

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. В.ДВ.04.02 Разработка гибридных интеллектуальных систем

Направление подготовки 02.03.02 Фундаментальная информатика и
информационные технологии

Профиль Современные методы машинного обучения и компьютерного зрения

Форма обучения очная

Квалификация бакалавр

Краснодар 2025

Рабочая программа дисциплины «Б1.В.ДВ.04.01 Коллективная разработка информационных систем» составлена в соответствии с федеральным государственным образовательным стандартом высшего образования (ФГОС ВО) по направлению подготовки 01.03.02 Прикладная математика и информатика.

Программу составил(и):

Добровольская Н.Ю. доцент, канд. пед. наук, доцент

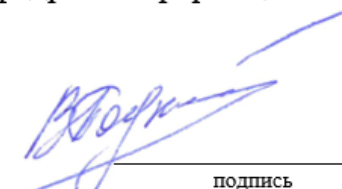


подпись

Рабочая программа обсуждена на заседании кафедры информационных технологий протокол № 1 «26» августа 2025г.

Заведующий кафедрой


Подколзин В.В.



подпись

Утверждена на заседании учебно-методической комиссии факультета компьютерных технологий и прикладной математики протокол № 1 «28» августа 2025г.

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

Сформировать у студентов комплексный опыт участия в полном жизненном цикле разработки промышленного ИИ-продукта в составе кросс-функциональной команды, от постановки задачи от бизнеса до внедрения работающего MVP, интегрирующего современные подходы (RAG, агенты, MLOps).

1.2 Задачи дисциплины

Отработать принципы Agile (Scrum/KanBan) на практике, включая ролевую модель, артефакты (бэклог, спринты) и регулярные события (планирование, стендапы, демо, ретроспективы).

Научиться формализовывать бизнес-требования в технические задания на систему в целом и на ИИ-компоненты в частности, с учетом ограничений и метрик успеха.

Спроектировать и согласовать архитектуру гибридной системы, включающей backend, frontend, мобильное приложение, ML-пайплайн и инфраструктурные сервисы.

Организовать параллельную исследовательскую (поиск и валидация ML-моделей) и инженерную (разработка кода, CI/CD, тестирование) работу в команде.

Обеспечить интеграцию всех компонент, тестирование и демонстрацию работоспособного MVP партнеру, подготовив пайплайн для его дальнейшего развития..

1.3 Место дисциплины (модуля) в структуре образовательной программы

Дисциплина «Разработка гибридных интеллектуальных систем» относится к «Часть, формируемая участниками образовательных отношений» Блока 1 «Дисциплины по выбору Б1.В.ДВ.4» учебного плана. Данная дисциплина использует знания студентов, полученные на дисциплинах – Web – разработка, современные методы машинного обучения, обработка естественного языка, генеративные нейронные сети, современные методы компьютерного зрения, микросервисная архитектура, современные методы компьютерного зрения, Мультиагентные системы, разработка мобильных приложений, Системы искусственного интеллекта, DataOps & ML Ops, DevOps, Объектно-ориентированное программирование и шаблоны проектирования, Технологии управления данными NoSQL, Подготовка данных машинного обучения, Анализ данных машинного обучения, Технологии обработки больших данных.

1.4 Профессиональные роли в структуре образовательной программы

Роль 1: Data Engineer (Инженер по данным)

Задачи:

1. Проектирование и построение ETL-процессов
2. Создание и оптимизация хранилищ данных
3. Обеспечение качества и доступности данных
4. Настройка инфраструктуры для обработки больших данных
5. Интеграция разрозненных источников данных
6. Работа с данными в области природопользования, медицины, связи и телекоммуникаций

Роль 2: ML Engineer (Инженер МО)

Задачи:

1. Реализация ML-моделей в продуктивных системах
2. Оптимизация производительности и масштабирование моделей
3. Разработка ML-пайплайнов и автоматизация процессов
4. Мониторинг качества моделей в продуктиве
5. Интеграция ML-решений с бизнес-приложениями

Роль 3: MLOps (Специалист по эксплуатации ИИ)

Задачи:

1. Автоматизация процессов обучения и развертывания моделей
2. Мониторинг производительности ML-систем
3. Управление версиями моделей и данных
4. Обеспечение CI/CD для ML-проектов
5. Оптимизация вычислительных ресурсов

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:

УК-1	Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач
УК-1.2	Выбирает оптимальный вариант решения задачи, аргументируя свой выбор
	Знать: Критерии сравнения альтернативных решений. Уметь: Проводить сравнительный анализ вариантов. Владеть: Навыками аргументации и принятия решений.
УК-3	Способен осуществлять социальное взаимодействие и реализовывать свою роль в команде
УК-3.2	Применяет методы командного взаимодействия; планирует и организует командную работу
	Знать: Методы и инструменты организации командной работы. Уметь: Распределять задачи и контролировать их выполнение. Владеть: Техниками командной координации и планирования.
УК-5	Способен воспринимать межкультурное разнообразие общества в социально-историческом, этическом и философском контекстах
УК-5.2	Интерпретирует проблемы современности с позиции этики и философских знаний
	Знать: Современные социально-этические проблемы. Уметь: Применять этические frameworks для анализа. Владеть: Навыками этической оценки технологий.
ПК-3	Способен применять основные алгоритмические и программные решения в области информационно-коммуникационных технологий, а также участвовать в их разработке
ПК-3.2	Определяет элементы проблемной области и их взаимодействие, архитектуру программной системы, ее функциональные возможности и логику работы с использованием основных концептуальных положений функционального, логического, объектно-ориентированного и визуального направлений программирования
	Знать: Принципы проектирования архитектуры ПО. Уметь: Проектировать компоненты системы и их взаимодействие. Владеть: Навыками проектирования программной архитектуры.
ПК-4	Способен планировать необходимые ресурсы и этапы выполнения работ в области информационно-коммуникационных технологий, составлять соответствующие технические описания и инструкции
ПК-4.2	Применяет современные приемы работы с инструментальными средствами, поддерживающими создание программных продуктов и программных комплексов на базе языков программирования, баз данных и пакетов прикладных программ
	Знать: Приемы работы с инструментальными средствами. Уметь: Использовать инструменты для создания программных комплексов.

	Владеть: Техниками интеграции различных инструментов.
ПК-4.3	Способен использовать методы эффективного управления командой при разработке, внедрении и сопровождении программных продуктов
	Знать: Методы управления командой в IT-проектах. Уметь: Организовывать работу команды на разных этапах. Владеть: Практиками управления командой разработки.
SS-2	<i>Способен осуществлять свою трудовую деятельность с учётом необходимости эффективной коммуникации и взаимодействия в рамках коллективной проектной работы в сфере ИИ</i>
SS-2.1	Эффективно коммуницирует с участниками проектной команды при планировании, реализации и анализе результатов работы
	Знать: Принципы эффективной коммуникации в проектах. Уметь: Ясно излагать идеи и задачи команде. Владеть: Навыками проведения эффективных встреч.
SS-2.2	Учитывает профессиональные и ролевые особенности коллег при совместной разработке технических решений и представлении результатов
	Знать: Особенности ролей в проектной команде по ИИ. Уметь: Адаптировать стиль коммуникации под аудиторию. Владеть: Методами междисциплинарного взаимодействия.
SS-3	<i>Способен осуществлять свою трудовую функцию с учетом неопределенности как сущностной черты функционирования искусственного интеллекта</i>
SS-3.2	Определяет релевантность применения ИИ для решения конкретных задач, анализирует поведение ИИ в техническом, социальном и правовом контекстах, переносит идеи и методы за пределы исходной предметной области
	Знать: Критерии применимости ИИ для решения задач. Уметь: Анализировать поведение ИИ в различных контекстах. Владеть: Методами анализа границ применимости моделей.
LC-1	<i>Способен проводить анализ бизнес-проблем с оценкой перспективности применения ИИ для их решения, осуществлять постановку задачи машинного обучения, формулировать требования к системе ИИ</i>
LC-1.1	Оценивает технические требования и разрабатывает техническое задание на системы искусственного интеллекта в конкретной предметной области
	Знать: Структуру и содержание ТЗ на систему ИИ. Уметь: Формализовывать бизнес-требования в технические. Определяет и формализует проблему предметной области, решение которой требует применения искусственного интеллекта Владеть: Методами разработки технического задания.
LC-4.1	<i>Способен управлять процессом жизненного цикла ИИ-продукта</i>
LC-4.1.1	Осуществляет запуск и ведение проекта в области ИИ, в том числе планирование и контроль задач, оценку ресурсов
	Знать: Уметь: Осуществляет ведение (запуск и управление) проектов в области ИИ, в том числе подбор команды, планирование и контроль задач, оценка ресурсов Владеть:
LC-4.2	<i>Способен руководить работой команды проекта в области ИИ</i>
LC-4.2.1	Координирует и контролирует работу команд проекта с целью достижения общих целей проекта
	Знать: Методы и инструменты координации командной работы в проектах.

