

Приложение № 1
к приказу ректора
АНО ВО «Центральный университет»
от «28» мая 2026 г. № 0528.63

**Автономная некоммерческая организация высшего образования
«Центральный университет»**

УТВЕРЖДАЮ

Ректор АНО ВО «Центральный
университет»

Е.В. Ивашкевич

**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА –
ДОПОЛНИТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА ДЛЯ
ВЗРОСЛЫХ
«УПРАВЛЕНИЕ ИИ-ЦЕНТРИЧНОЙ РАЗРАБОТКОЙ ПРОГРАММНОГО
ОБЕСПЕЧЕНИЯ»**

Трудоемкость обучения: 80 ак. часов

Москва

2026

1. ОБЩАЯ ХАРАКТЕРИСТИКА ПРОГРАММЫ

1.1. Общие положения

Настоящая дополнительная общеобразовательная программа – дополнительная общеразвивающая программа для взрослых «Управление ИИ-центричной разработкой программного обеспечения» (далее – дополнительная общеразвивающая программа для взрослых) разработана на основании Федерального закона от 29 декабря 2012 г. № 273-ФЗ «Об образовании в Российской Федерации», приказа Минпросвещения России от 27 июля 2022 г. № 629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам» и приказа Минобрнауки России от 11 октября 2023 г. № 1678 «Об утверждении Правил применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ».

Дополнительная общеразвивающая программа для взрослых реализуется в Автономной некоммерческой организации высшего образования «Центральный университет» (далее – АНО ВО «Центральный университет», Центральный университет).

Разработчик программы: Генаров Виталий Андреевич, Генеральный директор ООО «ЭдЮнит», преподаватель АНО ВО «Центральный университет».

Дополнительная общеразвивающая программа для взрослых разработана в инициативном порядке.

Программа реализуется на русском языке.

1.2. Цель реализации программы

Дополнительная общеразвивающая программа для взрослых нацелена на удовлетворение индивидуальных потребностей обучающихся в интеллектуальном развитии; профессиональную ориентацию обучающихся.

Направленность программы: техническая.

Цель программы – формирование у обучающихся базового представления и начальных навыков управления полным циклом разработки программного обеспечения с использованием ИИ-инструментов и ИИ-агентов: от исследования потребностей пользователей и формулировки требований до координации ИИ-агентов, контроля качества, развёртывания и последующей доработки продукта.

1.3. Категории обучающихся

Основными категориями обучающихся, на которых рассчитана дополнительная общеразвивающая программа для взрослых, являются магистранты и участники акселерационной программы Центрального Университета «Ускоритель» с базовыми навыками в технической или продуктовой областях, имеющие опыт реализации ИТ-проекта в любой роли.

1.4. Идентификаторы достижения планируемых результатов обучения

Обучающийся, освоивший дополнительную общеразвивающую программу для взрослых, должен:

знать:

- жизненный цикл создания программного продукта, роли участников и базовую терминологию ИИ-центричной разработки;
- ограничения ИИ-агентов и языковых моделей: границы контекстного окна, склонность к ошибкам, ситуации, требующие обязательной проверки результата;
- риски безопасности при разработке с использованием ИИ: утечка данных, уязвимости сгенерированного кода, внедрение в запросы.

уметь:

- выдвигать гипотезы о потребностях пользователей, формулировать продуктовые гипотезы и оформлять требования в виде пользовательских сценариев и документации с помощью ИИ;
- настраивать контекст проекта с помощью ИИ — правила, инструкции для агентов, структуру документации — для повышения качества работы ИИ;
- управлять процессом разработки с помощью ИИ: планирование задач, проверка кода, тестирование, развёртывание;
- проводить проверку качества и безопасности кода, созданного ИИ, с помощью ИИ.

владеть:

- методами составления запросов для задач разработки: пошаговое рассуждение, декомпозиция, примеры на входе;
- инструментами координации ИИ-агентов и протоколом подключения моделей к инфраструктуре проекта;
- навыками управления продуктовым циклом в ИИ-центричной разработке — от идеи и спецификации до рабочего прототипа и его доработки.

1.5. Трудоёмкость программы

Нормативная трудоёмкость обучения по данной программе – 80 академических часов, включая все виды контактной и самостоятельной работы слушателя.

1.6. Форма и сроки обучения

Обучение по программе осуществляется в очной форме с применением электронного обучения.

Электронный документ

Минимальный срок обучения на программе составляет 10 недель.

1.7. Режим занятий

Для всех занятий академический час устанавливается продолжительностью 40 минут.

Учебная нагрузка устанавливается не более 8 академических часов в неделю, включая все виды контактной и самостоятельной учебной работы слушателя.

2. ПРОГРАММА УЧЕБНОГО КУРСА

2.1. Учебный план дополнительной общеразвивающей программы для взрослых «Управление ИИ-центричной разработкой программного обеспечения»

Продолжительность обучения – 80 ак. часов.

Форма обучения – очная.

