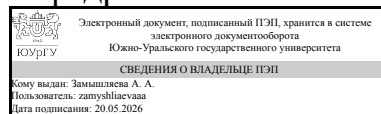


УТВЕРЖДАЮ:  
Заведующий выпускающей  
кафедрой



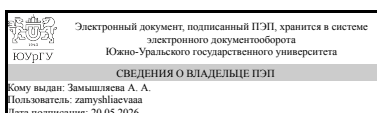
А. А. Замышляева

## РАБОЧАЯ ПРОГРАММА

**дисциплины 1.Ф.П0.06** Разработка цифровых сервисов с использованием LLM для направления 01.03.02 Прикладная математика и информатика  
**уровень** Бакалавриат  
**профиль подготовки** Искусственный интеллект, глубокое обучение и анализ данных  
**форма обучения** очная  
**кафедра-разработчик** Центр ОП топ-уровня в сфере ИИ "ВиртУм"

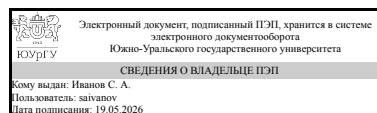
Рабочая программа составлена в соответствии с ФГОС ВО по направлению подготовки 01.03.02 Прикладная математика и информатика, утверждённым приказом Минобрнауки от 10.01.2018 № 9

Зав.кафедрой разработчика,  
д.физ.-мат.н., проф.



А. А. Замышляева

Разработчик программы,  
к.физ.-мат.н., доцент



С. А. Иванов

## 1. Цели и задачи дисциплины

Целью дисциплины является формирование у студентов практических навыков и системного понимания процесса разработки цифровых сервисов на основе больших языковых моделей (LLM), включая проектирование, настройку и интеграцию автономных ИИ-агентов и мультиагентных систем. Дисциплина направлена на подготовку специалистов, способных создавать интеллектуальные сервисы, автоматизирующие сложные задачи в бизнесе, образовании, поддержке пользователей и других сферах. Основные задачи дисциплины: познакомить студентов с архитектурой и принципами работы LLM-агентов; развить навыки проектирования и реализации агентов с памятью, инструментами и механизмами планирования; освоить технологии создания мультиагентных систем с различными стратегиями взаимодействия; научить интегрировать агентов в цифровые сервисы с использованием современных подходов к развертыванию (Docker, REST API, микросервисы).

## Краткое содержание дисциплины

Дисциплина посвящена практической разработке цифровых сервисов на основе LLM, с акцентом на создание и интеграцию автономных ИИ-агентов и мультиагентных систем. Курс охватывает весь цикл разработки — от проектирования архитектуры агента до его внедрения в реальные бизнес-процессы. В рамках дисциплины рассматриваются следующие разделы: архитектура LLM-агентов, разработка и настройка агентов, мультиагентные системы, интеграция агентов в сервисы. В рамках освоения дисциплины изучаются основные компоненты ИИ-агента: модель, память (краткосрочная и долгосрочная), набор инструментов (интеграция с API, поиск, исполнение кода), механизмы планирования задач (ReAct, DFS, BFS), рассматриваются типы агентов — от простых ботов до автономных систем с рефлексией, особое внимание уделяется мультимодальным агентам, способным обрабатывать текст, изображения и аудио, практическое освоение инструментов для создания агентов: LangChain, CrewAI, GigaChat API. Студенты учатся управлять контекстом, использовать кэширование промптов, реализовывать динамический выбор инструментов и обрабатывать "тупиковые" ситуации, изучаются принципы проектирования систем из нескольких взаимодействующих агентов, рассматриваются модели координации: кооперация (Autogen), конкуренция, иерархические структуры, архитектура исполнитель-критик, анализируются подходы к оркестрации (централизованная и децентрализованная), динамическое распределение задач, обмен знаниями и разрешение конфликтов между агентами, создание REST API для взаимодействия с агентом, упаковка сервиса в Docker-контейнер, обеспечение безопасности и масштабируемости, а также рассматриваются технологии оптимизации моделей (квантование GGUF/GPTQ, LoRA/QLoRA).