Уметь: Обеспечивает взаимодействие всех участников проекта, включая бизнес-лидеров, команду Data Science и разработчиков ПО Владеть: Техниками контроля сроков и качества выполнения работ для достижения целей проекта
--

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)					
		8					
Контактная работа, в том числе:	30,2	30,2					
Аудиторные занятия (всего):	28	28					
Занятия лекционного типа	14	14					
Лабораторные занятия	14	14					
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:	41,8	41,8					
Курсовая работа							
Проработка учебного (теоретического) материала	20	20					
Выполнение индивидуальных заданий (подготовка сообщений, презентаций)	21,8	21,8					
Реферат							
Подготовка к текущему контролю							
Контроль:							
Подготовка к экзамену							
Общая трудоемкость	час.	72	72				
	в том числе контактная работа	30,2	30,2				
	зач. ед	2	2				

2.2 Структура дисциплины

Распределение видов учебной работы и их трудоёмкости по разделам дисциплины.

Разделы (темы) дисциплины, изучаемые в 8 семестре

№	Наименование разделов (тем)	Количество часов				
		Всего	Аудиторная работа			Внеаудиторная работа
			Л	ПЗ	ЛР	
1	2	3	4	5	6	7
1.	Процессы в ИТ.	10	2		2	6
2.	Специфика управления ИИ проектом.	10	2		2	6
3.	Архитектура гибридных интеллектуальных систем.	10	2		2	6

№	Наименование разделов (тем)	Количество часов				
		Всего	Аудиторная работа			Внеаудиторная работа
			Л	ПЗ	ЛР	СРС
1	2	3	4	5	6	7
4.	Инфраструктура и MLOps для быстрого и стабильного цикла.	10	2		2	6
5.	Паттерны интеграции современных ИИ-компонент: RAG и агенты	10	2		2	6
6.	Качество кода.	10	2		2	6
7.	Финальная демонстрация.	9,8	2		2	5,8
ИТОГО по разделам дисциплины		69,8	14		14	41,8
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.	Процессы в IT.	От индивидуального гения к командному результату: процессы в IT. Рассматриваем эволюцию методологий разработки от Waterfall к Agile. Детально разбираем Scrum и KanBan: роли, артефакты, события. Акцент на роли Product Owner (в лице партнера) и необходимости самоорганизации команды. Обсуждаем типичные антипаттерны в студенческих проектах и как их избежать. Формулируем ключевые метрики успеха проекта: бизнес-ценность, работающий продукт, удовлетворенность команды..	Т, ЛР
2.	Специфика управления ИИ проектом.	ИИ-проект – это особо сложный зверь. Специфика управления. Анализируем "двойную петлю" разработки в ИИ-проектах: инженерная разработка vs. исследовательская деятельность. Обсуждаем управление гипотезами, экспериментами и неопределенностью. Разбираем структуру ТЗ на ИИ-компонент: метрики качества (business, ML, engineering), требования к данным, критерии приемки. Говорим о важности быстрого построения baseline (например, на правилах или классическом ML) для оценки прироста от сложных моделей.	Т, ЛР
3.	Архитектура гибридных интеллектуальных систем.	Архитектура гибридных интеллектуальных систем.	Т, ЛР

№	Наименование раздела (темы)	Содержание раздела (темы)	Форма текущего контроля
1	2	3	4
		Строим эталонную архитектуру MVP, объединяющую web/mobile-клиенты, API Gateway, микросервисы (backend-for-frontend, сервис с ИИ-ядром), базы данных (SQL/NoSQL), брокеры сообщений. Особое внимание на проектирование ИИ-сервиса: модели как артефакты, сервис инференса, пайплайн обработки данных и переобучения. Обсуждаем компромиссы между скоростью, стоимостью и точностью при выборе моделей и инфраструктуры.	
4.	Инфраструктура и MLOps для быстрого и стабильного цикла.	Инфраструктура и MLOps для быстрого и стабильного цикла. Показываем, как поднять инфраструктуру в вузовском кластере или облаке партнера с помощью Docker и Kubernetes. Объясняем принципы CI/CD для ML: версионирование кода, данных и моделей (DVC, MLflow); автоматическое тестирование и развертывание. Разбираем пайплайн сбора фидбека для дообучения моделей. Делаем акцент на логировании, мониторинге и отказоустойчивости сервисов.	Т, ЛР
5.	Паттерны интеграции современных ИИ-компонент: RAG и агенты	Паттерны интеграции современных ИИ-компонент: RAG и агенты. Глубокое погружение в проектирование RAG-систем: чанкинг документов, выбор эмбединг-моделей, векторные базы данных, стратегии реранкинга и промпт-инжиниринг для LLM. Обсуждаем проектирование агентов: архитектуры (ReAct, Planner-Executor), управление контекстом и памятью, инструменты (tools) для взаимодействия с внешним миром. Разбираем кейсы, где гибридный RAG+агенты решает реальные бизнес-задачи.	Т, ЛР
6.	Качество кода.	Качество не только кода: тестирование, безопасность, документирование. Обсуждаем пирамиду тестирования в полномасштабном проекте: unit-тесты для бизнес-логики и ML-кода, интеграционные тесты для API и взаимодействия с БД, контрактное тестирование. Затрагиваем основы безопасности: OWASP Top 10 для web, безопасность ML-моделей (атаки, инверсия). Учим писать минимально достаточную документацию: ADR (Architecture Decision Record), README, онбординг для новых разработчиков.	Т, ЛР
7.	Финальная демонстрация.	Финальный рывок: демо, презентация, рефлексия и развитие проекта. Готовимся к финальной демонстрации: как структурировать pitch для бизнес-партнера (проблема, решение, метрики, демо). Учимся	ЛР

№	Наименование раздела (темы)	Содержание раздела (темы)	Форма текущего контроля
1	2	3	4
		давать конструктивную обратную связь по проектам других команд. Проводим итоговую ретроспективу: что прошло хорошо, что можно улучшить в процессах. Обсуждаем roadmap развития MVP: какие гипотезы проверить следующими, как масштабировать систему, что передать партнеру для production.	

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.	Процессы в IT.	Лаб 1 (Спринт 0): Формирование команд и инициация проекта. Команды определяют роли (Scrum-мастер, аналитики, разработчики и т.д.) и выбирают методологию (Scrum или KanBan). На основании вводного брифа от условного партнера (преподаватель) проводят первичный анализ требований. Формируют первоначальный Product Backlog в Jira/YouTrack, выделяют цели первого спринта (Sprint Goal). Создают Git-репозиторий с README и базовой структурой проектов. Результат: Утвержденный бэклог, выбранные инструменты, распределение ролей, репозиторий.	Т, ЛР
2.	Специфика управления ИИ проектом.	Лаб 2 (Спринт 1): Уточнение требований и проектирование архитектуры. Системные и бизнес-аналитики детализируют пользовательские истории, разрабатывают UML-диаграммы (Use Case, Activity). Архитекторы предлагают и защищают	Т, ЛР

№	Наименование раздела (темы)	Наименование лабораторных работ	Форма текущего контроля
1	2	3	4
		перед "архитектурным советом" (преподаватель) высокоуровневую схему системы и схему взаимодействия ИИ-компонентов. Команда принимает решение о технологическом стеке для каждой части. Результат: Детализированный бэклог с критериями приемки, утвержденная архитектурная диаграмма, технологический стек.	
3.	Архитектура гибридных интеллектуальных систем.	Лаб 3 (Спринт 2): Инфраструктура и исследовательский прототип ИИ. MLOps-инженеры и администраторы поднимают dev-окружение: настраивают Docker-контейнеры, разворачивают в кластере базовые сервисы (БД, брокер). ML-исследователи параллельно работают над экспериментами: подбирают и обучают модели, строят baseline, валидируют гипотезы по данным партнера. Бэкенд-разработчики делают заготовки сервисов. Результат: Работающее dev-окружение, Jupyter-ноутбук с доказательством работоспособности выбранного ML-подхода, заделы API.	Т, ЛР
4.	Инфраструктура и MLOps для быстрого и стабильного цикла.	Лаб 4 (Спринт 3): Реализация ядра системы и ML-пайплайна. Бэкенд-разработчики реализуют ключевые API, интегрируются с БД. MLOps-инженеры "продакшинизируют" лучшую модель из исследований: упаковывают в сервис, настраивают CI/CD пайплайн для её автоматического тестирования и деплоя. Фронтенд-разработчики начинают верстать интерфейсы по готовым mock-up. Результат: Работающие API core-логики, развернутый ML-сервис с документацией к API, CI-пайплайн для модели.	Т, ЛР
5.	Паттерны интеграции современных ИИ-компонент: RAG и агенты	Лаб 5 (Спринт 4): Интеграция и первый end-to-end сценарий. Команда интегрирует frontend, backend и ML-сервис для реализации одного сквозного пользовательского сценария (например, загрузка данных и получение предсказания). Пишутся и запускаются	Т, ЛР

№	Наименование раздела (темы)	Наименование лабораторных работ	Форма текущего контроля
1	2	3	4
		первые интеграционные тесты. Начинается разработка мобильного приложения (Kotlin). Проводится демо первого рабочего сценария. Результат: Работающий сквозной сценарий, набор интеграционных тестов, прототип мобильного приложения.	
6.	Качество кода.	Лаб 6 (Спринт 5): Доработка функционала, тестирование, исправление ошибок. Команда наращивает функциональность, реализуя оставшиеся high-priority истории из бэклога. QA-инженеры (или все по очереди) проводят ручное тестирование UI/UX. Разработчики фиксируют баги, выявленные в интеграционных и ручных тестах. Улучшается пайплайн MLOps (логирование, мониторинг). Результат: Реализовано >80% функционала MVP, исправлены критические баги, обновлен пайплайн.	ЛР
7.	Финальная демонстрация.	Лаб 7 (Спринт 6): Финальная сборка, презентация и передача. Команда проводит финальное приемочное тестирование, собирает все компоненты в стабильную сборку. Готовит презентацию для партнера: постановка проблемы, архитектура, демонстрация работы, достигнутые метрики, план развития. Оформляет всю документацию и "инструкцию по запуску" проекта. Проводит итоговую ретроспективу. Результат: Готовый к демо MVP, презентация, полный пакет документации и кода, записка с выводами по проекту.	ЛР

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»	10
Итого			10

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 «Разработка гибридных интеллектуальных систем».

