

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«Магнитогорский государственный технический университет
им. Г.И. Носова»

Многопрофильный колледж

РАБОЧАЯ ПРОГРАММА ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

ПМ.01 «Разработка кода для искусственного интеллекта»

«профессионального цикла»

программы подготовки специалистов среднего звена

**по специальности 09.02.13 Интеграция решений с применением технологий искусственного
интеллекта**

Квалификация: Специалист по работе с искусственным интеллектом

Форма обучения

очная на базе основного общего образования

Рабочая программа профессионального модуля «Разработка кода для искусственного интеллекта» разработана на основе: ФГОС по специальности среднего профессионального образования 09.02.13 Интеграция решений с применением технологий искусственного интеллекта, утвержденного приказом Министерства просвещения Российской Федерации от «24» декабря 2024 г. №1025.

Организация-разработчик: Многопрофильный колледж ФГБОУ ВО «Магнитогорский государственный технический университет им. Г. И. Носова»

Разработчик (и):

преподаватель отделения №2 «Информационных технологий и транспорта»

Многопрофильного колледжа ФГБОУ ВО «МГТУ им. Г.И. Носова»

Анастасия Александровна Жак

преподаватель отделения №2 «Информационных технологий и транспорта»

Многопрофильного колледжа ФГБОУ ВО «МГТУ им. Г.И. Носова»

Оксана Викторовна Кобыльская

ОДОБРЕНО

Предметно-цикловой комиссией

«ПЦК 10»

Председатель Ремез Т.Б.

Протокол № 5.1 от «11» февраля 2026г.

Методической комиссией МпК

Протокол № 4 от «18» февраля 2026г.

СОДЕРЖАНИЕ

1 ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ	4
1.1 Цель и место модуля в структуре образовательной программы	4
1.2 Перечень планируемых результатов освоения профессионального модуля	4
1.3 Обоснование часов профессионального модуля в рамках вариативной части	8
2 СТРУКТУРА И СОДЕРЖАНИЕ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ	13
2.1 Структура профессионального модуля	13
2.2 Тематический план и содержание профессионального модуля	14
2.3 Перечень практических и лабораторных занятий	34
3 УСЛОВИЯ РЕАЛИЗАЦИИ РАБОЧЕЙ ПРОГРАММЫ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ	42
3.1 Материально-техническое обеспечение	42
3.2 Учебно-методическое и информационное обеспечение реализации программы	42
3.3 Учебно-методическое обеспечение самостоятельной работы обучающихся	43
4 КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ	46
4.1 Текущий контроль	46
4.2 Промежуточная аттестация	47
4.2.1 Оценочные средства для зачета, экзамена по МДК, практике	47
4.2.2 Экзамен по профессиональному модулю	55
Приложение 1 Образовательные технологии	61
Приложение 2 Фонд оценочных средств по дисциплине	64
Приложение 3 Методические указания	64

1 ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

1.1 Цель и место модуля в структуре образовательной программы

Рабочая программа профессионального модуля является частью профессионального цикла программы подготовки специалистов среднего звена по специальности 09.02.13 Интеграция решений с применением технологий искусственного интеллекта. Рабочая программа составлена для очной формы обучения.

Цель профессионального модуля: овладение видом профессиональной деятельности «Разработка кода для искусственного интеллекта».

В результате освоения профессионального модуля обучающиеся получают квалификацию «Специалист по работе с искусственным интеллектом».

Модуль «Разработка кода для искусственного интеллекта» включен в обязательную часть образовательной программы по специальности 09.02.13.

1.2 Перечень планируемых результатов освоения профессионального модуля

Результаты освоения профессионального модуля соотносятся с планируемыми результатами освоения образовательной программы, представленными в разделе 4 ППСЗ.