## 2. Компетенции обучающегося, формируемые в результате освоения дисциплины

Планируемые результаты освоения ОП ВО (компетенции)	Планируемые результаты обучения по дисциплине
ПК-18 [LLM-4] Проектирует, разрабатывает и	Знает: - [И-1, СУ] способы настройки агентов и

интегрирует интеллектуальных агентов на базе генеративных моделей	управления их контекстом и задачами Имеет практический опыт: - [И-3, СУ] организации взаимодействия между агентом и внешними источниками
ПК-19 [LLM-5] Организует взаимодействие с генеративными моделями через проектирование, анализ и применение промптов	Знает: - [И-1, СУ] базовые шаблоны промптов Умеет: - [И-2, СУ] применять цепочки (Chain of Thought) и условную логику [И-4, СУ] оптимизировать промпты под точность, длину, уменьшение галлюцинаций Имеет практический опыт: - [И-3, ПУ] разработки интерфейсов и систем взаимодействия с учётом UX/UI
ПК-24 [FC-2] Способен проводить передовые исследования в области фундаментальных и генеративных моделей	Знает: - [И-2, БУ] основы prompt engineering : zero- few-shot, chain-of-thought, prompt chaining; работы с векторными БД , RAG-системами и агентами Умеет: - [И-2, БУ] разрабатывать эффективные промты для решения задач с использованием LLM (zero- few-shot, chain-of-thought, prompt chaining, instruction tuning); продвинутых RAG-систем; агентов с вызовом функций Имеет практический опыт: - [И-2, БУ] создания эффективных промтов и агентов с помощью LangChain, CrewAI; реализации RAG-систем с использованием векторных БД (FAISS, Chroma)

### 3. Место дисциплины в структуре ОП ВО

Перечень предшествующих дисциплин, видов работ учебного плана	Перечень последующих дисциплин, видов работ
Основы компьютерного зрения, Интеллектуальная обработка естественного языка	Проектно-исследовательский семинар, Производственная практика (преддипломная, стажировка) (8 семестр)

Требования к «входным» знаниям, умениям, навыкам студента, необходимым при освоении данной дисциплины и приобретенным в результате освоения предшествующих дисциплин:

Дисциплина	Требования
Интеллектуальная обработка естественного языка	Знает: -[И-3, БУ] базовые адаптивные методы дообучения (prefix, adapter)[И-5, СУ] способы настройки пайплайнов с кастомными компонентами, -[И-2, БУ] способы создания простых последовательностей промптов, -[И-1, БУ] классические инструменты парсинга текстов: регулярные выражения, токенизация, морфологический анализ, синтаксический анализ[И-2, СУ] основные архитектуры сетей, используемые для векторизации текстовых данных: Word2Vec, Doc2Vec, Glove, FastText, рекуррентные нейронные сети и сети-трансформеры (энкодеры) Умеет: -[И-2, СУ] реализовать рассуждение на основе цепочек (ReAct, Plan&Solve), -[И-2, СУ] адаптировать и

	<p>валидировать датасеты под задачи обработки естественного языка[И-7, СУ] анализировать прирост метрик моделей в задачах обработки естественного языка в зависимости от этапов обучения[И-5, СУ] оптимизировать векторные базы данных; настраивать механизмы RAG; применять техники ускорения и повышения точности (reranking, rephrasing), -[И-3, СУ] управлять параметрами генерации для контроля результата настройки API при работе с LLM, -[И-2, СУ] самостоятельно найти подходящую модель для векторизации текстовых данных в открытых источниках и применить её для конкретной задачи[И-3, СУ] адаптировать и дорабатывать существующие архитектуры (например, fine-tuning BERT, GPT, T5) под конкретные задачи (классификация, генерация, NER); оптимизировать пайплайны обработки данных и обучения (ускорение через ONNX, Quantization, распределенные вычисления); строить CI/CD-процессы для NLP-моделей (тестирование, мониторинг дрейф данных) Имеет практический опыт: -[И-6, СУ] подбора параметров моделей под задачи обработки естественного языка с помощью grid и random search[И-1, СУ] применения fine-tune к предобученным моделям на новых датасетах[И-5, СУ] настройки retriever и reader под разные типы запросов, -[И-5, СУ] настройки system prompts и ввода ограничений, -[И-3, СУ] разворачивания сервисов в продакшн-среде (Docker, Kubernetes, облачные NLP-API)</p>
<p>Основы компьютерного зрения</p>	<p>Знает: -[И-2, СУ] отличия и способы применения нейронных сетей для отслеживания объектов (семейство R-CNN, YOLO), -[И-1, СУ] способы оптимизации гиперпараметров для улучшения качества; способы создания сложных пайплайнов аугментации (albuementations)[И-2, СУ] Нейросетевые архитектуры для анализа изображений VGG, Inception, ResNet, EfficientNet и т.д., особенности обучения и дообучения; архитектуры FCN и Unet в задачах сегментации, функции потерь для задачи сегментации; одностадийные (SSD, YOLO) и двухстадийные (FASTER R-CNN, Mask R-CNN) детекторы в задачах детекции, функции потерь в задаче детекции, -[И-1, СУ] принципы и методы переноса знаний и адаптации моделей в компьютерном зрении Умеет: -[И-2, СУ] применять принцип построения вычислительного блока Google Inception, -[И-1, СУ] работать с видео: извлечение кадров, обработка временных последовательностей[И-2, СУ] разрабатывать алгоритмы сегментации изображений (раделение-слияние регионов, нормализованный разрез графа, mean shift),</p>

