitlab-ci.yml для сборки образов и деплоя в Kubernetes.
- Для GitHub: использование GitHub Actions для автоматизации тестирования и развертывания.
- Интеграция с container registry (Docker Hub, GitLab Registry).

Шаблонный репозиторий:

Включает:

- gitignore для Java/Python.

- Dockerfile и docker-compose.yml для каждого микросервиса.
- Манифесты Kubernetes (deployment, service, ingress).
- Папку monitoring с конфигурами Prometheus и Grafana.
- README.md с инструкциями по запуску и тестированию.

Автотесты:

- Пример: тестирование API эндпоинтов, проверка здоровья сервисов, нагрузочное тестирование.
- Генерация отчетов о покрытии тестами и качестве кода (SonarQube).

Порядок проверки корректности:

Чек-лист:

- Наличие Git-репозитория у всех студентов/команд.
- Успешное прохождение CI/CD пайплайна.
- Корректная работа развернутого приложения в Kubernetes.
- Наличие настроенного мониторинга и логирования.

4.4. Методические указания по организации лабораторных работ

Условия применения:

- Наличие Git-репозитория у всех студентов/команд.
- Успешное прохождение CI/CD пайплайна.
- Корректная работа развернутого приложения в Kubernetes.
- Наличие настроенного мониторинга и логирования.

Цели, задачи и ожидаемые результаты:

Цели:

- Закрепить теоретические основы микросервисной архитектуры на практике.
- Развить навыки разработки, развертывания и управления микросервисами.

Задачи преподавателя:

- Разработка плана лабораторных работ (в соответствии с РПД, п. 2.3.3).
- Подготовка индивидуальных заданий с автотестами.
- Организация проверки через Git и CI/CD.

Ожидаемые результаты студентов:

- Закрепить теоретические основы микросервисной архитектуры на практике.
- Развить навыки разработки, развертывания и управления микросервисами.

Порядок реализации:

План лабораторных работ:

- ЛР1: Создание базового микросервиса с REST API (Spring Boot/Flask).
- ЛР2: Контейнеризация микросервиса (Docker + Docker Compose).
- ЛР3: Настройка взаимодействия между микросервисами (REST/gRPC).
- ЛР4: Реализация event-driven архитектуры с Apache Kafka.
- ЛР5: Оркестрация микросервисов в Kubernetes (Helm charts).
- ЛР6: Настройка CI/CD пайплайна (GitLab CI/GitHub Actions).
- ЛР7: Мониторинг микросервисов (Prometheus + Grafana).
- ЛР8: Логирование и трейсинг (ELK Stack, OpenTelemetry).
- ЛР9: Реализация аутентификации и авторизации (JWT, OAuth2).
- ЛР10: Нагрузочное тестирование и оптимизация производительности.

Пример индивидуального задания (ЛР4 — SVD):

Задача: Развернуть приложение из трех микросервисов в Kubernetes кластере

Автотесты: Проверка доступности эндпоинтов, балансировки нагрузки, горизонтального масштабирования

Контрольные вопросы:

- Закрепить теоретические основы микросервисной архитектуры на практике.
- Развить навыки разработки, развертывания и управления микросервисами.

Критерии оценки:

Зачтено: Основной функционал реализован, код соответствует требованиям, студент отвечает на контрольные вопросы.

Не зачтено: Код не работает, не соответствует ТЗ или студент не может объяснить принципы его работы.

Порядок проверки корректности:

Чек-лист:

- Наличие репозитория с выполненными заданиями.
- Успешное прохождение автотестов.
- Корректная работа развернутого приложения.

4.5. Методические указания по организации проектной деятельности студентов

Условия применения:

- Курс рассчитан на студентов 3 курса (6 семестр).
- Время на проект — до 20 часов на команду.
- Доступ к облачным платформам (Yandex Cloud, SberCloud) или локальным Kubernetes кластерам

Цели, задачи и ожидаемые результаты:

Цели:

- Применить знания микросервисной архитектуры для решения практических задач.
- Развить навыки командной работы и проектного менеджмента.

Задачи преподавателя:

- Подготовка кейсов (например, система электронной коммерции, аналитическая платформа).
- Формирование ТЗ для экзаменационного проекта.
- Разработка системы оценки результатов.

Ожидаемые результаты студентов:

- Готовый микросервисный проект с полным циклом CI/CD.
- Умение работать в команде и презентовать результаты.

Порядок реализации:

Примеры проектов:

- Разработка системы рекомендаций на микросервисах.
- Создание платформы для сбора и анализа данных с IoT-устройств.
- Построение отказоустойчивого API Gateway для агрегации сервисов.
- Реализация ETL-пайплайна в микросервисной архитектуре.

ТЗ для проекта «Применение РСА для визуализации данных»:

- Реализовать сервисы: каталог, корзина, заказы, пользователи.
- Настроить взаимодействие через REST API и message broker (Kafka).
- Обеспечить мониторинг и логирование.
- Настроить CI/CD для автоматического развертывания.

Критерии оценки:

Зачтено: Проект реализован в соответствии с ТЗ, демонстрирует понимание принципов микросервисной архитектуры.