Требования к результатам освоения модуля

Код	Наименование вида деятельности и профессиональных компетенций
ВД 1	Разработка кода для искусственного интеллекта
ПК 1.1.	Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием
ПК 1.2.	Разрабатывать программные модули в соответствии с техническим заданием
ПК 1.3.	Оформлять программный код в соответствии с техническим заданием
ПК 1.4.	Использовать систему контроля версий программного кода с учетом обеспечения возможности организации групповой разработки
ПК 1.5.	Выполнять отладку программных модулей с использованием специализированных программных средств
ПК 1.6.	Выполнять тестирование программного кода
ПК 1.7.	Составлять тестовые сценарии

Код	Наименование общих компетенций
ОК 01.	Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам
ОК 02.	Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности
ОК 04.	Эффективно взаимодействовать и работать в коллективе и команде
ОК 05.	Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста
ОК 09.	Пользоваться профессиональной документацией на государственном и иностранном языках

В результате освоения профессионального модуля обучающийся:

Индекс и наименование ОК, ПК	Результаты освоения		
	Владеет навыками	Умеет	Знает
ПК 1.1. Формировать алгоритмы разработки программных модулей в	Н 1.1.1 анализа технического задания и выделения	У 1.1.1 анализировать техническое задание и	З 1.1.1 структуру и содержание технического задания на программный модуль

соответствии с техническим заданием	функциональных требований к модулю ИИ	выделять функциональные требования к программному модулю ИИ	
	Н 1.1.2 разработки блок-схем и псевдокода алгоритмов обработки данных	У 1.1.2 разрабатывать алгоритмы для мобильных ИИ-приложений и серверных модулей	З 1.1.2 основные алгоритмы обработки данных (сортировка, поиск, фильтрация) и принципы алгоритмизации
ПК 1.2. Разрабатывать программные модули в соответствии с ТЗ	Н 1.2.1 создания программных модулей на языке Python с использованием библиотек ИИ	У 1.2.1 писать программный код на Python с использованием циклов, условных операторов, функций и классов	З 1.2.1 синтаксис и семантику языка Python, принципы ООП
	Н 1.2.2 разработки мобильных приложений на Kotlin/Java с интеграцией ИИ-компонентов	У 1.2.2 применять библиотеки для ИИ (NumPy, Pandas, scikit-learn, TensorFlow, PyTorch)	З 1.2.2 назначение и возможности библиотек для ИИ
		У 1.2.3 разрабатывать программные модули для мобильных приложений с интеграцией ИИ на языках Kotlin/Java	З 1.2.3 языки программирования (Kotlin, Java), фреймворки для мобильной разработки (Android SDK, Jetpack Compose)
ПК 1.3. Оформлять программный код в соответствии с ТЗ	Н 1.3.1 оформления кода по стандарту PEP8/Kotlin conventions	У 1.3.1 оформлять код в соответствии со стандартами (PEP8, Kotlin conventions) и документировать его	З 1.3.1 правила оформления программного кода (PEP8, Kotlin conventions), форматы документации (Markdown, reStructuredText, KDoc)
	Н 1.3.2 генерации документации с помощью docstring, Sphinx, KDoc	У 1.3.2 генерировать документацию с помощью ИИ-ассистентов и инструментов автоматизации	З 1.3.2 инструменты для автоматической проверки стиля кода (flake8, pylint, black, ktlint)
ПК 1.4. Использовать систему контроля версий программного кода	Н 1.4.1 работы с локальными и удалёнными репозиториями Git	У 1.4.1 создавать локальный и удалённый репозитории Git, выполнять базовые операции: commit, push, pull, branch, merge	З 1.4.1 принципы работы систем контроля версий, основные команды Git и их назначение