	включая семантическую сегментацию; применять преобразование Хафа и RANSAC; применять алгоритмы детекции характеристических точек (детектор Харриса, детектор Фестнера, SUSAN, блобы, DoG); применять дескрипторы изображений, например, SIFT, -[И-1, БУ] применять существующие инструменты для реализации процессов переноса знаний и адаптации моделей Имеет практический опыт: -[И-2, СУ] разработки решений с применением backbone сетей, -[И-1, СУ] сравнения разных предобученных под конкретную задачу моделей; проведения transfer learning на своих данных
--	---

#### 4. Объём и виды учебной работы

Общая трудоёмкость дисциплины составляет 3 з.е., 108 ч., 54,5 ч. контактной работы

Вид учебной работы	Всего часов	Распределение по семестрам в часах	
		Номер семестра	
		7	
Общая трудоёмкость дисциплины	108	108	
<i>Аудиторные занятия:</i>	48	48	
Лекции (Л)	16	16	
Практические занятия, семинары и (или) другие виды аудиторных занятий (ПЗ)	0	0	
Лабораторные работы (ЛР)	32	32	
<i>Самостоятельная работа (СРС)</i>	53,5	53,5	
Разработка сервиса на основе GigaChat	43,5	43,5	
Подготовка к диф. зачёту	10	10	
Консультации и промежуточная аттестация	6,5	6,5	
Вид контроля (зачет, диф.зачет, экзамен)	-	диф.зачет	

#### 5. Содержание дисциплины

№ раздела	Наименование разделов дисциплины	Объем аудиторных занятий по видам в часах			
		Всего	Л	ПЗ	ЛР
1	Архитектура LLM-агентов	8	4	0	4
2	Разработка и настройка агентов	10	4	0	6
3	Мультиагентные системы	16	4	0	12
4	Интеграция агентов в сервисы	14	4	0	10

##### 5.1. Лекции

№ лекции	№ раздела	Наименование или краткое содержание лекционного занятия	Кол-во часов

1-2	1	Что такое ИИ-агент? LLM Agentic Workflow. Архитектура агента: модель, инструменты, память, планирование. Типы агентов: от простого бота до автономного агента с рефлексией. Рабочая память (short-term) vs. База знаний (long-term). Протоколы MCP и A2A. Инструменты: API, поиск, исполнение кода. Планирование задач (ReAct, DFS, BFS). Мультимодальные агенты (Llava, Gemma 3).	4
3-4	2	Работа с памятью и контекстом: краткосрочная, долгосрочная память, рефлексия. Настройка промптов: few-shot, zero-shot, Prompt chaining, flow, conditionals, промптинг-грамматика. Ограничения на генерацию: Token limits, temperature, top_p, context. Инструменты для разработки агентов: LangChain (GigaChat), CrewAI. Кэширование промптов. Динамический выбор инструментов (Router chains). Обработка "тупиковых" сценариев.	4
5-6	3	Основы мультиагентных систем: коммуникация, координация, конфликты. Проектирование взаимодействия между агентами: централизованное и децентрализованное управление. Обмен знаниями. Кооперация (Autogen) vs. Конкуренция. Архитектура исполнитель-критик. Ролевые модели агентов. Оркестрация: Centralized vs. Decentralized. Иерархические мультиагентные системы. Мастер-агент. Динамическое перераспределение задач.	4
7	4	Внедрение агентов в бизнес-процессы: чатботы, помощники, автоматизация задач. Сервисная архитектура: REST API, микросервисы, Docker, безопасность, квантование моделей (GGUF, GPTQ). LoRA/QLoRA для кастомных агентов. Развертывание: FastAPI + Docker. Асинхронные агенты (Celery/RabbitMQ).	2
8	4	Изучение кейсов партнёров: 1. Сервис по распознаванию текста. 2. Сервис для консультаций по вопросам получения мер государственной поддержки. 3. Ответы на вопросы по документации 4. Работа с базой данных	2