Оценочные средства включает контрольные материалы для проведения **текущего контроля** в форме открытой защиты командных проектов и ответов на вопросы на зачёте.

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	Процессы в IT.	УК-1.2; УК-3.2; УК-5.2; ПК-3.2; ПК-4.2; ПК-4.3; SS-2.1; SS-2.2; SS-3.2; LC-1.1; LC-4.1.1; LC-4.2.1	Лабораторная работа №1	Защита командных проектов Вопросы к зачету 1-3
2	Специфика управления ИИ проектом.	УК-1.2; УК-3.2; УК-5.2; ПК-3.2; ПК-4.2; ПК-4.3; SS-2.1; SS-2.2; SS-3.2; LC-1.1; LC-4.1.1; LC-4.2.1	Лабораторная работа №2	Вопросы к зачету 4-6, Защита командных проектов
3	Архитектура гибридных интеллектуальных систем.	УК-1.2; УК-3.2; УК-5.2; ПК-3.2; ПК-4.2; ПК-4.3; SS-2.1; SS-2.2; SS-3.2; LC-1.1; LC-4.1.1; LC-4.2.1	Лабораторная работа №3	Вопросы к зачету 7-9, Защита командных проектов
4	Инфраструктура и MLOps для быстрого и стабильного цикла.	УК-1.2; УК-3.2; УК-5.2; ПК-3.2; ПК-4.2; ПК-4.3; SS-2.1; SS-2.2; SS-3.2; LC-1.1; LC-4.1.1; LC-4.2.1	Лабораторная работа №4	Вопросы к зачету 10-12, Защита командных проектов
5	Паттерны интеграции современных ИИ-компонент: RAG и агенты	УК-1.2; УК-3.2; УК-5.2; ПК-3.2; ПК-4.2; ПК-4.3; SS-2.1; SS-2.2; SS-3.2; LC-1.1; LC-4.1.1; LC-4.2.1	Лабораторная работа №5	Вопросы к зачету 13-15, Защита командных проектов
6	Качество кода.	УК-1.2; УК-3.2; УК-5.2; ПК-3.2; ПК-4.2; ПК-4.3;	Лабораторная работа №6	Вопросы к зачету 16-18, Защита командных проектов

		SS-2.1; SS-2.2; SS-3.2; LC-1.1; LC-4.1.1; LC-4.2.1		
7	Финальная демонстрация.	УК-1.2; УК-3.2; УК-5.2; ПК-3.2; ПК-4.2; ПК-4.3; SS-2.1; SS-2.2; SS-3.2; LC-1.1; LC-4.1.1; LC-4.2.1	Лабораторная работа №7	Вопросы к зачету 19-21, Защита командных проектов

Показатели, критерии и шкала оценки сформированных компетенций

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

УК-1	<i>Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач</i>
УК-1.2	Выбирает оптимальный вариант решения задачи, аргументируя свой выбор Знать: Критерии сравнения альтернативных архитектурных и технологических решений. Уметь: Проводить сравнительный анализ вариантов реализации компонентов ИС. Владеть: Навыками аргументации технических решений в команде.
УК-3	<i>Способен осуществлять социальное взаимодействие и реализовывать свою роль в команде</i>
УК-3.2	Применяет методы командного взаимодействия; планирует и организует командную работу Знать: Методы и инструменты организации командной работы в процессе разработки ИС. Уметь: Распределять разработческие задачи и контролировать их выполнение. Владеть: Техниками командной координации и планирования в гибкой методологии.
УК-5	<i>Способен воспринимать межкультурное разнообразие общества в социально-историческом, этическом и философском контекстах</i>
УК-5.2	Интерпретирует проблемы современности с позиции этики и философских знаний Знать: Современные социально-этические проблемы в сфере информационных технологий. Уметь: Применять этические frameworks для анализа процесса разработки и результатов. Владеть: Навыками этической оценки создаваемых информационных систем.
ПК-3	<i>Способен применять основные алгоритмические и программные решения в области информационно-коммуникационных технологий, а также участвовать в их разработке</i>

ПК-3.2	<p>Определяет элементы проблемной области и их взаимодействие, архитектуру программной системы, ее функциональные возможности и логику работы с использованием основных концептуальных положений функционального, логического, объектно-ориентированного и визуального направлений программирования</p>
	<p>Знать: Принципы проектирования архитектуры ПО для масштабируемых ИС. Уметь: Проектировать компоненты системы и их взаимодействие в командном проекте. Владеть: Навыками проектирования программной архитектуры с учетом разделения ответственности.</p>
ПК-4	<i>Способен планировать необходимые ресурсы и этапы выполнения работ в области информационно-коммуникационных технологий, составлять соответствующие технические описания и инструкции</i>
ПК-4.2	<p>Применяет современные приемы работы с инструментальными средствами, поддерживающими создание программных продуктов и программных комплексов на базе языков программирования, баз данных и пакетов прикладных программ</p>
	<p>Знать: Приемы работы с инструментальными средствами в процессе командной разработки. Уметь: Использовать инструменты для создания и интеграции программных комплексов. Владеть: Техниками интеграции различных инструментов в единый контур разработки.</p>
ПК-4.3	<p>Способен использовать методы эффективного управления командой при разработке, внедрении и сопровождении программных продуктов</p>
	<p>Знать: Методы управления командой разработчиков в IT-проектах. Уметь: Организовывать работу команды на разных этапах жизненного цикла ИС. Владеть: Практиками управления распределенной командой разработки.</p>
SS-2	<i>Способен осуществлять свою трудовую деятельность с учётом необходимости эффективной коммуникации и взаимодействия в рамках коллективной проектной работы в сфере ИИ</i>
SS-2.1	<p>Эффективно коммуницирует с участниками проектной команды при планировании, реализации и анализе результатов работы</p>
	<p>Знать: Принципы эффективной коммуникации в проектах по разработке ИС. Уметь: Ясно излагать технические идеи и задачи команде. Владеть: Навыками проведения эффективных проектных встреч.</p>
SS-2.2	<p>Учитывает профессиональные и ролевые особенности коллег при совместной разработке технических решений и представлении результатов</p>
	<p>Знать: Особенности ролей в проектной команде по разработке ИС. Уметь: Адаптировать стиль коммуникации под техническую и нетехническую аудиторию. Владеть: Методами междисциплинарного взаимодействия в проекте ИС.</p>
SS-3	<i>Способен осуществлять свою трудовую функцию с учетом неопределенности как сущностной черты функционирования искусственного интеллекта</i>
SS-3.2	<p>Определяет релевантность применения ИИ для решения конкретных задач, анализирует поведение ИИ в техническом, социальном и правовом контекстах, переносит идеи и методы за пределы исходной предметной области</p>

	<p>Знать: Критерии применимости ИИ для решения задач в контексте разрабатываемой ИС.</p> <p>Уметь: Анализировать поведение ИИ как компонента ИС в различных контекстах.</p> <p>Владеть: Методами анализа границ применимости моделей в проектируемой системе.</p>
LC-1	<i>Способен проводить анализ бизнес-проблем с оценкой перспективности применения ИИ для их решения, осуществлять постановку задачи машинного обучения, формулировать требования к системе ИИ</i>
LC-1.1	Оценивает технические требования и разрабатывает техническое задание на системы искусственного интеллекта в конкретной предметной области
	<p>Знать: Структуру и содержание ТЗ на систему ИИ, включая разделы по взаимодействию с другими компонентами и командой.</p> <p>Уметь: Формализовывать бизнес-требования в технические спецификации, понятные для всех участников команды разработки.</p> <p>Владеть: Методами совместной разработки и согласования технического задания.</p>
LC-4.1	<i>Способен управлять процессом жизненного цикла ИИ-продукта</i>
LC-4.1.1	Осуществляет запуск и ведение проекта в области ИИ, в том числе планирование и контроль задач, оценку ресурсов
	<p>Знать: Основы проектного управления в области ИИ и инструменты для коллективного планирования.</p> <p>Уметь: Осуществлять ведение (запуск и управление) проектов в области ИИ, в том числе подбор команды, планирование и контроль задач, оценка ресурсов.</p> <p>Владеть: Инструментами планирования, контроля и координации командной работы в проекте.</p>
LC-4.2	<i>Способен руководить работой команды проекта в области ИИ</i>
LC-4.2.1	Координирует и контролирует работу команд проекта с целью достижения общих целей проекта
	<p>Знать: Методы и инструменты координации командной работы в проектах, включая практики Agile.</p> <p>Уметь: Обеспечивать взаимодействие всех участников проекта, включая бизнес-лидеров, команду Data Science и разработчиков ПО.</p> <p>Владеть: Техниками контроля сроков, качества выполнения работ и синхронизации усилий команды.</p>

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

ТРЕБОВАНИЯ К КОМАНДНОМУ ПРОЕКТУ

Дисциплина: «Разработка гибридных интеллектуальных систем»

Формат: Рабочий MVP (Minimum Viable Product), разработанный командой из 15 человек.