№ п/п	Наименование раздела	Всего, академ. ч.	Контактная работа (академ. ч.)		Самост. работа (академ. ч.)	Формы аттестаци и контроля знаний
			лекции	семинары (практич. и лаборат. занятия)		
1.	Основы ИИ-центричной разработки	8	2	2	4	Домашнее задание
2.	Продуктовое исследование и анализ	16		8	8	Домашнее задание
3.	Архитектура и планирование	8	1	3	4	Домашнее задание
4.	Разработка и качество	24		12	12	Домашнее задание
5.	Развёртывание, тестирование и итерация	16		8	8	Домашнее задание
6.	«Результаты»	6	1	1	4	
	ВСЕГО:	78	4	34	40	
	Итоговая аттестация	2		2		Зачет
	ИТОГО:	80	4	36	40	

**2.2. Учебно-тематический дополнительной общеразвивающей
программы для взрослых «Управление ИИ-центричной
разработкой программного обеспечения»**

№ п/п	Наименование раздела и темы	Всего, академ. ч.	Контактная работа (академ. ч.)		Самост. работа (академ. ч.)	Формы аттестаци и контроля знаний
			лекции	семинары (практич. и лаборат. занятия)		
1.	Основы ИИ-центричной разработки	8	2	2	4	Домашнее задание
1.1.	Введение: парадигма, цикл, роли. Модели, контекстное окно, составление запросов. Обзор инструментов ИИ-разработки	4	1	1	2	
1.2.	Управление контекстом проекта: правила, инструкции для агентов, навыки, организация документации	4	1	1	2	
2.	Продуктовое исследование и анализ	16		8	8	Домашнее задание
2.1.	Продуктовое мышление: гипотезы о потребностях, синтетические персоны, уровни готовности продукта	4		2	2	
2.2.	Системный и бизнес-анализ. Пользовательские сценарии (user stories), задачи пользователей (JTBD). Моделирование заинтересованных сторон	4		2	2	
2.3.	Документация требований	4		2	2	
2.4.	Проектирование структуры данных	4		2	2	
3.	Архитектура и планирование	8	1	3	4	Домашнее задание

3.1.	Архитектурное проектирование. Паттерны и архитектура ИИ-сгенерированного кода. Принципы разработки: пять принципов ООП-проектирования (SOLID), «делай проще» (KISS), «не пиши лишнего» (YAGNI).	4	1	1	2	
3.2.	Планирование разработки: декомпозиция на задачи, приоритизация, контроль версий, подключение через протокол MCP (Model Context Protocol). Обзор инструментов.	4		2	2	
4.	Разработка и качество	24		12	12	Домашнее задание
4.1.	Разработка с ИИ. Часть 1: настройка среды, подключение контекста, первые задачи для агента	4		2	2	
4.2.	Документация: создание и генерация с помощью ИИ	4		2	2	
4.3.	Автоматическое тестирование с помощью ИИ	4		2	2	
4.4.	Разработка с ИИ. Часть 2: продолжение работы, рефакторинг (refactoring), работа с зависимостями	4		2	2	
4.5.	Отладка (debugging) ИИ-сгенерированного кода	4		2	2	
4.6.	Проверка кода (code review) с помощью ИИ	4		2	2	
5.	Развёртывание, тестирование и итерация	16		8	8	Домашнее задание

5.1.	Развёртывание (deployment)	4		2	2	
5.2.	Пользовательское тестирование (user testing)	4		2	2	
5.3.	Доработка пользовательских сценариев	4		2	2	
5.4.	Разработка с ИИ. Часть 3: реализация доработок, финализация инженерного прототипа	4		2	2	
6.	«Результаты»	6	1	1	4	
6.1.	Мониторинг (monitoring), аналитика, обратная связь. Итерация	6	1	1	4	
	ВСЕГО:	78	4	34	40	
	Итоговая аттестация	2		2		Зачет
	ИТОГО:	80	4	36	40	

2.3. Календарный учебный график

№ п/п	Наименование раздела учебного курса	Темы раздела	Акад ем. часов	Учебные недели									
				1	2	3	4	5	6	7	8	9	10
1.	Основы ИИ-центричной разработки	Введение: парадигма, цикл, роли. Модели, контекстное окно, составление запросов. Обзор инструментов ИИ-разработки	4	4									
		Управление контекстом проекта: правила, инструкции для агентов, навыки, организация документации	4	4									
2.	Продуктовое исследование и анализ	Продуктовое мышление: гипотезы о потребностях, синтетические персоны, уровни готовности продукта	4		4								
		Системный и бизнес-анализ. Пользовательские сценарии (user stories), задачи пользователей (JTBD). Моделирование заинтересованных сторон	4		4								
		Документация требований	4			4							
		Проектирование структуры данных	4			4							
3.	Архитектура и планирование	Архитектурное проектирование. Паттерны и архитектура ИИ-сгенерированного кода. Принципы разработки: пять принципов ООП-проектирования (SOLID), «делай проще» (KISS), «не	4				4						