## 5.2. Практические занятия, семинары

Не предусмотрены

## 5.3. Лабораторные работы

№ занятия	№ раздела	Наименование или краткое содержание лабораторной работы	Кол-во часов
1-2	1	Создание агента с инструментами (поиск в Google, калькулятор, Wolfram Alpha)	4
3-5	2	Реализация агента, способного выполнять задачи из пользовательского запроса. Отладка и тестирование поведения агента. Например: Агент для анализа данных (Pandas + SQL + LLM); агент-ассистент в Telegram с долгосрочной памятью; голосовой агент (Whisper → LLM → TTS) и тд.	6
6-7	3	Создание системы из двух и более взаимодействующих агентов	4
8-9	3	Реализация диалога между агентами для достижения общей цели	4
10	3	Сравнение результатов разных стратегий взаимодействия	2
11	3	Отладка конфликтов и несогласованных действий агентов	2
12-13	4	Разработка REST API для взаимодействия с агентом	4
14-15	4	Сборка и запуск агентного сервиса в Docker	4
Защита проекта:	4	Защита проекта: демонстрация и презентация своего LLM-	2

демонстрация и презентация своего LLM-агента или сервиса		агента или сервиса	
--	--	--------------------	--

#### 5.4. Самостоятельная работа студента

Выполнение СРС			
Подвид СРС	Список литературы (с указанием разделов, глав, страниц) / ссылка на ресурс	Семестр	Кол-во часов
Разработка сервиса на основе GigaChat	Все материалы основной литературы, материалы презентаций, материалы лабораторных работ	7	43,5
Подготовка к диф. зачёту	Все материалы основной литературы, материалы презентаций	7	10

#### 6. Фонд оценочных средств для проведения текущего контроля успеваемости, промежуточной аттестации

Контроль качества освоения образовательной программы осуществляется в соответствии с Положением о балльно-рейтинговой системе оценивания результатов учебной деятельности обучающихся.

##### 6.1. Контрольные мероприятия (КМ)

№ КМ	Се-местр	Вид контроля	Название контрольного мероприятия	Вес	Макс. балл	Порядок начисления баллов	Учи-тыва-ется в ПА
1	7	Текущий контроль	Создание агента с инструментами (поиск в Google, калькулятор, Wolfram Alpha)	3	3	На лабораторной работе выдается базовый вариант python-кода и 3 задания к нему. Студент выполняет задания, загружает отчет на портал и показывает преподавателю. Каждое верно решенное задание 1 балл.	дифференцированный зачет
2	7	Текущий контроль	Реализация агента, способного выполнять задачи из пользовательского запроса. Отладка и тестирование поведения агента.	3	3	На лабораторной работе выдается базовый вариант python-кода и 3 задания к нему. Студент выполняет задания, загружает отчет на портал и показывает преподавателю. Каждое верно решенное задание 1 балл.	дифференцированный зачет
3	7	Текущий контроль	Создание системы из двух и более взаимодействующих агентов	3	3	На лабораторной работе выдается базовый вариант python-кода и 3 задания к нему. Студент выполняет задания,	дифференцированный зачет