Срок: 7 учебных недель (7 спринтов).

1. ЦЕЛЬ ПРОЕКТА

Продемонстрировать на реальном кейсе от пром-партнёра способность команды организовать полный цикл разработки гибридной информационной системы с интеграцией современных ИИ-компонентов (RAG, агенты, ML-сервисы), используя гибкие методологии (Scrum/KanBan) и промышленные практики DevOps/MLOps.

2. КРИТЕРИИ ПРИЕМКИ (DONE-КРИТЕРИИ)

Проект считается успешно завершённым, если команда предоставила:

- рабочий продукт, то есть развёрнутое и доступное для демонстрации веб-приложение (frontend + backend), телеграм бот и/или мобильное приложение, решающие ключевую задачу пром-партнёра;
- интегрированный ИИ, то есть как минимум один корректно работающий ИИ-компонент, реализованный по одному из паттернов (RAG, сервис предсказаний, интеллектуальный агент и тд), с обоснованием выбора модели;
- исходный код и инфраструктуру, то есть весь код размещён на Git-репозиториях ВУЗа, проект можно собрать и развернуть с помощью Docker-контейнеров;
- результаты проверки качества, то есть бэкенд-сервисы покрыты интеграционными тестами, фронтенд и мобильное приложение прошли как минимум ручное тестирование по чек-листу, составленному командой, или реализованы автотесты, проведено smoke-тестирование всего сценария;
- полную документацию, то есть полный пакет документов, включающий техническое задание, описание архитектуры, инструкцию по запуску и пользовательскую документацию.

3. СТРУКТУРА И СОДЕРЖАНИЕ ПРОЕКТА (ПО ЭТАПАМ)

Этап 1: Инициация и планирование (Спринт 1-2)

- **Product Backlog:** Сформированный в Jira/YouTrack/Linear, содержащий пользовательские истории (User Stories) в формате «As a [роль], I want [цель], so that [причина]» с критериями приемки (Acceptance Criteria). История должна иметь оценку в story points.
- **Техническое задание (ТЗ):** Документ, описывающий бизнес-требования, цели системы, основных пользователей и их сценарии (Use Case Diagram/User Story Map). Должна быть выделена отдельная секция «**Требования к ИИ-компоненте**» с описанием задачи, метрик качества (business, ML), требований к входным/выходным данным и формату API.
- **Архитектурное решение:** Диаграмма (C4 уровень 1-2 или аналог), отображающая высокоуровневые компоненты системы (клиенты, шлюз, микросервисы, БД, ML-сервис, брокеры) и ключевые взаимодействия между ними. **Обязательно:** отдельная схема пайплайна обработки данных и инференса модели.

Этап 2: Разработка и исследование (Спринт 2-5)

- **Исследовательский прототип ИИ:** Jupyter-ноутбук или Python-скрипт, демонстрирующий полный цикл работы с данными для ИИ-компоненты: предобработка, обучение/настройка нескольких моделей (включая baseline), их оценка по выбранным метрикам. Код должен быть воспроизводим.
- **Продакшен-код ИИ:** Код ML-сервиса (например, на FastAPI), инкапсулирующий логику инференса лучшей модели. Сервис должен иметь health-check, логирование, документацию API (Swagger/OpenAPI).
- **Исходный код системы:** Структурированные репозитории с кодом frontend (React/Vue/etc.), backend (Spring/Go/Python), мобильного приложения (Kotlin), конфигурациями для Docker и оркестратора (docker-compose, Kubernetes manifests).
- **Инфраструктура как код (IaC):** Dockerfile для каждого сервиса, docker-compose.yml для локального запуска всей системы. Приветствуется настройка Kubernetes (если позволяет инфраструктура ВУЗа).

Этап 3: Интеграция и тестирование (Спринт 4-6)

- **CI/CD конфигурация:** Файлы конфигурации для GitHub Actions/GitLab CI/Jenkins, обеспечивающие автоматическую сборку, прогон unit- и интеграционных тестов для backend и ML-сервиса.

- **Набор тестов:** Автоматические интеграционные тесты, проверяющие работу ключевых API endpoints и их взаимодействие с БД. Чек-лист для ручного тестирования UI/UX, заполненный результатами.
- **Система в сборе:** В конце каждого спринта система должна собираться целиком и демонстрировать инкрементальный прогресс (например, в спринте 3 работает один сценарий, в спринте 5 — все основные).

Этап 4: Финальная сборка и документирование (Спринт 7)

- **Финальная версия MVP:** Стабильная сборка всех компонентов, развёрнутая на тестовом стенде (ВУЗ или облако партнёра).
- **Пакет документации:**
 - README.md в корне репозитория: общее описание, инструкция по быстрому запуску.
 - ARCHITECTURE.md с детальным описанием принятых решений (ADR - Architecture Decision Record).
 - API.md или доступный Swagger UI.
 - Пользовательская инструкция (скриншоты, шаги).
 - Презентация проекта (отдельные требования см. ниже).
- **Видео-демо:** Короткое (3-5 мин.) записанное видео, демонстрирующее работу системы от лица конечного пользователя.

4. КРИТЕРИИ ОЦЕНКИ ПРОЕКТА

1. **Реализация бизнес-логики (25%):** Функциональная полнота, соответствие ТЗ, работающий end-to-end сценарий.
2. **Качество ИИ-компоненты (25%):** Корректность архитектурного выбора, обоснованность выбора модели, качество её интеграции, соблюдение MLOps практик.
3. **Инженерное качество (20%):** Чистота кода, качество тестов, надёжность инфраструктуры (сборка/деплой), работающий CI/CD.
4. **Организация работы команды (15%):** Ведение бэклога, проведение регулярных событий (стендапы, планирование, ретро), распределение ролей и вовлечённость участников (по данным из инструментов и отчетам).
5. **Документация и презентация (15%):** Полнота, ясность и структурированность документации; качество и убедительность финальной презентации и демо

ТРЕБОВАНИЯ К ФИНАЛЬНОЙ ПРЕЗЕНТАЦИИ ПРОЕКТА

Хронометраж: Строго 20 минут (15 мин. на выступление + 5 мин. на вопросы).

Цель: Продемонстрировать партнёру и комиссии ценность созданного решения, глубину проработки и профессиональность команды.

СТРУКТУРА И СОДЕРЖАНИЕ ПРЕЗЕНТАЦИИ:

Слайд 1. Титульный. Название проекта, команда, пром-партнёр, логотипы. (30 сек.)

Слайд 2. Проблема и Цель. *Что мы решали?* Кратко и ясно о бизнес-проблеме партнёра. Конкретная цель проекта в 1-2 предложениях. Ключевые метрики успеха (например, "сократить время обработки запроса с 30 мин. до 5", "повысить точность классификации до 95%"). (1.5 мин.)

Слайд 3. Команда и процесс. Как мы работали?* Схематично показать распределение ролей (Scrum-мастер, аналитики, dev, ML, MLOps, QA). Назвать использованную методологию (Scrum/KanBan) и ключевые инструменты (Jira, Git, Docker, Kubernetes). **Обязательно показать график бурн-дауна/апа или доску KanBan на финальной стадии, чтобы визуализировать прогресс по спринтам. (2 мин.)

**Слайд 4. Архитектурное решение. Что мы построили?* Показать итоговую архитектурную диаграмму (C4, контейнеры). Кратко пояснить ответственность каждого ключевого компонента (клиент, API Gateway, Backend Service, ML Service, Vector DB и т.д.). Сделать акцент на месте и роли ИИ-компоненты в общей системе. (2.5 мин.)

**Слайд 5. Фокус на ИИ: исследование и реализация. Как мы выбрали и внедрили "интеллект"? Ключевой слайд.*

* Постановка задачи (классификация, RAG, генерация).

* Какие данные и как готовили.

* Какие модели/архитектуры сравнивали (таблица с метриками: точность, скорость, размер). *Почему выбрали именно эту?*

* Паттерн интеграции: RAG (схема: документы -> эмбединг -> векторная БД -> запрос -> LLM), агент (схема: планирование -> исполнение инструментов), или ML-сервис.

* Как обеспечивали качество и observability (логи, метрики). (3.5 мин.)

****Слайд 6. DevOps/MLOps практики. Как мы обеспечивали качество и надёжность?*** Показать схему CI/CD пайплайна для backend и ML-модели (версионирование кода/данных/моделей, автотесты, автоматический деплой). Упомянуть инструменты (Docker, GitHub Actions, MLflow). Показать, как настроен мониторинг. (1.5 мин.)