		пиши лишнего» (YAGNI).											
		Планирование разработки: декомпозиция на задачи, приоритизация, контроль версий, подключение через MCP. Обзор инструментов	4				4						
4.	Разработка и качество	Разработка с ИИ. Часть 1: настройка среды, подключение контекста, первые задачи для агента	4					4					
		Документация: создание и генерация с помощью ИИ	4					4					
		Автоматическое тестирование с помощью ИИ	4						4				
		Разработка с ИИ. Часть 2: продолжение работы, рефакторинг (refactoring), работа с зависимостями	4						4				
		Отладка (debugging) ИИ-сгенерированного кода	4							4			
		Проверка кода (code review) с помощью ИИ	4							4			
5.	Развёртывание, тестирование и итерация	Развёртывание (deployment)	4								4		
		Пользовательское тестирование (user testing)	4								4		
		Доработка пользовательских сценариев	4									4	
		Разработка с ИИ. Часть 3: реализация доработок, финализация инженерного прототипа	4									4	

6.	«Результаты»	Мониторинг (monitoring), аналитика, обратная связь. Итерация	6										6
7.	Итоговая аттестация	Защита проектов	2										2

3. РАБОЧАЯ ПРОГРАММА КУРСА

3.1. Содержание курса

№ п/п	Наименование раздела учебного курса	Наименование темы	Содержание темы (тезисно)
1.	Основы ИИ-центричной разработки	Введение: парадигма, цикл, роли. Модели, контекстное окно, составление запросов. Обзор инструментов ИИ-разработки	1. Эволюция подходов к разработке: от ручного кодирования к ИИ-центричной разработке; новые роли — человек как продуктовый управляющий и руководитель агентов. 2. Жизненный цикл программного продукта: от исследования потребностей до мониторинга и итерации; ключевые этапы и роли на каждом. 3. Как работают большие языковые модели: контекстное окно, принципы составления запросов, выбор модели под задачу. 4. Обзор инструментов: среды разработки с ИИ (Cursor, Antigravity, VS Code), терминальные агенты, платформы развёртывания. 5. Выбор и утверждение индивидуального проекта на курс.
		Управление контекстом проекта: правила, инструкции для агентов, навыки, организация документации	1. Контекст как ключевой ресурс ИИ-центричной разработки: что входит в контекст, как его ограниченность влияет на качество результата. 2. Правила проекта и инструкции для агентов: как описать структуру, ограничения и стандарты проекта так, чтобы ИИ им следовал. 3. Навыки, агенты и ИИ-роли: как настраивать специализированные роли и навыки для агентов под задачи проекта. 4. Организация документации и хранения знаний: как структурировать информацию, чтобы не перегружать контекстное окно и давать агенту доступ к нужному.
2.	Продуктовое исследование и анализ	Продуктовое мышление: гипотезы о потребностях, синтетические персоны, уровни готовности продукта	1. Продуктовое мышление: начинать с проблемы, а не с решения; формулировка гипотез о потребностях пользователей. 2. Исследование с помощью ИИ: поиск информации агентами, анализ рынка, генерация и проверка гипотез. 3. Синтетические персоны: моделирование пользователей с помощью ИИ, проведение интервью с синтетическими персонами для быстрой валидации идей. 4. Уровни готовности продукта: прототип, инженерный прототип полного сценария, минимальный жизнеспособный продукт —

			разница между ними; цель курса — инженерный прототип, а не запуск продукта на рынок.
		Системный и бизнес-анализ. Пользовательские сценарии (user stories), задачи пользователей (JTBD). Моделирование заинтересованных сторон	1. Системный и бизнес-анализ с помощью ИИ: выявление заинтересованных сторон, ограничений, бизнес-требований. Моделирование заинтересованных сторон. 2. Формулировка пользовательских сценариев и задач пользователей: фреймворки «задачи, которые нужно выполнить» (JTBD) и пользовательские истории (user story). 3. Трансформация гипотез в структурированные требования: от свободных формулировок к формату, пригодному для работы ИИ-агентов.
		Документация требований	1. Форматы документации требований: от свободного описания к структурированному документу требований к продукту; что должен содержать и зачем. 2. Техническое задание как вход для ИИ-агента: как писать спецификации, которые агент сможет выполнить качественно; уровень детализации и однозначность формулировок. 3. Генерация и доработка документации требований с помощью ИИ: итеративный процесс от черновика до готового документа.
		Проектирование структуры данных	1. Сущности, атрибуты и связи: как определить, какие данные хранит приложение и как они связаны между собой. 2. Проектирование структуры данных с помощью ИИ: генерация схем на основе требований, проверка полноты и непротиворечивости. 3. Описание структуры данных как часть контекста для агента: как зафиксировать схему, чтобы ИИ корректно генерировал код работы с данными.
3.	Архитектура и планирование	Архитектурное проектирование. Паттерны и архитектура ИИ-сгенерированного кода. Принципы разработки: пять принципов ООП-проектирования (SOLID), «делай проще» (KISS), «не пиши лишнего» (YAGNI).	1. Базовая архитектура приложения: клиент, сервер, база данных, внешние сервисы; потоки данных и ответственность компонентов. 2. Принципы разработки: пять принципов ООП-проектирования (SOLID), «делай проще» (KISS), «не пиши лишнего» (YAGNI). Как они помогают удерживать качество ИИ-сгенерированного кода и формулировать ограничения для агента. 3. Паттерны как язык коммуникации с ИИ: устойчивые структуры кода (репозиторий, сервис, адаптер), которые задают рамки для генерации и рефакторинга. 4. Архитектурные правила в контексте проекта: как описать слои, зависимости и структуру каталогов, чтобы агент их соблюдал при генерации кода.