						загружает отчет на портал и показывает преподавателю. Каждое верно решенное задание 1 балл.	
4	7	Текущий контроль	Реализация диалога между агентами для достижения общей цели	3	3	На лабораторной работе выдается базовый вариант python-кода и 3 задания к нему. Студент выполняет задания, загружает отчет на портал и показывает преподавателю. Каждое верно решенное задание 1 балл.	дифференцированный зачет
5	7	Текущий контроль	Сравнение результатов разных стратегий взаимодействия	3	3	На лабораторной работе выдается базовый вариант python-кода и 3 задания к нему. Студент выполняет задания, загружает отчет на портал и показывает преподавателю. Каждое верно решенное задание 1 балл.	дифференцированный зачет
6	7	Текущий контроль	Отладка конфликтов и несогласованных действий агентов	3	3	На лабораторной работе выдается базовый вариант python-кода и 3 задания к нему. Студент выполняет задания, загружает отчет на портал и показывает преподавателю. Каждое верно решенное задание 1 балл.	дифференцированный зачет
7	7	Текущий контроль	Разработка REST API для взаимодействия с агентом	3	3	На лабораторной работе выдается базовый вариант python-кода и 3 задания к нему. Студент выполняет задания, загружает отчет на портал и показывает преподавателю. Каждое верно решенное задание 1 балл.	дифференцированный зачет
8	7	Текущий контроль	Сборка и запуск агентного сервиса в Docker	3	3	На лабораторной работе выдается базовый вариант python-кода и 3 задания к нему. Студент выполняет задания, загружает отчет на портал и показывает преподавателю. Каждое верно решенное задание 1 балл.	дифференцированный зачет

9	7	Текущий контроль	Защита проекта: демонстрация и презентация своего LLM-агента или сервиса	10	10	<p>На защите студент (или команда) представляет:</p> <ol style="list-style-type: none"> <li>1. Работающий сервис — LLM-агент или мультиагентная система.</li> <li>2. Презентацию, включающую: <ul style="list-style-type: none"> <li>- Цель проекта и пользовательские сценарии.</li> <li>- Архитектуру агента (диаграмма с компонентами: LLM, память, инструменты, API).</li> <li>- Демонстрацию работы (скринкаст или живой запуск).</li> </ul> </li> <li>3. Краткий отчёт (3–5 страниц) с описанием: <ul style="list-style-type: none"> <li>- Цели и задачи.</li> <li>- Реализованных функций (память, инструменты, взаимодействие).</li> <li>- Проблем и решений.</li> <li>- Исходный код проекта в репозитории (GitHub, GitLab и др.) с инструкцией по запуску и файлом <code>docker-compose.yml</code>.</li> </ul> </li> </ol> <p>Критерии оценки:  Реализация архитектуры агента (память, инструменты, планирование) :2  Качество интеграции (API, Docker, асинхронность): 2  Работоспособность и устойчивость сервиса: 2  Демонстрация и презентация: 2  Отчёт и документация: 2  Итого: 10</p>	дифференцированный зачет
10	7	Текущий контроль	Тест 1	1	1	<p>Тест проводится по теме лекции. В тесте 3 вопроса. На прохождение теста дается одна попытка. Тест доступен только во время лекции.</p>	дифференцированный зачет

						Ограничение по времени - 9 минут. Вес каждого вопроса – 0.33. Максимальное количество баллов за тест – 1 балл.	
11	7	Текущий контроль	Тест 2	1	1	Тест проводится по теме лекции. В тесте 3 вопроса. На прохождение теста дается одна попытка. Тест доступен только во время лекции. Ограничение по времени - 9 минут. Вес каждого вопроса – 0.33. Максимальное количество баллов за тест – 1 балл.	дифференцированный зачет
12	7	Текущий контроль	Тест 3	1	1	Тест проводится по теме лекции. В тесте 3 вопроса. На прохождение теста дается одна попытка. Тест доступен только во время лекции. Ограничение по времени - 9 минут. Вес каждого вопроса – 0.33. Максимальное количество баллов за тест – 1 балл.	дифференцированный зачет
13	7	Текущий контроль	Тест 4	1	1	Тест проводится по теме лекции. В тесте 3 вопроса. На прохождение теста дается одна попытка. Тест доступен только во время лекции. Ограничение по времени - 9 минут. Вес каждого вопроса – 0.33. Максимальное количество баллов за тест – 1 балл.	дифференцированный зачет
14	7	Промежуточная аттестация	Итоговое тестирование	-	20	В финальном тесте 20 вопросов. Каждый вопрос оценивается 1 баллом. Ограничение по времени на прохождение теста - 40 минут. Вопросы выбираются случайным образом из всех разделов дисциплины, по 4-5 вопросов из	дифференцированный зачет