**Слайд 7. ЖИВОЕ ДЕМО (или записанное видео). Что получилось в итоге? Сердце*

презентации. Чёткий, заранее отрепетированный сценарий от лица пользователя.

Показать решение той самой бизнес-проблемы со слайда 2. Акцентировать, как в интерфейсе проявляется работа ИИ-компоненты (например, "здесь система, используя RAG, находит в документах вот этот релевантный фрагмент"). (3 мин.)

**Слайд 8. Итоги и дальнейшее развитие. Что дальше?*

* Краткие результаты: достигли ли заявленных метрик? Что сделано?

* Трудности и как их преодолели (технические, организационные).

* Roadmap на будущее: какие функции/улучшения планируются в гипотетическом следующем релизе (2-3 пункта). (1.5 мин.)

Слайд 9. Спасибо за внимание! Вопросы. Контакты команды. (30 сек.)

Зачетно-экзаменационные материалы для промежуточной аттестации (зачет)

Вопросы для подготовки к зачету

1. Объясните на примере, как регулярные короткие совещания (стендапы) в гибких методологиях помогают команде управлять рисками и блокировками. Какую типовую ошибку на таких совещаниях важно избегать?

2. В чем заключается принципиальная разница между ролью Scrum-мастера и традиционного менеджера проекта? Какая из этих ролей фокусируется на устранении препятствий для работы команды?

3. Опишите типичные негативные последствия для проекта и команды, когда один участник берет на себя роль «героя-одиночки», пытаясь решить все ключевые задачи самостоятельно.

4. Что подразумевается под «двойной петлей» разработки в проектах с машинным обучением и почему это требует особого подхода к планированию?

5. Зачем при постановке задачи на ML-компонент необходимо разделять бизнес-метрики (например, конверсия) и инженерно-математические метрики (например, F1-мера)? Приведите пример такого разделения для гипотетической задачи.

6. Почему создание простого базового решения (baseline) является критически важным шагом перед разработкой сложной интеллектуальной модели? Как результаты baseline влияют на дальнейшие решения по проекту?

7. Обоснуйте, в каких случаях при разработке системы с несколькими клиентами (веб, мобильное приложение) целесообразно использовать паттерн Backend-for-Frontend (BFF) вместо единого API.

8. Какие преимущества и недостатки имеет вынос ML-модели в отдельный микросервис по сравнению с её интеграцией в код основного backend-приложения?

9. Что такое «холодный старт» ML-сервиса и какие архитектурные или организационные меры помогают минимизировать его влияние на пользовательский опыт?

10. Объясните важность сквозного версионирования (код, данные, конфигурации, модели) для воспроизводимости экспериментов и стабильности работы ML-системы в продакшене.

11. Чем принципиально отличается эталонный CI/CD пайплайн для ML-сервиса от пайплайна для обычного приложения? Назовите как минимум два специфичных для ML этапа.

12. Опишите гипотетический процесс, позволяющий автоматически обнаружить снижение качества работы модели в эксплуатации (концептуальный дрейф) и инициировать её переобучение.

13. Опишите пошаговую схему работы RAG-системы. На каком из этапов, по вашему мнению, возникает наибольшее количество потенциальных проблем с качеством итогового ответа?

14. В чем ключевое концептуальное отличие интеллектуального агента, способного к планированию и использованию инструментов, от модели, работающей в режиме простого «вопрос-ответ»?

15. На какие ключевые характеристики, помимо производительности, нужно обращать внимание при выборе модели для получения текстовых эмбеддингов в RAG-системе?

16. Что такое контрактное тестирование и как оно помогает поддерживать стабильность взаимодействия между независимыми сервисами, особенно когда один из них часто обновляется?

17. Назовите две специфические угрозы безопасности, связанные непосредственно с эксплуатацией ML-моделей, и предложите базовые методы защиты от них.

18. Для чего фиксируют Архитектурные Решения (ADR)? Какой минимальный набор информации должна содержать такая запись, чтобы быть полезной в будущем?

19. При подготовке финальной демонстрации проекта для заказчика всегда есть выбор между демонстрацией идеально работающего узкого сценария и более сырой, но полной версии. Какой подход вы считаете более правильным для MVP и почему?

20. Если в ходе ретроспективы команда пришла к выводу о хроническом недостатке обратной связи от условного заказчика, какие конкретные изменения в процессах взаимодействия можно предложить для следующего проекта?

21. По каким критериям и на основании каких данных принимается решение о том, что успешно запущенный MVP требует не точечных улучшений, а перехода к разработке масштабной новой версии (v2.0)?

Темы проектов формируются из кейсов, предоставленных промышленными партнёрами, список кейсов может быть изменён или дополнен к моменту проведения дисциплины. Существующий список кейсов приведён ниже.

Кейсы ПАО «Сбербанк»

1. Генеративный ИИ для автоматического составления инвестиционных обзоров

Описание:

Аналитики Сбера ежедневно составляют десятки аналитических и инвестиционных обзоров по рынкам, компаниям, макроэкономике. Задача — исследовать применение LLM для генерации кратких сводок и аналитических отчетов на основе входных данных: биржевые котировки, макроэкономические показатели, рыночные события.

Цель:

Разработать инструмент, способный по структурированным данным и краткому описанию формировать инвестиционный обзор в деловом стиле.

Ожидаемый результат:

Модель, генерирующая аналитические тексты длиной 500–1000 слов с разделами «обзор событий», «рекомендации», «прогнозы», оформленные в формате банка.

2. НЛП-анализ жалоб клиентов в свободной форме**Описание:**

В рамках клиентского сервиса Сбербанк обрабатывает обращения из чатов, мобильного приложения и жалобной формы. Требуется построить модель семантического анализа, выделяющую суть обращения, определяющую тональность и потенциальную серьёзность инцидента.

Цель:

Автоматизировать классификацию обращений для ускорения маршрутизации и выявления повторяющихся болевых точек в продуктах и процессах.

Ожидаемый результат:

Прототип модели, автоматически выделяющей темы жалоб (например, «ошибка в приложении», «двойное списание»), их эмоциональную окраску и критичность.

3. Генерация сценариев фишинговых писем для обучения сотрудников**Описание:**

Банк проводит киберучения, включая рассылку тестовых фишинговых писем сотрудникам для повышения их устойчивости к социальным атакам. Проект предполагает использование генеративной модели для создания реалистичных фишинговых писем различных типов (поддельные счета, HR-запросы, ИТ-поддержка).

Цель:

Создать генератор, способный на основе заданных параметров (тема, стиль, уровень угрозы) создавать тексты фишинга для тренировок.

Ожидаемый результат:

Набор разнообразных примеров фишинга и оценка их эффективности по реакции сотрудников, а также классификация моделей угроз.

4. Мультиmodalный ассистент для банковских отделений**Описание:**

Физические отделения Сбербанка внедряют интерактивных консультантов.

Предполагается создание мультиmodalного ИИ-ассистента, который воспринимает речь и визуально ориентируется в пространстве (распознаёт клиента, документы, банкоматы), а также отвечает голосом.

Цель:

Разработать базовый прототип, имитирующий функциональность помощника: ответы на типовые запросы, визуальные подсказки, навигация по отделению.

Ожидаемый результат:

Интерактивная модель, объединяющая голосовой ввод, зрительное восприятие (например, QR-код паспорта), текстовый вывод и жестовую реакцию.

5. Объяснимость и контроль генеративных моделей в банковском ИИ**Описание:**

Банк активно использует LLM и NLP-сервисы (в чат-ботах, генерации шаблонов ответов, автоответах на e-mail), однако встает вопрос: как объяснять и контролировать поведение таких моделей, особенно в юридически значимых коммуникациях?

Цель:

Исследовать подходы к трассировке решений LLM (например, через логирование reasoning chain, пост-фильтрацию ответов, встроенные правила).

Ожидаемый результат:

Концепция системы explainability + compliance-модуля, обеспечивающего соответствие генерации стандартам банка и регулятора.

6. Генерация пользовательских сценариев работы в мобильном приложении**Описание:**

Банк хочет использовать генеративный ИИ для быстрой симуляции пользовательских сценариев — например, как клиент оформляет вклад, переводит средства, получает уведомление о риске мошенничества.

Цель:

Разработать генератор пошаговых сценариев пользовательского поведения с вариативностью (молодой клиент, пенсионер, ИП).

Ожидаемый результат:

Набор автоматически сгенерированных UX-сценариев, оформленных в виде сценариев для QA или UX-исследований, с логикой действий и типичными ошибками пользователя.

7. Генерация synthetic data для банковских моделей**Описание:**

Модели в Сбере требуют большого объёма транзакционных и клиентских данных, которые нельзя использовать напрямую из-за требований ЦБ и ФЗ-152. Задача — разработать метод генерации синтетических банковских данных, максимально близких к реальным по распределениям и поведению.

Цель:

Создать безопасный pipeline генерации данных (например, транзакций, профилей клиентов, шаблонов расходов) для обучения моделей.

Ожидаемый результат:

Синтетический датасет и отчет о метриках приближённости к реальному (TSNE, K-L divergence и др.), с оценкой пригодности для обучения скоринговых или антифрод-моделей.