Не зачтено: Проект не реализован или не соответствует базовым требованиям ТЗ.

Порядок проверки корректности:

- Наличие репозитория с кодом и документацией.
- Соответствие ТЗ.
- Качество кода и архитектурных решений.
- Полнота реализации заявленного функционала.

5. Перечень учебной литературы, информационных ресурсов и технологий

5.1. Учебная 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.»
4. . Кафка Д., Мурти Д. Нигам С. Микросервисная архитектура. Реализация и проектирование: Пер. с англ. — М.: ДМК-Пресс, 2017. — 432 с.
5. Фаулер М., Палмер Р., Льюис Джеймс. Микросервисы. Паттерны разработки и лучшие практики: Пер. с англ. — М.: ДМК-Пресс, 2018. — 510 с.
6. МакКроски Р. Микросервисы. Создание расширяемых и отказоустойчивых приложений: Пер. с англ. — М.: ДМК-Пресс, 2018. — 460 с.
7. Саммервиль Дж., Жерарди Р. Микросервисы в действии: Пер. с англ. — СПб.: Питер, 2018. — 304 с.
8. Ганри Ш. Микросервисы. Надежные и масштабируемые системы: Пер. с англ. — М.: ООО «ДМК Пресс», 2019. — 340 с.

9. Баджуска Д., Москетто Р. Микросервисы изнутри. Основы, инструменты и практические рекомендации: Пер. с англ. — М.: ДМК-Пресс, 2020. — 480 с.2.

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

5.2. Периодические издания:

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

5.3. 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. 4 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

5.5 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.

Кейсы ПАО «Сбербанк»

Кейс 1: Микросервисная архитектура для генеративного ИИ инвестиционных обзоров

Описание:

Сбербанк разрабатывает систему автоматического составления инвестиционных обзоров на основе LLM. Задача — спроектировать микросервисную архитектуру для обработки структурированных данных (биржевые котировки, макроэкономические показатели) и генерации аналитических отчетов.

Цель:

Разработать архитектуру из специализированных микросервисов: сбора данных, предобработки, генерации текста, валидации контента и доставки отчетов.

Ожидаемый результат:

- Схема взаимодействия микросервисов через message broker (Kafka)

- API Gateway для агрегации сервисов
- Конфигурация мониторинга качества генерации

Кейс 2: Распределенная система анализа жалоб клиентов

Описание:

Требуется построить отказоустойчивую систему семантического анализа обращений клиентов с использованием NLU-моделей в микросервисной архитектуре.

Цель:

Спроектировать систему из микросервисов классификации, тонального анализа, извлечения сущностей и маршрутизации обращений.

Ожидаемый результат:

- Архитектура с разделением ответственности между сервисами
- Реализация механизма повторной обработки сообщений
- Настройка централизованного логирования и трейсинга

Кейс 3: Микросервисы для синтетических банковских данных

Описание:

Разработка безопасного pipeline генерации синтетических банковских данных для обучения ML-моделей с соблюдением требований ЦБ и ФЗ-152.

Цель:

Создать систему из изолированных микросервисов генерации, валидации и анонимизации данных.

Ожидаемый результат:

- Архитектура с разделением доступа к данным
- Сервис оценки качества синтетических данных
- Механизмы обеспечения конфиденциальности

Кейсы от «АВА ЛАБ»

Кейс 1: Микросервисная платформа для мультимодального анализа строительных площадок

Описание:

Разработка системы мониторинга строительных объектов с обработкой изображений, видео и текстовых запросов в микросервисной архитектуре.

Цель:

Спроектировать систему из специализированных сервисов компьютерного зрения, NLP и агрегации данных.

Ожидаемый результат:

- Архитектура с асинхронной обработкой мультимодальных данных
- Сервис оркестрации workflow анализа
- Механизмы кеширования и оптимизации запросов

Кейс 2: Распределенная система генерации проектной документации

Описание:

Создание микросервисной платформы для генерации черновиков проектной документации на основе LLM с учетом нормативов и шаблонов.

Цель:

Разработать систему из сервисов управления шаблонами, генерации контента, валидации и экспорта документов.

Ожидаемый результат:

- Схема взаимодействия сервисов через REST API и message queue
- Сервис контроля соответствия нормативам
- Механизм версионирования документов

Кейс 3: Микросервисная архитектура для BI-системы с RAG**Описание:**

Построение микросервисной архитектуры для BI-системы Fastboard с использованием LLM и RAG для обработки запросов на естественном языке.

Цель:

Спроектировать систему из сервисов векторного поиска, классификации запросов, генерации SQL и визуализации данных.

Ожидаемый результат:

- Архитектура с разделением сервисов обработки данных и ML-моделей
- Сервис кеширования результатов запросов
- Мониторинг производительности и качества ответов

Общие рекомендации по реализации кейсов:

- Использовать containerization (Docker) для изоляции сервисов
- Применять orchestration (Kubernetes) для управления микросервисами
- Внедрить service mesh (Istio) для управления трафиком между сервисами
- Настроить centralized logging и distributed tracing
- Реализовать circuit breaker и retry mechanisms для устойчивости
- Обеспечить security best practices (mTLS, RBAC, secret management)

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.