					каждой темы.	
--	--	--	--	--	--------------	--

## 6.2. Процедура проведения, критерии оценивания

Вид промежуточной аттестации	Процедура проведения	Критерии оценивания
дифференцированный зачет	<p>При оценивании результатов учебной деятельности обучающегося по дисциплине используется балльно-рейтинговая система оценивания результатов учебной деятельности обучающихся (Положение о БРС утверждено приказом ректора от 24.05.2019 г. № 179, в редакции приказа ректора от 10.03.2022 г. № 25-13/09).</p> <p>Процедура прохождения промежуточной аттестации осуществляется согласно Положению о текущем контроле успеваемости и промежуточной аттестации (приказ ректора от 27.02.2024 № 33-13/09). Оценка за дисциплину формируется на основе полученных оценок за контрольно-рейтинговые мероприятия текущего контроля следующим образом: • Отлично: Величина рейтинга обучающегося по дисциплине 85...100 %. • Хорошо: Величина рейтинга обучающегося по дисциплине 75...84 %. • Удовлетворительно: Величина рейтинга обучающегося по дисциплине 60...74 %. • Неудовлетворительно: Величина рейтинга обучающегося по дисциплине 0...59 %. Если студент согласен с оценкой, полученной по результатам текущего контроля, то он может в день, предшествующий промежуточной аттестации дать свое согласие на автомат в личном кабинете. В случае явки студента на промежуточную аттестацию, давшего свое согласие на автомат в личном кабинете, студент имеет право пройти мероприятия текущего контроля по дисциплине на промежуточной аттестации для улучшения своего рейтинга в день ее проведения. Снижение оценки в этом случае запрещено. Если студент не дал согласия в личном кабинете, то он может согласиться с оценкой лично на промежуточной аттестации в день ее проведения. Если студент не согласен с оценкой, то он имеет право пройти контрольно-рейтинговые мероприятия на промежуточной аттестации для улучшения своего рейтинга в день ее проведения.</p> <p>Фиксация результатов учебной деятельности по дисциплине проводится в день промежуточной аттестации на основе согласия студента, данного им в личном кабинете. При отсутствии согласия в журнале дисциплины фиксация результатов происходит при личном присутствии студента. Если студент не дал согласие в личном кабинете и не явился на промежуточную аттестацию – ему выставляется «неявка». Промежуточная аттестация проводится в форме тестирования. Тестирование проводится в системе edu.susu.ru, тест содержит 20 вопросов, на выполнение теста дается 40 минут. В этом случае оценка за дисциплину рассчитывается на основе полученных оценок за контрольно-рейтинговые мероприятия текущего контроля и промежуточной аттестации.</p>	В соответствии с пп. 2.5, 2.6 Положения

### 6.3. Паспорт фонда оценочных средств

Компетенции	Результаты обучения	№ КМ													
		1	2	3	4	5	6	7	8	9	10	11	12	13	14
ПК-18	Знает: - [И-1, СУ] способы настройки агентов и управления их контекстом и задачами	+	+	+	+	+	+	+	+	+	+	+	+	+	+
ПК-18	Имеет практический опыт: - [И-3, СУ] организации взаимодействия между агентом и внешними источниками	+	+	+	+	+	+	+	+			+		+	
ПК-19	Знает: - [И-1, СУ] базовые шаблоны промптов	+	+	+	+	+	+	+	+	+	+	+	+	+	
ПК-19	Умеет: - [И-2, СУ] применять цепочки (Chain of Thought) и условную логику [И-4, СУ] оптимизировать промпты под точность, длину, уменьшение галлюцинаций	+	+	+	+	+	+	+	+		+			+	
ПК-19	Имеет практический опыт: - [И-3, ПУ] разработки интерфейсов и систем взаимодействия с учётом UX/UI		+	+	+	+	+		+					+	
ПК-24	Знает: - [И-2, БУ] основы prompt engineering : zero- few-shot, chain-of-thought, prompt chaining; работы с векторными БД , RAG-системами и агентами	+	+	+	+	+	+	+	+	+	+	+	+	+	
ПК-24	Умеет: - [И-2, БУ] разрабатывать эффективные промты для решения задач с использованием LLM (zero- few-shot, chain-of-thought, prompt chaining, instruction tuning); продвинутых RAG-систем; агентов с вызовом функций	+	+	+	+	+	+	+	+					+	
ПК-24	Имеет практический опыт: - [И-2, БУ] создания эффективных промтов и агентов с помощью LangChain, StegAI; реализации RAG-систем с использованием векторных БД (FAISS, Chroma)	+	+	+	+	+	+	+	+					+	