8. Модель анализа инвестиционной привлекательности малого бизнеса**Описание:**

Банк активно развивает кредитование и инвестиционные инструменты для малого и среднего предпринимательства (МСП). Требуется создать модель, которая на основе открытых и банковских данных (выручка, расходы, тип деятельности, отзывы, онлайн-активность) оценивает инвестиционную привлекательность МСП.

Цель:

Разработать систему рейтинговой оценки компаний малого бизнеса с возможностью визуализации факторов и динамики показателей.

Ожидаемый результат:

Модель, присваивающая компании инвестиционный рейтинг (например, А–Е), объясняющая ключевые параметры и дающая рекомендации для инвестора.

9. Индивидуальная оценка кредитоспособности клиента на основе поведенческих данных**Описание:**

Современный кредитный скоринг выходит за рамки финансовых данных. Необходимо исследовать, как поведенческие и цифровые следы (частота входа в мобильный банк, способы оплаты, география, время отклика) влияют на персональную оценку риска.

Цель:

Разработать ML-модель, оценивающую вероятность дефолта по нестандартным поведенческим признакам (возможно — с explainable AI).

Ожидаемый результат:

Прототип скоринговой модели, которая, помимо стандартных данных, учитывает цифровой профиль клиента и объясняет решения (SHAP, LIME и др.).

10. Предиктивная аналитика возврата инвестиций по инфраструктурным проектам

Описание:

В ряде случаев Сбербанк выступает участником/инвестором в региональных инфраструктурных проектах (жилые массивы, дороги, технопарки). Задача — оценить прогнозируемую эффективность вложений с учётом демографии, миграции, экономической активности.

Цель:

Разработать модель, прогнозирующую ROI на горизонте 3–5 лет, используя внешние источники данных: Росстат, ЕГРЮЛ, кадастр, соцмедиа.

Ожидаемый результат:

Аналитическая модель с возможностью геовизуализации и сценарного анализа (рост/спад, госпрограммы, смена трафика и т.п.).

11. Анализ поведения пользователей в экосистеме цифрового рубля

Описание:

Сбербанк участвует в пилотных проектах по внедрению цифрового рубля. Интерес представляет исследование пользовательских паттернов: как изменяются модели потребления, скорости операций, уровень доверия, сравнение с классическим безналом.

Цель:

Построить модель анализа поведения клиентов, участвующих в транзакциях с цифровым рублем: частота, средний чек, контексты.

Ожидаемый результат:

Отчёт и ML-модель, классифицирующая типы пользователей и выявляющая ключевые различия в предпочтениях и барьерах цифровой валюты.

12. Сравнение text2video / text2img моделей

Описание:

Сбербанк заинтересован в сравнении text2video / text2img моделей (открытые модели, особенно китайские). Задача требует применения облачных ресурсов партнера для машинного обучения. От студентов требуется навык запуска открытых моделей, планирования, структурирования и логирования экспериментов, совместной работы. Задача может быть распараллелена для сравнения множества моделей независимо в группе студентов.

Цель:

Провести сравнение работы актуальных открытых моделей text2video / text2img.

Ожидаемый результат:

Таблица с результатами экспериментов модель / репозиторий / функционал / требования / оценка производительности / X примеров генераций (было/стало), human_eval по принципу арены (какая лучше)

Кейсы от «АВАЛАБ»

1. LLM и RAG для BI-системы Fastboard

Описание:

Для разрабатываемой компанией BI-системы Fastboard требуется разработать интерфейс на естественном языке для построения отчетов на больших массивах данных в ClickHouse. С помощью LLM необходимо классифицировать запросы пользователей на естественном языке и извлекать фактические параметры для дальнейшего вызова веб-сервиса отчетов.

Цель:

Разработать промпты для классификации и обработки запросов пользователей LLM и преобразования их к вызовам типовых отчетов с фактическими параметрами, извлекаемыми из запроса.

Ожидаемый результат:

Инструмент на основе LLM, позволяющий запрашивать данные о продажах.

2. Анализ обращений клиентов и CRM-переписки

Описание:

В службе клиентского сервиса застройщика ежедневно обрабатываются десятки обращений (e-mail, звонки, мессенджеры). Требуется реализовать систему семантического анализа и классификации NLU: выявлять суть обращений, уровень удовлетворенности, отслеживать повторяющиеся запросы.

Цель:

Автоматизировать первичный разбор и маршрутизацию запросов по тематике (сдача объекта, отделка, документы, жалоба и т.д.).

Ожидаемый результат:

Прототип, который выделяет суть обращений и формирует дашборд по текущим «болям» клиентов.

3. Генеративный ИИ для создания проектной документации по ТЗ

Описание:

В рамках проектирования объектов девелоперской компании архитекторы и инженеры тратят значительное время на подготовку текстовой проектной документации (обоснование решений, пояснительные записки, описания инженерных систем). Задача — исследовать возможность использования LLM для генерации черновиков проектной документации на основе исходных данных: этажность, материалы, климат, назначение, нормы.

Цель:

Разработать прототип текстового генератора, который помогает специалистам быстрее формировать документацию в соответствии с шаблонами и нормативами.

Ожидаемый результат:

Инструмент на основе LLM, создающий логически стройный и нормативно грамотный текст, поддающийся быстрой редакции инженером.

4. Мультимодальный агент для анализа строительных площадок

Описание:

ООО «АВА ЛАБ» разрабатывает систему для мониторинга строительных объектов. Требуется создать прототип мультимодального ИИ-агента, способного анализировать изображения со стройплощадки (видео/фото), а также принимать голосовые и текстовые запросы (например, «проверь монтаж перекрытия на 5 этаже»).

Цель:

Объединить возможности компьютерного зрения (распознавание стадии строительства, техники, нарушений) и НЛП (понимание запросов, отчетов).

Ожидаемый результат:

Интерактивный агент, который на запрос специалиста может показать нужный участок, прокомментировать прогресс, зафиксировать нарушения.

4. Генерация рекламного контента для жилых комплексов

Описание:

«АВА ГРУПП» регулярно запускает маркетинговые кампании для жилых комплексов. Необходимо исследовать использование диффузионных моделей для генерации изображений (визуализации интерьеров, окрестностей, видов из окон) и LLM — для описаний квартир, преимуществ района, инфраструктуры.

Цель:

Создать инструменты для быстрой генерации продающих материалов без привлечения дизайнеров и копирайтеров на первых этапах.

Ожидаемый результат:

Набор сгенерированных карточек объектов с текстом, изображением и логикой «живого» рекламного сообщения.

6. Генерация документации и шаблонов договоров

Описание:

Юридический департамент регулярно работает с договорами долевого участия, актами приёма-передачи и другими документами. Использование LLM может значительно сократить время на подготовку черновиков — достаточно ввести параметры сделки.

Цель:

Создать систему, которая генерирует адаптированные тексты документов по вводным данным (тип объекта, этаж, площадь, ФИО, сроки и пр.).

Ожидаемый результат:

Генератор документов в формате Word или PDF с автоматической подстановкой параметров и соблюдением юридического стиля.

7. Модель прогнозирования сроков сдачи объектов на основе текстовых и визуальных данных**Описание:**

Девелоперская компания ведёт аналитический архив по срокам строительства. С помощью мультимодальных моделей (текстовые отчёты + фото стройки) можно прогнозировать вероятность отклонения от графика сдачи.

Цель:

Разработать модель, которая по текущему статусу объекта (фото, отчёт СМР) оценивает риски задержек.

Ожидаемый результат:

Прототип, который показывает вероятность отклонений и даёт текстовые пояснения (основанные на распознанных признаках — «не завершены фасадные работы», «монтаж инженерии не начат»).

8. Обратная генерация — ИИ-помощник для покупателей квартир**Описание:**

Будущие покупатели часто задают типовые вопросы о квартирах, планировках, ипотеке, акциях, сроках. Вместо call-центра предлагается реализовать LLM-бота, который обрабатывает текстовые и голосовые запросы, показывает планировки, ссылается на PDF-документы и может «объяснять» информацию простым языком.

Цель:

Упростить коммуникацию с клиентами на этапе выбора квартиры и повысить качество первичного контакта.

Ожидаемый результат:

Демо-бот, способный отвечать на вопросы о жилом комплексе, ориентируясь в его характеристиках и маркетинговых документах.

Перечень компетенций (части компетенции), проверяемых оценочным средством

УК-1.2; УК-3.2; УК-5.2; ПК-3.2; ПК-4.2; ПК-4.3; SS-2.1; SS-2.2; SS-3.2; LC-1.1; LC-4.1.1; LC-4.2.1

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

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

Итоговой формой контроля сформированности компетенций у обучающихся по дисциплине является зачет. Студенты обязаны сдать зачет в соответствии с расписанием и учебным планом.

ФОС промежуточной аттестации состоит из вопросов к зачету и открытой защиты командных проектов.

Форма проведения зачета: устно.

Экзаменатору предоставляется право задавать студентам дополнительные вопросы по всей учебной программе дисциплины.

Результат сдачи зачета заносится преподавателем в экзаменационную ведомость и зачетную книжку.