		Планирование разработки: декомпозиция на задачи, приоритизация, контроль версий, подключение через МСР. Обзор инструментов	1. Декомпозиция продукта на задачи: от документа требований к списку задач, пригодных для выполнения ИИ-агентами; оформление задач в средствах управления проектом. Обзор инструментов управления задачами. 2. Приоритизация и управление очередностью задач: как определить порядок выполнения, управлять зависимостями и объёмом работ. 3. Контроль версий: работа с ветками, фиксация изменений, запросы на слияние в ИИ-центричном процессе. 4. Протокол подключения моделей (МСР): подключение агента к репозиторию, средствам управления задачами, базам данных и внешним сервисам.
4.	Разработка и качество	Разработка с ИИ. Часть 1: настройка среды, подключение контекста, первые задачи для агента	1. Первая сессия разработки: настройка среды, подключение контекста проекта, запуск первых задач для ИИ-агента. 2. Координация агентов при разработке: как ставить задачи, контролировать выполнение, принимать или отклонять результат. 3. Типичные проблемы первой итерации: потеря контекста, отклонение от архитектуры, несоответствие требованиям — как диагностировать и исправлять.
		Документация: создание и генерация с помощью ИИ	1. Документация как вход для ИИ: как писать документы, которые одновременно полезны людям и повышают качество работы агентов. 2. Генерация документации с помощью ИИ: описание компонентов, интерфейсов (API), инструкций по развёртыванию; автоматическое обновление при изменении кода. 3. Поддержание документации в актуальном состоянии: организация хранения, связь с кодом и требованиями, предотвращение устаревания.
		Автоматическое тестирование с помощью ИИ	1. Типы тестов и их роль в ИИ-центричной разработке: модульные (unit), интеграционные (integration), сквозные (end-to-end) — как каждый помогает контролировать ИИ-сгенерированный код. 2. Генерация тестов с помощью ИИ: стратегия «сначала тест — потом код» (test-first), доработка сгенерированных тестов, проверка покрытия (coverage). 3. Тесты как спецификация поведения: использование тестов для фиксации ожиданий, чтобы агент не ломал работающий код при доработках.
		Разработка с ИИ. Часть 2: продолжение	1. Вторая сессия разработки: продолжение работы над проектом с учётом написанных тестов и документации.

		работы, рефакторинг (refactoring), работа с зависимостями	2. Рефакторинг (refactoring) ИИ-сгенерированного кода: вытаскивание логики в правильные слои, устранение дублирования, приведение к паттернам. 3. Работа с зависимостями и внешними сервисами: подключение баз данных, сторонних интерфейсов (API), платформ развёртывания.
		Отладка (debugging) ИИ-сгенерированного кода	1. Особенности ошибок ИИ-кода: код работает, но нарушает бизнес-логику; использование несуществующих интерфейсов (API); скрытые дефекты (bugs). 2. Цикл отладки (debugging cycle): воспроизведение, локализация, анализ, гипотеза, исправление с помощью ИИ, подтверждение; осознанное принятие или отклонение исправлений. 3. Инструменты отладки: точки останова, логирование, профилирование — зачем они нужны, даже когда основную часть кода пишет агент.
		Проверка кода (code review) с помощью ИИ	1. Проверка кода (code review) в ИИ-центричном процессе: что проверять, когда код написан агентом — соответствие требованиям, архитектуре, принципам проектирования. 2. ИИ как помощник при проверке: автоматический поиск проблем, объяснение незнакомых фрагментов, предложение улучшений. 3. Статический анализ (static analysis) и линтеры (linters): автоматизированные проверки стиля, типичных ошибок и уязвимостей как обязательный слой контроля.
5.	Развёртывание, тестирование и итерация	Развёртывание (deployment)	1. Общий конвейер доставки (CI/CD pipeline): от изменений в коде к работающему сервису — сборка (build), проверка, развёртывание (deploy), откат (rollback). 2. Окружения (environments) и стратегии развёртывания: локальное, тестовое (staging), рабочее (production); развёртывание через платформы (Vercel, Supabase), базовые сценарии отката. 3. Генерация конфигураций конвейера с помощью ИИ: создание сценариев сборки и развёртывания; границы того, что по-прежнему решает человек.
		Пользовательское тестирование (user testing)	1. Сценарное тестирование (usability testing) с реальными пользователями: подготовка сценариев, проведение тестирования, фиксация результатов. 2. Контрольные списки приёмочного тестирования (UAT — user acceptance testing):