	Н 1.4.2 применять промты для генерации описаний коммитов и pull request'ов, разрешать конфликты слияния	У 1.4.2 применять промты для генерации описаний коммитов и pull request'ов, разрешать конфликты слияния	З 1.4.2 модели ветвления (Git Flow, GitHub Flow), принципы организации групповой разработки
ПК 1.5. Выполнять отладку программных модулей с использованием специализированных программных средств	Н 1.5.1 использования отладчиков PyCharm/VS Code/Android Studio	У 1.5.1 устанавливать брейкпоинты, выполнять пошаговую отладку, использовать watch для отслеживания переменных	З 1.5.1 методы и средства отладки программного кода, типы ошибок (синтаксические, логические, времени выполнения),
	Н 1.5.2 применения логирования для диагностики ошибок	У 1.5.2 применять логирование (logging, Logcat) для диагностики ошибок, генерировать точки останова с помощью промтов	З 1.5.2 уровни логирования и формат записи логов, инструменты отладки (pdb, PyCharm Debugger, Android Studio Profiler)
	Н 1.5.3 применения инструментов автоматизации тестирования (pytest, JUnit, GitHub Actions)	У 1.5.3 разрабатывать тестовую документацию (тест-кейсы, чек-листы, тест-планы)	З 1.5.3 виды и методы тестирования программного обеспечения (модульное, интеграционное, системное, регрессионное)
ПК 1.6. Выполнять тестирование программного кода	Н 1.6.1 написания модульных тестов с использованием unittest/pytest/JUnit	У 1.6.1 писать модульные тесты с использованием unittest и pytest, запускать тесты и анализировать результаты	З 1.6.1 виды тестирования (модульное, интеграционное, системное), структуру тестового случая
	Н 1.6.2 оценки покрытия кода и автоматизации тестирования	У 1.6.2 проводить unit-тестирование и UI-тестирование мобильных приложений	З 1.6.2 инструменты тестирования (JUnit, Espresso, Mockito, pytest), виды тестирования
		У 1.6.3 оценивать покрытие кода тестами (pytest-cov), генерировать тесты с помощью ИИ-инструментов	З 1.6.3 принципы написания надёжных тестов
	Н 1.6.4 настройки CI/CD-пайплайнов для	У 1.6.4 проводить анализ результатов	З 1.6.4 метрики качества тестирования, принципы оценки покрытия кода

	автоматического запуска тестов	тестирования и формировать отчёты	
ПК 1.7. Составлять тестовые сценарии	Н 1.7.1 разработки тест-кейсов и чек-листов	У 1.7.1 разрабатывать тест-кейсы для проверки функциональности модуля, оформлять тестовый сценарий в виде документа	З 1.7.1 структуру тестового сценария (предусловия, шаги, ожидаемый результат, постусловия), методы проектирования тестов
	Н 1.7.2 автоматизации выполнения тестов через CI/CD	У 1.7.2 автоматизировать выполнение тестов с помощью CI (GitHub Actions), использовать промты для генерации тестовых сценариев	З 1.7.2 принципы непрерывной интеграции (CI/CD), инструменты автоматизации тестирования
	Н 1.7.3 использования промт-инжиниринга для генерации тестовых данных и сценариев	У 1.7.3 проектировать сценарии тестирования для ИИ-компонентов (распознавание, классификация, генерация)	З 1.7.3 особенности тестирования ИИ-систем (валидация моделей, метрики точности, обработка неопределённости)
ОК 01. Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам		Уо 01.01. Анализировать задачу, составлять план действий.	Зо 01.01. Актуальный профессиональный контекст, методы решения задач.
ОК 02. Использовать современные средства поиска информации		Уо 02.01. Эффективно искать релевантную информацию для составления промтов.	Зо 02.01. Источники информации (документация, научные статьи, форумы).

ОК 04. Эффективно взаимодействовать и работать в коллективе и команде		Уо 04.01. Организовывать работу в команде, использовать общие промты.	Зо 04.01. Психологические основы деятельности коллектива.
ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста		Уо 05.01. Грамотно оформлять промты и документацию к ним.	Зо 05.01. Правила оформления документов и устных сообщений.
ОК 09. Пользоваться профессиональной документацией на государственном и иностранном языках		Уо 09.01. Читать документацию на английском языке (OpenAI, Anthropic).	Зо 09.01. Правила чтения и перевода технической документации.