Оценивание уровня освоения дисциплины основывается на качестве выполнения студентом заданий текущего контроля и ответов на вопросы зачета.

Критерии оценки:

Объявляется открытая защита проектов, на которой присутствуют представители пром-партнёров и преподаватели практики, а также желающие преподаватели центра искусственного интеллекта. По итогам 20-минутной презентации комиссия определяет защищен проект, или нет.

Далее каждый из преподавателей практиков разбирает со своей подгруппой итоговые результаты каждого из студентов.

Если команда защитила проект, то представители команды самостоятельно объявляют студентов, которые выполнили свои части проекта и выявляют студентов, никак не участвовавших в командной разработке. Студенты, участвовавшие в работе получают оценку «зачтено». Студенты, не участвовавшие в командной работе сдают зачёт отдельный день. Данным студентам ставится задача в рамках улучшения существующего проекта, к дню зачёта студент должен выполнить эту задачу и ответить на теоретический вопрос. Если студент сдаёт задачу и отвечает на теоретический вопрос, он получает оценку «зачтено». Если студент не делает задачу, или не отвечает на теоретический вопрос, он получает оценку «не зачтено».

Если команда не защитила проект, то преподаватель выявляет на основании документации проекта и инфраструктуры проекта студентов, недобросовестная работа которых повлекла за собой неудачную защиту проекта. Этим студентам ставится отдельная задача, не связанная с проектом в соответствии с занимаемыми ролями. К дню зачёта студент должен выполнить эту задачу и ответить на теоретический вопрос. Если студент сдаёт задачу и отвечает на теоретический вопрос, он получает оценку «зачтено». Если студент не делает задачу, или не отвечает на теоретический вопрос, он получает оценку «не зачтено». Студенты, которые выполнили свои части в рамках командной работы, готовятся только к теоретической части зачёта. Если такие студенты отвечают на теоретический вопрос, они получают оценку «зачтено», иначе – оценку «не зачтено».

Методические рекомендации, определяющие процедуры оценивания лабораторных работ:

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

По каждой лабораторной работе проводится совещание, предусмотренное выбранной методологией разработки. Каждый из студентов сдаёт выполненную работу в формате, определенном коллективом для его роли, например система задач в таск трекаре, документация в вики системе, ТЗ, код на гите, отчёт о проведенных тестах и тд.

При невыполнении задачи студентом преподаватель выявляет причины и ставит сроки завершения задачи, или в случае критичности распределяет задачу между другими членами команды.

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

Цель: Предоставить студенту мощности для разработки и развертывания Мл решений.

Задачи:

Предоставить гит ВУЗа под командный проект.

Предоставить виртуальные машины под инфраструктуру.

Предоставить мощности под задачи обучения моделей.

Предоставить мощности под развертывание системы документации и таск трекер.

4.4. Методические указания по организации лабораторных работ

Лабораторные работы – часть общего командного проекта, который ведется согласно выбранной гибкой методологии.

4.5. Методические указания по организации проектной деятельности студентов

1. Цель и задачи

Цель: Интегрировать полученные в ходе обучения в университете знания и навыки в рамках реализации сквозного проекта, максимально приближенного к реальным условиям промышленной разработки.

Задачи:

Сформировать опыт полного жизненного цикла разработки ПО в команде

Закрепить навыки проектного планирования, управления задачами и командной коммуникации

Развить умение принимать архитектурные и технологические решения

Отработать процедуры поставки и сопровождения программного продукта

2. Организация проектной деятельности

Продолжительность: 1 семестр (параллельно с лабораторными работами)

Формат: Сквозной проект с поэтапной сдачей результатов

Команды: 15 человек с распределением ролей

Процесс: Гибкая методология разработки (Scrum/Kanban)
Результат: Работающее приложение с полной инфраструктурой

3. Этапы выполнения проекта приведены выше в требованиях к проекту.

4. Роли в команде

Team Lead

Архитектор

Бизнес аналитик

Системный аналитик

МЛ исследователь

МЛ инженер

Главный разработчик

Разработчик бек

Разработчик фронт

Разработчик мобильного приложения/бота

Администратор-девопс

Администратор МЛопс

Тестировщик-бек

Тестировщик фронт

5. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля)

5.1 Основная литература:

1. Методы программирования : учебно-методическое пособие / авторы В. В. Подколзин, А. Н. Полетайкин, Е. П. Лукашик [и др.] ; Министерство науки и высшего образования Российской Федерации, Кубанский государственный университет. - Краснодар : Кубанский государственный университет, 2020. - 174 с.

2. Носова, Л. С. Case-технологии и язык UML [Электронный ресурс] : учебно-методическое пособие / Л. С. Носова. — 2-е изд. — Электрон. текстовые данные. — Челябинск, Саратов : Южно-Уральский институт управления и экономики, Ай Пи Эр Медиа, 2019. — 67 с. — 978-5-4486-0670-0. — Режим доступа: <http://www.iprbookshop.ru/81479.html>.

3. Доррер, Г. А. Методология программной инженерии : учебное пособие / Г. А. Доррер. — Красноярск : Сибирский государственный университет науки и технологий имени академика М.Ф. Решетнева, 2021. — 190 с.

4. Иванов, П. К. Глубокое обучение для NLP: от RNN до Transformer / П. К. Иванов, Е. А. Смирнова. – Санкт-Петербург : БХВ-Петербург, 2022. – 256 с. – ISBN 978-5-9775-0987-6.

5. Кузнецов, М. А. Языковые модели и их применение : учебник / М. А. Кузнецов. – Москва : Лань, 2023. – 415 с. – ISBN 978-5-8114-4567-8.

6. Петров, Д. С. NLP на Python: от классики до нейросетей / Д. С. Петров. – Москва : Питер, 2022. – 320 с. – ISBN 978-5-4461-2345-6.

7. Sun, X., Li, J., Kovalenko, A.V., Feng, W., Ou, Y. Integrating Reinforcement Learning and Learning From Demonstrations to Learn Nonprehensile Manipulation //IEEE Transactions on Automation Science and Engineering, 2023, 20(3), 1735–1744, DOI: 10.1109/TASE.2022.3185071, Q1

8. Petukhova, A.V.; Kovalenko, A.V.; Ovsyannikova, A.V. Algorithm for Optimization of Inverse Problem Modeling in Fuzzy Cognitive Maps. Mathematics 2022, 10, 3452. DOI: 10.3390/math10193452, Q1

9. Lu Y., Xu X., Wang L. Smart manufacturing process and system automation—a critical review of the standards and envisioned scenarios //Journal of Manufacturing Systems. – 2020. – Т. 56. – С. 312-325.

10. Ajiga D. et al. The role of software automation in improving industrial operations and efficiency //International Journal of Engineering Research Updates. – 2024. – Т. 7. – №. 1. – С. 22-35.

11. Laplante P. A., Kassab M. What every engineer should know about software engineering. – CRC Press, 2022.

12. Документация по библиотекам NLP:

– spaCy: <https://spacy.io/usage>

– Hugging Face Transformers: <https://huggingface.co/docs/transformers/index>

– NLTK: <https://www.nltk.org/>

5.2 Дополнительная литература:

1. Полетайкин, А. Н. Социальные и экономические информационные системы: законы функционирования и принципы построения : учеб. пособие / А. Н. Полетайкин ; Сиб. гос. ун-т телекоммуникаций и информатики. - Новосибирск : СибГУТИ, 2016. - 240 с. : ил.

2. Влацкая, И.В. Проектирование и реализация прикладного программного обеспечения : учебное пособие / И.В. Влацкая, Н.А. Заельская, Н.С. Надточий ; Кафедра компьютерной безопасности и математического обеспечения информационных систем, Министерство образования и науки Российской Федерации, Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Оренбургский государственный университет». - Оренбург : ОГУ, 2015. - 119 с. – http://biblioclub.ru/index.php?page=book_red&id=439107&sr=1

3. Леоненков, А.В. Язык UML в анализе и проектировании программных систем и бизнес-процессов. Лекция 1. Базовые принципы и понятия технологии разработки объектно-ориентированных информационных систем на основе UML 2. Презентация / А.В. Леоненков. - М. : Национальный Открытый Университет «ИНТУИТ», 2014. - 34 с.– http://biblioclub.ru/index.php?page=book_red&id=238441&sr=1

4. Мостовой Я.А. Управление программными проектами [Электронный ресурс]: учебное пособие/ Мостовой Я.А.— Электрон. текстовые данные.— Самара: Поволжский государственный университет телекоммуникаций и информатики, 2016.— 103 с.— Режим доступа: <http://www.iprbookshop.ru/71894.html>.— ЭБС «IPRbooks».

5. Громов, К. Л. Генеративные модели в NLP: GPT и не только / К. Л. Громов. – Москва : Интернет-Университет, 2022. – 275 с. – ISBN 978-5-9556-0123-4. Лебедев, С. А. Этика искусственного интеллекта: NLP и большие языковые модели / С. А. Лебедев. – Санкт-Петербург : Альфа-книга, 2023. – 150 с. – ISBN 978-5-98281-456-9.

6. Тихонов, Р. В. Оптимизация NLP-моделей: квантование, дистилляция / Р. В. Тихонов. – Москва : Техносфера, 2022. – 195 с. – ISBN 978-5-94836-678-1.