			связь пользовательских сценариев с критериями приёмки из документации требований. 3. Анализ результатов: выявление расхождений между ожиданиями пользователей и поведением продукта, приоритизация найденных проблем.
		Доработка пользовательских сценариев	1. Анализ обратной связи от пользовательского тестирования: систематизация замечаний, выявление закономерностей, отделение критичных проблем от косметических. 2. Обновление пользовательских историй (user stories) и задач пользователей (JTBD): корректировка требований на основе реального поведения пользователей. 3. Переприоритизация списка задач (backlog): какие задачи добавить, какие отложить, как обновить документацию требований и контекст для агентов.
		Разработка с ИИ. Часть 3: реализация доработок, финализация инженерного прототипа	1. Третья сессия разработки: реализация доработок по результатам тестирования и обновлённым пользовательским сценариям. 2. Итеративная работа с агентом: уточнение запросов на основе накопленного опыта, использование истории проекта как контекста. 3. Финализация функциональности: доведение продукта до инженерного прототипа полного сценария, подготовка к мониторингу.
6.	«Результаты»	Мониторинг (monitoring), аналитика, обратная связь. Итерация	1. Минимальный набор метрик (metrics) и журналов (logs) после запуска: что отслеживать, чтобы вовремя обнаружить проблемы и понять поведение пользователей. 2. Замыкание цикла: как результаты мониторинга и аналитики превращаются в новые гипотезы и задачи — возврат к исследованию потребностей. 3. Настройка ИИ-агентов для мониторинга: автоматический анализ журналов, оповещения (alerts), генерация отчётов.

3.2. Методические указания для обучающихся по освоению курса

Лекция – систематическое, последовательное, монологическое изложение преподавателем учебного материала, как правило, теоретического характера.

В процессе лекций рекомендуется вести конспект лекций: кратко и схематично фиксировать основные идеи, выводы и обобщения лекции; выделять важные мысли, ключевые слова и термины. Необходимо отметить вопросы или материалы, которые вызывают затруднения, и попытаться найти ответы в

рекомендованной литературе. Если разобраться в материале не удастся, следует сформулировать вопрос и задать его преподавателю на консультации или во время семинарского (практического) занятия.

Участие в семинаре (практическом занятии) – активная работа обучающегося на семинаре (практическом занятии), его ответы на вопросы преподавателя и участие в дискуссии.

Для успешного участия в семинаре (практическом занятии) обучающимся рекомендуется заранее ознакомиться с темой обсуждения, прочитать необходимые материалы и подготовить вопросы. Важно активно слушать и вовлекаться в дискуссию, высказывая свои мнения и аргументируя их. При ответах на вопросы преподавателя стоит быть уверенным, четким и логичным, опираясь на изученный материал. Также полезно поддерживать диалог с другими обучающимися, чтобы обогатить обсуждение и расширить свои знания.

Домашнее задание – набор задач по темам недели.

При работе над домашними заданиями важно внимательно ознакомиться с требованиями и сроками выполнения. Рекомендуется разбивать задания на этапы, чтобы избежать перегрузки и лучше усвоить материал, использовать различные источники информации, включая учебники и онлайн-ресурсы, для более глубокого понимания темы.

Проект – исследовательская работа по курсу и презентация результатов.

Для успешной подготовки к проекту рекомендуется: четко определить цели и задачи проекта; составить план работы, разбив проект на этапы с указанием сроков выполнения каждого из них; использовать разнообразные источники информации и инструменты для исследования темы; регулярно проверять прогресс и вносить коррективы в план, если это необходимо.

Самостоятельная работа – работа обучающихся, направленная на углубленное изучение отдельных тем и вопросов курса.

В процессе самостоятельной работы обучающиеся взаимодействуют с рекомендованными материалами при минимальном участии преподавателя. Задачи обучающегося включают работу с конспектами лекций (обработка текста), повторное изучение учебных материалов планов и тезисов ответов, изучение дополнительных тем, выполнение учебно-исследовательских заданий и другое.

4. ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ

4.1. Требования к кадровым условиям реализации программы

Реализация программы обеспечивается штатными руководящими и научно-педагогическими работниками АНО ВО «Центральный университет», а также внешними совместителями, работающими по договорам гражданско-правового характера. Научно-педагогические работники, осуществляющие преподавание данной программы, имеют образование, соответствующее профилю курса, или конкретный опыт реализации разработок и иной формы практической деятельности по направлению курса.

4.2. Требования к материально-техническим условиям реализации программы

Университет располагает материально-технической базой, соответствующей действующим противопожарным правилам и нормам и обеспечивающей проведение всех видов учебной деятельности, предусмотренных учебным планом.

Помещения представляют собой учебные аудитории для проведения занятий лекционного типа, занятий семинарского (практического) типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы и помещения для хранения и профилактического обслуживания учебного оборудования. Помещения укомплектованы специализированной мебелью и техническими средствами обучения, служащими для представления учебной информации большой аудитории.