1.3 Обоснование часов профессионального модуля в рамках вариативной части

Дополнительные профессиональные компетенции	Дополнительные знания, умения, навыки	Номер и наименование темы	Объем часов	Обоснование включения в рабочую программу
	У 1.1.1, У 1.1.2, У 1.2.1, У 1.2.2, Уо 01.01, Уо 02.01, З 1.1.1, З 1.1.2, З 1.2.1, З 1.2.2, Зо 01.01, Зо 02.01	Раздел 1. Темы 1.1–1.2. Основы алгоритмизации и программирования на Python	86	Запрос работодателя на специалистов, способных разрабатывать эффективные алгоритмы для ИИ-систем с применением современных методов алгоритмизации и

				генерации кода через LLM
	У 1.2.1, У 1.2.2, У 1.2.3, Уо 02.02, З 1.2.1, З 1.2.2, З 1.2.3, Зо 02.02	Раздел 1. Темы 2.1–2.2. Работа с массивами, табличными данными и визуализация	72	Необходимость формирования компетенций по предобработке и визуализации больших объемов данных для обучения моделей ИИ с использованием автоматизированных инструментов
	У 1.2.1, У 1.2.2, У 1.2.3, Уо 01.02, З 1.2.1, З 1.2.2, З 1.2.3, Зо 01.02	Раздел 1. Темы 2.3–2.4. Библиотеки машинного обучения и нейросетевые библиотеки	148	Формирование ключевых компетенций по созданию, обучению и оценке моделей ИИ, востребованных на рынке труда; интеграция техник промт-инжиниринга для ускорения разработки
	У 1.3.1, У 1.3.2, У 1.4.1, У 1.4.2, Уо 04.01, З 1.3.1, З 1.3.2, З 1.4.1, З 1.4.2, Зо 04.01	Раздел 1. Темы 3.1–3.2. Стандарты оформления кода и система контроля версий	28	Обеспечение готовности выпускников к работе в команде, соблюдению корпоративных стандартов кодирования и использованию современных инструментов совместной разработки
	У 1.2.3, У 1.4.2, Уо 02.02, З 1.2.3, З 1.4.2, Зо 02.02	Раздел 2. Тема 2.3. Интеграция облачных ИИ-сервисов	28	Запрос работодателей на навыки использования отечественных и зарубежных облачных ИИ-решений в мобильной разработке
	У 1.2.2, У 1.5.2, Уо	Раздел 2. Тема 2.4.	26	Требование рынка:

	01.01, З 1.2.2, З 1.5.2, Зо 01.01	Оптимизация моделей для мобильных платформ		специалисты по мобильной разработке с ИИ должны уметь адаптировать модели для работы на устройствах с ограниченной вычислительной мощностью и памятью
	У 1.6.1, У 1.6.2, У 1.7.1, У 1.7.2, Уо 04.02, З 1.6.1, З 1.6.2, З 1.7.1, З 1.7.2, Зо 04.02	Раздел 2. Тема 3.3. Тестирование ИИ-компонентов	20	Востребованность специалистов, способных проверять качество работы специфических ИИ-функций (распознавание изображений, голоса) на мобильных устройствах
	У 1.6.1, У 1.6.2, У 1.7.1, У 1.7.2, Уо 04.02, З 1.6.1, З 1.6.2, З 1.7.1, З 1.7.2, Зо 04.02	Раздел 3. МДК.01.03 Отладка и тестирование программных модулей	28	Востребованность специалистов, способных проверять качество работы специфических ИИ-функций (распознавание изображений, голоса) на мобильных устройствах
	У 1.5.1, У 1.5.2, З 1.5.1, З 1.5.2	Раздел 3. Тема 1.1. Основы отладки и диагностики программного кода	24	Освоение навыков пошаговой отладки, работы с брейкпоинтами и внедрения многоуровневого логирования для диагностики ошибок в ИИ-модулях
	У 1.6.1, У 1.6.2, З 1.6.1, З 1.6.2	Раздел 3. Тема 1.2. Модульное и интеграционное тестирование	32	Формирование умений написания юнит-тестов (pytest, unittest), использования моков и оценки покрытия кода для обеспечения надежности