7. Polykovskiy, Daniil, et al. "Molecular sets (MOSES): a benchmarking platform for molecular generation models." *Frontiers in pharmacology* 11 (2020): 565644.

8. Khrabrov, Kuzma, et al. "\$\nabla^2\$ DFT: A Universal Quantum Chemistry Dataset of Drug-Like Molecules and a Benchmark for Neural Network Potentials." *Advances in Neural Information Processing Systems* 37 (2024): 36869-36889.

5.3 Периодические издания:

1. Базы данных компании «Ист Вью» <http://dlib.eastview.com>

2. Электронная библиотека GREBENNIKON.RU <https://grebennikon.ru/>

Конференции А*:

1. <https://openreview.net/forum?id=FMMF1a9ifL>
2. <https://openreview.net/forum?id=EIUrNM9U8c#discussion>
3. <https://openreview.net/forum?id=JoO6mtCLHD>
4. <https://aclanthology.org/2024.findings-emnlp.760/>
5. <https://aclanthology.org/2020.coling-main.588/>
6. https://link.springer.com/chapter/10.1007/978-3-030-72113-8_30
7. https://link.springer.com/chapter/10.1007/978-3-031-42448-9_10
8. <https://aclanthology.org/2024.findings-naacl.288/>

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.prlib.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>

Бесплатные образовательные ресурсы

1. Jupyter Notebook – интерактивные вычисления
2. Visual Studio Code – редактор кода с поддержкой Python
3. Google Scholar/arXiv – доступ к научным публикациям

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.uceba.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/](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/>

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

По курсу предусмотрено проведение лекционных занятий, на которых дается основной систематизированный материал по методологиям и инструментам коллективной разработки. В ходе лекций разбираются ключевые концепции: от классических (Waterfall) и гибких (Agile, Scrum, Kanban) методологий до современных практик DevOps и CI/CD. Особое внимание уделяется инструментальным средствам - системам контроля версий (Git), трекерам задач (Jira), платформам код-ревью (GitLab, GitHub), контейнеризации (Docker) и автоматизации развертывания. После каждой лекции рекомендуется выполнить практические задания по освоению рассмотренных инструментов и методик.

Лабораторные занятия курса посвящены практическому освоению инструментов коллективной разработки. На занятиях детально разбираются рабочие процессы: от организации репозитория и ветвления до настройки пайплайнов непрерывной интеграции и развертывания. Студенты работают в командах над общим проектом, применяя такие инструменты как Git, Docker, GitLab CI/CD, GitHub Actions. После каждого лабораторного занятия предлагаются задания для самостоятельного закрепления материала - отработка команд Git, модификация CI/CD конфигураций, настройка процессов код-ревью.

При самостоятельной работе студентам необходимо изучать рекомендованную литературу (учебники, документацию инструментов, лучшие практики) для глубокого понимания принципов коллективной разработки. Выполняя проектные задания, студент должен уметь: формулировать требования к проекту; организовывать рабочее пространство команды; настраивать процессы совместной работы; проектировать и реализовывать архитектуру приложения; внедрять автоматизацию сборки, тестирования и развертывания. Особое внимание уделяется навыкам разрешения конфликтов, проведения код-ревью и оптимизации процессов разработки.

Важнейшим компонентом курса является самостоятельная проектная работа, в ходе которой студенты в командах разрабатывают законченные информационные системы с полным жизненным циклом разработки. Такой проект позволяет закрепить навыки проектирования и реализации комплексных решений в условиях, максимально приближенных к промышленной разработке, включая планирование спринтов, распределение задач, код-ревью, непрерывную интеграцию и поставку.

Для студентов с ограниченными возможностями здоровья предусмотрены дополнительные индивидуальные консультации, на которых преподаватель детально разъясняет сложные аспекты дисциплины, помогает адаптировать задания и обеспечивает специальные условия для освоения практических навыков работы с текстовыми данными. Индивидуальный подход позволяет таким студентам полноценно участвовать в учебном процессе и достигать требуемых результатов обучения.

Кейсы ПАО «Сбербанк»

1. Генеративный ИИ для автоматического составления инвестиционных обзоров

Описание:

Аналитики Сбера ежедневно составляют десятки аналитических и инвестиционных обзоров по рынкам, компаниям, макроэкономике. Задача — исследовать применение LLM для генерации кратких сводок и аналитических отчетов на основе входных данных: биржевые котировки, макроэкономические показатели, рыночные события.

Цель:

Разработать инструмент, способный по структурированным данным и краткому описанию формировать инвестиционный обзор в деловом стиле.

Ожидаемый результат:

Модель, генерирующая аналитические тексты длиной 500–1000 слов с разделами «обзор событий», «рекомендации», «прогнозы», оформленные в формате банка.

2. Генерация сценариев фишинговых писем для обучения сотрудников

Описание:

Банк проводит киберучения, включая рассылку тестовых фишинговых писем сотрудникам для повышения их устойчивости к социальным атакам. Проект предполагает использование генеративной модели для создания реалистичных фишинговых писем различных типов (поддельные счета, HR-запросы, ИТ-поддержка).

Цель:

Создать генератор, способный на основе заданных параметров (тема, стиль, уровень угрозы) создавать тексты фишинга для тренировок.

Ожидаемый результат:

Набор разнообразных примеров фишинга и оценка их эффективности по реакции сотрудников, а также классификация моделей угроз.

Кейсы от «АВАЛАБ»

3. Генерация рекламного контента для жилых комплексов

Описание:

«АВА ГРУПП» регулярно запускает маркетинговые кампании для жилых комплексов. Необходимо исследовать использование диффузионных моделей для генерации изображений (визуализации интерьеров, окрестностей, видов из окон) и LLM — для описаний квартир, преимуществ района, инфраструктуры.

Цель:

Создать инструменты для быстрой генерации продающих материалов без привлечения дизайнеров и копирайтеров на первых этапах.

Ожидаемый результат:

Набор сгенерированных карточек объектов с текстом, изображением и логикой «живого» рекламного сообщения.

4. Генерация документации и шаблонов договоров

Описание:

Юридический департамент регулярно работает с договорами долевого участия, актами приёма-передачи и другими документами. Использование LLM может значительно сократить время на подготовку черновиков — достаточно ввести параметры сделки.

Цель:

Создать систему, которая генерирует адаптированные тексты документов по вводным данным (тип объекта, этаж, площадь, ФИО, сроки и пр.).

Ожидаемый результат:

Генератор документов в формате Word или PDF с автоматической подстановкой параметров и соблюдением юридического стиля.

5. Обратная генерация — ИИ-помощник для покупателей квартир

Описание:

Будущие покупатели часто задают типовые вопросы о квартирах, планировках, ипотеке, акциях, сроках. Вместо call-центра предлагается реализовать LLM-бота, который обрабатывает текстовые и голосовые запросы, показывает планировки, ссылается на PDF-документы и может «объяснять» информацию простым языком.

Цель:

Упростить коммуникацию с клиентами на этапе выбора квартиры и повысить качество первичного контакта.

Ожидаемый результат:

Демо-бот, способный отвечать на вопросы о жилом комплексе, ориентируясь в его характеристиках и маркетинговых документах.

7. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю)

7.1 Перечень информационно-коммуникационных технологий

1. Облачные платформы и сервисы

Google Colab – облачная среда для выполнения кода на Python с GPU/TPU

Kaggle – платформа для работы с датасетами и соревнований по ML

Hugging Face Spaces – развертывание NLP-моделей в виде демо

AWS/GCP/Azure/YandexCloud – облачные вычисления для обучения и инференса моделей

2. Системы управления версиями и коллаборации

Git/GitHub/GitLab – контроль версий кода и совместная разработка

Notion/Trello – организация проектной деятельности

3. Инструменты для работы с данными

Label Studio – разметка датасетов

DVC (Data Version Control) – управление версиями данных

4. Коммуникационные технологии

Telegram – координация работы в команде

МТС Линк – проведение онлайн-консультаций и защиты проектов

7.2 Перечень лицензионного и свободно распространяемого программного обеспечения

1. Лицензионное ПО

VSCoде – IDE для Python (свободнораспространяемое)

LibreOffice– оформление отчетов (свободнораспространяемое)

2. Свободное ПО (Open Source)

Библиотеки для NLP:

NLTK – обработка текста

sраСу – промышленные NLP-пайплайны

Hugging Face Transformers – предобученные модели (BERT, GPT)

Gensim – тематическое моделирование и word2vec

Фреймворки для ML:

PyTorch/TensorFlow – разработка нейросетей

scikit-learn – классические алгоритмы ML

Инструменты для визуализации:

Streamlit/Gradio – создание веб-интерфейсов для моделей

Matplotlib/Seaborn – графики и анализ данных

СУБД:

SQLite/PostgreSQL – хранение структурированных данных

FAISS/Annoy – векторный поиск

8. 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.	Практические занятия	Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения
4.	Групповые (индивидуальные) консультации	Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения, компьютерами, программным обеспечением
5.	Текущий контроль, промежуточная аттестация	Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения, компьютерами, программным обеспечением
6.	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.