Изучение курса обеспечивается в учебных аудиториях, оснащенных:

- столами и стульями;
- компьютерной техникой;

— специализированным оборудованием, включая демонстрационное оборудование.

Помещения для самостоятельной работы обучающихся, в том числе приспособленные для использования инвалидами и лицами с ограниченными возможностями здоровья, оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа к в электронную информационно-образовательную среду Университета.

Обучающимся предоставляется доступ (в том числе удаленный) к ресурсам информационно-телекоммуникационной сети «Интернет», электронным ресурсам (в том числе электронным библиотечным системам, современным профессиональным базам данных и информационным справочным системам):

№	Наименование портала (издания, курса, документа)	Ссылка
1.	Научная электронная библиотека elibrary.ru библиотека	https://elibrary.ru/defaultx.asp
2.	База данных для IT-специалистов	https://habr.com
3.	База данных ScienceDirect	https://www.sciencedirect.com
4.	Официальный сайт Министерства науки и высшего образования Российской Федерации	https://minobrnauki.gov.ru/
5.	Федеральный портал «Российское образование»	https://www.edu.ru/
6.	Информационная система "Единое окно доступа к образовательным ресурсам"	http://window.edu.ru/
7.	Единая коллекция цифровых образовательных ресурсов	http://school-collection.edu.ru/
8.	Федеральный центр информационно - образовательных ресурсов	http://fcior.edu.ru/

Перечень информационных технологий, используемых при осуществлении образовательного процесса по курсу, в том числе комплект лицензионного программного обеспечения, современные профессиональные базы данных и информационные справочные системы:

Наименование ПО	Производство	Лицензионное / свободно распространяемое
Операционные системы:		
Microsoft Imagine (Windows Client, Server)	зарубежное	лицензионное

Браузеры:		
Яндекс.Браузер	отечественное	свободно распространяемое
Google Chrome	зарубежное	свободно распространяемое
Офисные приложения:		
Microsoft Imagine (Visio, OneNote)	зарубежное	лицензионное
TeXstudio	зарубежное	свободно распространяемое
Adobe Acrobat Reader	зарубежное	свободно распространяемое
Программное обеспечение для планирования и учета времени:		
Toggle app	зарубежное	свободно распространяемое
Системы управления проектами:		
Microsoft Imagine (Project)	зарубежное	лицензионное
Системы управления базами данных:		
Microsoft Imagine (SQL Server)	зарубежное	лицензионное
Системы резервного копирования (backup):		
Acronis Backup Advanced for HyperV	зарубежное	лицензионное
Справочно-правовые системы:		
КонсультантПлюс: справочно-правовая система	отечественное	лицензионное
Средства антивирусной защиты:		
Kaspersky Endpoint Security для бизнеса Стандартный Russian Edition	отечественное	лицензионное
Среды разработки:		
Visual Studio Code	зарубежное	свободно распространяемое
Bash (Unix shell)	зарубежное	свободно распространяемое
Anaconda	зарубежное	свободно распространяемое
Robotic Operating System	зарубежное	свободно распространяемое
CopelliaSim	зарубежное	свободно распространяемое
Google Colaboratory	зарубежное	свободно распространяемое
Пакеты программных средств и библиотек:		
AutoPsy	зарубежное	свободно распространяемое
Interactive Disassembler (IDA)	зарубежное	свободно распространяемое
Системы управления библиографической информацией:		
Zotero	зарубежное	свободно распространяемое
Сервисы и службы:		
Bind	зарубежное	свободно распространяемое
Docker	зарубежное	свободно распространяемое
ИИ-инструменты:		
Claude	зарубежное	свободно распространяемое

4.3. Учебно-методическое обеспечение программы

Университет располагает полным набором лицензионного и свободно распространяемого программного обеспечения, включая продукты отечественного производства.

Каждый слушатель в течение всего периода обучения получает индивидуальный неограниченный доступ к электронно-библиотечной системе и

электронной информационно-образовательной среде университета. Эти системы предоставляют возможность доступа к ресурсам из любой точки, где есть подключение к сети Интернет, как на территории университета, так и за его пределами.

Слушателям обеспечен удаленный доступ к современным профессиональным базам данных и информационным справочным системам.

Основная литература:

1. Чекмарев, А. В. Управление цифровыми проектами и процессами : учебник для вузов / А. В. Чекмарев. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2026. — 424 с. — (Высшее образование). — ISBN 978-5-534-18522-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/586395>.

2. Чернышев, С. А. Принципы, паттерны и методологии разработки программного обеспечения : учебник для вузов / С. А. Чернышев. — Москва : Издательство Юрайт, 2026. — 176 с. — (Высшее образование). — ISBN 978-5-534-14383-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/588769>.

3. Введение в программную инженерию : учебник / В.А. Антипов, А.А. Бубнов, А.Н. Пылькин, В.К. Столчев. — Москва : КУРС : ИНФРА-М, 2025. — 336 с. - ISBN 978-5-906923-22-6. - Текст : электронный. - URL: <https://znanium.ru/catalog/product/2173919>.