				модулей
	У 1.6.3, У 1.7.1, З 1.6.3, З 1.7.1	Раздел 3. Тема 1.3. Тестирование ИИ-моделей и нейросетевых компонентов	40	Освоение специфических метрик оценки ИИ (accuracy, F1, ROC-AUC), тестирование устойчивости моделей к зашумленным данным и adversarial-атакам
	У 1.7.2, У 1.7.3, З 1.7.2, З 1.7.3	Раздел 3. Тема 1.4. Автоматизация тестирования и CI/CD	32	Навыки настройки пайплайнов непрерывной интеграции (GitHub Actions) для автоматического запуска тестов и контроля качества кода при сборке
	У 1.7.1, Уо 05.01, З 1.7.1, Зо 05.01	Раздел 3. Тема 1.5. Разработка тестовой документации и сценариев	24	Разработка тест-планов, чек-листов и сценариев с использованием ИИ-ассистентов; грамотное оформление отчетности о тестировании
	У 1.6.1, У 1.6.2, У 1.7.1, У 1.7.2, Уо 04.02, З 1.6.1, З 1.6.2, З 1.7.1, З 1.7.2, Зо 04.02	Раздел 3. Тема 1.3. Тестирование ИИ-компонентов (мобильные приложения)	28	Востребованность специалистов, способных проверять качество работы специфических ИИ-функций (распознавание изображений, голоса) на мобильных устройствах
Всего академических часов профессионального модуля в рамках вариативной части			596	

1.4 Трудоемкость профессионального модуля

Наименование составных частей профессионального	Объем в часах	В т.ч. в форме практической подготовки
---	---------------	--

модуля		
Теоретические занятия	208	-
Практические занятия	Не предусмотрено	-
Лабораторные занятия	746	566
Курсовая работа (проект)	Не предусмотрено	-
Консультации	Не предусмотрено	-
Самостоятельная работа	62	-
Практика, в т.ч.:	324	324
учебная	252	252
производственная	72	72
Промежуточная аттестация	60	-
Всего	1400	890

2 СТРУКТУРА И СОДЕРЖАНИЕ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

2.1 Структура профессионального модуля

Индекс ОК/ПК	Наименования разделов профессионального модуля/МДК	Формы промежуточной аттестации (семестр)					Объем профессионального модуля, час.									
							Объем ОП, час с	Самостоятельная работа	Всего	с п р е п о д а в а т е л е м						Промежуточная аттестация
		Экзамены	Зачеты	Зачет с оценкой	Курсовые проекты	Курсовые работы				в практической подготовке	лекции, уроки	практические занятия	лабораторные занятия	курсовой проект (задание)	Консультаций	
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
ПК 1.1–1.7 ОК 01, 02, 04, 05, 09	Раздел 1 МДК.01.01 Разработка программных модулей в системах искусственного интеллекта	3,4		6			722	40	722	334	132		514			36
ПК 1.1–1.7 ОК 01, 02, 04, 05, 09	Раздел 2. МДК.01.02 Разработка мобильных приложений с поддержкой искусственного интеллекта	6					190	14	190	120	44		120			12
ПК 1.5–1.7 ОК 01, 02, 04, 05, 09	Раздел 3. МДК.01.03 Отладка и тестирование программных модулей			6			152	8	152	112	32		112			
ПК 1.1–1.7 ОК 01, 02, 04, 05, 09	Учебная практика		4,5 ,6				252		252	252						
ПК 1.1–1.7 ОК 01, 02, 04, 05, 09	Производственная практика		6к				72		72	72						
ПК 1.1–1.7 ОК 01, 02, 04, 05, 09	Экзамен по профессиональному модулю	6					12		12							12
	Всего	4	4	2			1400	62	1400	890	208		746			60