Типовые контрольные задания по каждому мероприятию находятся в приложениях.

### 7. Учебно-методическое и информационное обеспечение дисциплины

#### Печатная учебно-методическая документация

а) *основная литература:*

Не предусмотрена

б) *дополнительная литература:*

Не предусмотрена

в) *отечественные и зарубежные журналы по дисциплине, имеющиеся в библиотеке:*

Не предусмотрены

г) *методические указания для студентов по освоению дисциплины:*

1. Разработка сервисов на основе Gigachat

*из них: учебно-методическое обеспечение самостоятельной работы студента:*

1. Разработка сервисов на основе Gigachat

#### Электронная учебно-методическая документация

№	Вид литературы	Наименование ресурса в электронной	Библиографическое описание

		форме	
1	Основная литература	ЭБС издательства Лань	Девятков, В. В. Системы искусственного интеллекта : учебник / В. В. Девятков. — Москва : МГТУ им. Баумана, 2023. — 279 с. — ISBN 978-5-7038-5939-1. — Текст : электронный // Лань : электронно-библиотечная система. <a href="https://e.lanbook.com/book/461009">https://e.lanbook.com/book/461009</a>
2	Дополнительная литература	ЭБС издательства Лань	Эль, А. А. GPT-4. Руководство по использованию API Open AI : руководство / А. А. Эль ; перевод с английского В. С. Яценкова. — Москва : ДМК Пресс, 2024. — 274 с. — ISBN 978-5-93700-299-0. — Текст : электронный // Лань : электронно-библиотечная система. <a href="https://e.lanbook.com/book/456752">https://e.lanbook.com/book/456752</a>
3	Дополнительная литература	ЭБС издательства Лань	Ананченко, И. В. Контейнеризатор приложений docker — установка, настройка, основы управления приложениями в средах с поддержкой контейнеризации : учебно-методическое пособие / И. В. Ананченко, Т. В. Зудилова, С. Е. Иванов. — Санкт-Петербург : НИУ ИТМО, 2023. — 54 с. — Текст : электронный // Лань : электронно-библиотечная система. <a href="https://e.lanbook.com/book/460064">https://e.lanbook.com/book/460064</a>
4	Основная литература	eLIBRARY.RU	Yunfan Shao, Linyang Li, Junqi Dai, and Xipeng Qiu. 2023. Character-LLM: A Trainable Agent for Role-Playing. In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pages 13153–13187, Singapore. Association for Computational Linguistics. URL - <a href="https://aclanthology.org/2023.emnlp-main.814/">https://aclanthology.org/2023.emnlp-main.814/</a> <a href="https://www.elibrary.ru/item.asp?id=67941960">https://www.elibrary.ru/item.asp?id=67941960</a>
5	Основная литература	eLIBRARY.RU	Deuksin Kwon, Sunwoo Lee, Ki Hyun Kim, Seojin Lee, Taeyoon Kim, and Eric Davis. 2023. What, When, and How to Ground: Designing User Persona-Aware Conversational Agents for Engaging Dialogue. In Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 5: Industry Track), pages 707–719, Toronto, Canada. Association for Computational Linguistics. URL - <a href="https://aclanthology.org/2023.acl-industry.68/">https://aclanthology.org/2023.acl-industry.68/</a> <a href="https://www.elibrary.ru/item.asp?id=67939402">https://www.elibrary.ru/item.asp?id=67939402</a>

Перечень используемого программного обеспечения:

Нет

Перечень используемых профессиональных баз данных и информационных справочных систем:

Нет

## 8. Материально-техническое обеспечение дисциплины

Не предусмотрено