4. Воронов, М. В. Системы искусственного интеллекта : учебник и практикум для вузов / М. В. Воронов, В. И. Пименов, И. А. Небаев. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2026. — 268 с. — (Высшее образование). — ISBN 978-5-534-17032-0. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/588642>.

5. Вешкурова, А. Б. Управление командой проекта : учебник для вузов / А. Б. Вешкурова, Н. А. Копылова. — Москва : Издательство Юрайт, 2026. — 105 с. — (Высшее образование). — ISBN 978-5-534-21931-9. — Текст : электронный

// Образовательная платформа Юрайт [сайт]. — URL:
<https://urait.ru/bcode/599114>.

Дополнительная литература:

1. Левенчук, А. И. Практическое системное мышление : учебник / А. И. Левенчук. — Москва : [б. и.], 2023. — 734 с.

5. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

5.1. Формы контроля

В процессе изучения курса в рамках текущего контроля успеваемости используются такие виды учебной работы, как лекции, семинары (практические занятия), домашние задания, проект, а также различные виды самостоятельной работы обучающихся по заданию преподавателя, направленные на развитие навыков профессиональной лексики, закрепление практических профессиональных компетенций, поощрение инициатив.

Итоговая аттестация по программе осуществляется в форме *зачета*, при этом проводится оценка компетенций, сформированных по курсу.

5.2. Система оценивания результатов обучения по курсу

Для оценивания текущего контроля успеваемости и итоговой аттестации используется десятибалльная шкала оценивания, которая соотносится с традиционной пятибалльной шкалой следующим образом:

Десятибалльная оценка	Пятибалльная оценка	Оценка за зачет	Общая характеристика результата обучения по курсу
10	Отлично	Зачтено	Обучающийся полностью владеет знаниями, изложенными в рабочей программе, и глубоко осмысляет курс. Он самостоятельно и логически последовательно отвечает на все вопросы, акцентируя внимание на наиболее важном. Умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя ключевые моменты и устанавливая причинно-следственные связи. Четко формулирует ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает
9	Отлично	Зачтено	
8	Отлично	Зачтено	

Десятибалльная оценка	Пятибалльная оценка	Оценка за зачет	Общая характеристика результата обучения по курсу
			сложные задачи. Обучающийся хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты курса с практическими задачами.
7	Хорошо	Зачтено	Обучающийся обладает знаниями предмета почти в полном объеме рабочей программы и самостоятельно, логически последовательно и всесторонне отвечает на все вопросы, акцентируя внимание на наиболее значимых моментах. Он умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя его ключевые аспекты и устанавливая причинно-следственные связи. Формулирует свои ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные ситуационные задачи. Обучающийся хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты предмета с практическими задачами.
6	Хорошо	Зачтено	
5	Удовлетворительно	Зачтено	Обучающийся обладает базовыми знаниями по курсу, но испытывает трудности при самостоятельных ответах и использует неточные формулировки. В ходе ответов он допускает ошибки, касающиеся сути вопросов. Обучающийся способен решать только самые простые задачи и владеет лишь минимальным набором методов исследования.
4	Удовлетворительно	Зачтено	
3	Не сдан	Не зачтено	Обучающийся не овладел обязательным минимумом знаний по
2	Не сдан	Не зачтено	

Десятибалльная оценка	Пятибалльная оценка	Оценка за зачет	Общая характеристика результата обучения по курсу
1	Не сдан	Не зачтено	предмету и не может ответить на вопросы, даже если преподаватель задает дополнительные наводящие вопросы.

5.3. Примерные задания текущего контроля успеваемости

Примерное домашнее задание по разделу «Основы ИИ-центричной разработки»

Задача: выбрать и утвердить индивидуальный проект на курс, настроить базовый контекст проекта для работы с ИИ-агентами.

Состав работ:

1. Описание выбранного проекта: проблема, целевая аудитория, ожидаемый функционал инженерного прототипа полного сценария.
2. Настроенный контекст проекта в репозитории: правила проекта, инструкции для ИИ-агентов, начальная структура документации, описание стандартов и архитектурных ограничений.

Критерии оценивания:

- Описание проекта оформлено в виде отдельного документа в репозитории, в нем сформулированы: понятны проблема, аудитория и ожидаемый результат.
- Правила и инструкции для агентов оформлены отдельными документами в репозитории.
- Структура документации позволяет агенту находить нужную информацию без перегрузки контекстного окна.
- Обучающийся может объяснить, какие настройки контекста и зачем были сделаны.

Примерное домашнее задание по разделу «Продуктовое исследование и анализ»

Задача: провести исследование потребностей, сформировать пользовательские сценарии, оформить документацию требований и спроектировать структуру данных.

Состав работ:

1. Результаты исследования потребностей: продуктовые гипотезы, итоги интервью с синтетическими персонами, краткий анализ рынка.

2. Набор пользовательских сценариев (user stories) и задач пользователей (JTBD), привязанных к гипотезам.
3. Документация требований к продукту: структурированный документ с описанием функциональности, ограничений и критериев приёмки.
4. Схема структуры данных: сущности, атрибуты, связи; формат, пригодный для использования ИИ-агентом.

Критерии оценивания:

- Представлены все четыре артефакта.
- Документация требований пригодна для постановки задач ИИ-агенту: формулировки конкретны, критерии приёмки измеримы.

Примерное домашнее задание по разделу «Архитектура и планирование»

Задача: разработать архитектурное решение и составить план разработки с приоритизацией.

Состав работ:

1. Архитектурное решение: описание компонентов, потоков данных, выбранных паттернов.
2. Архитектурные правила в контексте проекта (для ИИ-агента): слои, зависимости, структура каталогов.
3. План разработки: список задач с приоритизацией в системе управления задачами; описание зависимостей между задачами.
4. Настроенное подключение через MCP к репозиторию, системе управления задачами и внешним сервисам проекта.

Критерии оценивания:

- Архитектурные правила сформулированы так, чтобы ИИ-агент мог им следовать при генерации кода (правила конкретны и проверяемы).
- План разработки декомпозирован до задач, пригодных для постановки ИИ-агенту: у каждой задачи есть понятный критерий завершённости.
- Подключение через MCP работает: агент имеет доступ к необходимому набору источников контекста.

Примерное домашнее задание по разделу «Разработка и качество»

Задача: разместить в репозитории рабочую версию кода проекта, документацию и автоматические тесты.

Состав работ:

1. Работающий код проекта (первая итерация), реализующий ключевые задачи из плана разработки.
2. Документация проекта: описание компонентов, интерфейсов (API), инструкции по установке и запуску.
3. Автоматические тесты: модульные (unit), интеграционные (integration), сквозные (end-to-end) на критические сценарии; описание стратегии тестирования.
4. Результаты проверки кода (code review) и отладки с использованием ИИ: журнал замечаний, исправленные дефекты.

Критерии оценивания:

- Код реализует ключевые пользовательские сценарии и соответствует архитектурным правилам проекта.
- Тесты покрывают критические сценарии; обучающийся может объяснить выбранную стратегию покрытия (coverage).
- Документация соответствует текущему состоянию кода: описывает интерфейсы и порядок запуска.

Примерное домашнее задание по разделу «Развёртывание, тестирование и доработка»

Задача: развернуть прототип и провести пользовательское тестирование.

Состав работ:

1. Развёрнутый прототип: доступный по ссылке работающий продукт; описание окружений (environments) и сценария сборки и развёртывания.
2. Результаты пользовательского тестирования: сценарии тестирования, заполненные контрольные списки приёмочного тестирования (UAT), сводка замечаний от пользователей.
3. Доработанные пользовательские сценарии и обновлённый список задач на основе обратной связи; обоснование приоритетов доработок.
4. Реализованные доработки в коде: инженерный прототип полного пользовательского сценария.

Критерии оценивания:

- Прототип развёрнут и доступен по ссылке; описан базовый сценарий отката.
- Пользовательское тестирование проведено с реальными пользователями, результаты систематизированы.
- Доработки сценариев логически вытекают из результатов тестирования.
- Сформулированы доработки покрывающие приоритетные замечания; полный пользовательский сценарий проходит без критических ошибок.

5.4. Критерии оценки освоения дополнительной программы для взрослых

Формат итоговой аттестации: оценка портфолио проекта (набор артефактов всех этапов жизненного цикла) и защита итогового проекта.

Описание формата:

Обучающийся (или команда из 2–3 человек) представляет портфолио проекта — набор артефактов, созданных на каждом этапе курса, и демонстрирует работающий инженерный прототип. Длительность выступления — 10–15 минут, обсуждение — 5–10 минут.

Оцениваемые артефакты:

1. Результаты исследования потребностей: гипотезы, интервью с синтетическими персонами, анализ рынка.
2. Пользовательские сценарии и задачи пользователей (user stories, JTBD).
3. Документация требований.
4. Схема структуры данных.
5. Архитектурное решение.
6. План разработки: список задач с приоритизацией.
7. Настроенный контекст проекта: правила, инструкции для агентов.
8. Работающий код инженерного прототипа.
9. Документация проекта.
10. Тесты (автоматические и пользовательские).
11. Развёрнутый прототип.
12. Доработанные пользовательские сценарии по результатам тестирования.

Примерные критерии оценивания:

1. Полнота портфолио: наличие артефактов по каждому этапу жизненного цикла.

2. Осознанное применение ИИ-инструментов и агентов (обучающийся может объяснить, что и зачем делегировал ИИ на каждом этапе).

3. Работоспособность прототипа и реализация полного пользовательского сценария.

4. Связность артефактов: каждый следующий этап опирается на результаты предыдущего.

Зачёт выставляется при выполнении не менее 2 из 4 критериев.

6. ВЫХОДНЫЕ ДОКУМЕНТЫ

Лицам, успешно освоившим соответствующую дополнительную общеразвивающую программу для взрослых и прошедшим итоговую аттестацию, выдаётся сертификат об обучении, образец которого самостоятельно установлен образовательным учреждением.