2.2 Тематический план и содержание профессионального модуля

Наименование разделов и тем профессионального модуля (ПМ), междисциплинарных курсов (МДК)	Содержание учебного материала, практические и лабораторные занятия, курсовая работа (проект), самостоятельная работа, консультации, учебная и/или производственная практика	Объем, акад. ч / в том числе в форме практической подготовки, акад.ч.	Код ПК, ОК	Коды осваиваемых элементов компетенций
1	2	3	4	5
МДК.01.01 Разработка программных модулей в системах искусственного интеллекта		722 / 334		
Раздел 1. Разработка программных модулей в системах искусственного интеллекта		722 / 334		
Тема 1.1. Введение в искусственный интеллект и постановка задач	Содержание	16 / 10		
	1. Понятие искусственного интеллекта, основные направления.	8/0	ПК 1.1, ОК 01	З 1.1.1, З 1.1.2, Зо 01.01
	2. Жизненный цикл разработки программного модуля. Структура технического задания.	8/0	ПК 1.1, ОК 01	З 1.1.1, З 1.1.2, Зо 01.01
	В том числе лабораторных занятий	10 / 10		
	Лабораторное занятие №1. Анализ технического задания на разработку модуля ИИ.	10 / 10	ПК 1.1, ОК 01	У 1.1.1, У 1.1.2, Уо 01.01
	Самостоятельная работа	2		
	Изучение структуры ТЗ, выделение функциональных требований.	2	ПК 1.1	У 1.1.1, У 1.1.2
Тема 1.2. Базовые	Содержание	76 / 40		

конструкции языка Python	1. Переменные, типы данных, операторы, условные операторы.	38/0	ПК 1.2, ОК 02	З 1.2.1, З 1.2.2, З 1.2.3, Зо 02.01
	2. Циклы (for, while), работа со списками и словарями.	38/0	ПК 1.2, ОК 02	З 1.2.1, З 1.2.2, З 1.2.3, Зо 02.01
	В том числе лабораторных занятий	40 / 40		
	Лабораторное занятие №2. Разработка скрипта для обработки данных с использованием циклов и условных операторов.	20 / 20	ПК 1.2, ОК 02	Уд 1.2, Уо 02.01
	Лабораторное занятие №3. Создание функций для модульной обработки данных.	20 / 20	ПК 1.2, ОК 02	У 1.2.1, У 1.2.2, У 1.2.3, Уо 02.01
	Самостоятельная работа	12		
	Решение задач на программирование (5 программ по темам: условные операторы, циклы, функции).	12	ПК 1.2	У 1.2.1, У 1.2.2, У 1.2.3
Тема 1.3. Работа с массивами и табличными данными (NumPy, Pandas)	Содержание	48 / 36		
	1. Установка и импорт библиотек. Создание массивов NumPy.	24/0	ПК 1.2, ОК 02	З 1.2.1, З 1.2.2, З 1.2.3, Зо 02.02
	2. Серии и DataFrame в Pandas. Чтение/запись данных (CSV, Excel).	24/0	ПК 1.2, ОК 02	З 1.2.1, З 1.2.2, З 1.2.3, Зо 02.02
	В том числе лабораторных занятий	36 / 36		
	Лабораторное занятие №4. Загрузка и предобработка данных с помощью Pandas.	20 / 20	ПК 1.2, ОК 02	У 1.2.1, У 1.2.2, У 1.2.3, Уо 02.02

	Лабораторное занятие №5. Векторизованные вычисления с NumPy.	16 / 16	ПК 1.2, ОК 02	У 1.2.1, У 1.2.2, У 1.2.3, Уо 02.02
	Самостоятельная работа	6		
	Подготовка набора данных из открытых источников для обучения.	6	ПК 1.2	У 1.2.1, У 1.2.2, У 1.2.3
Тема 1.4. Визуализация данных (Matplotlib, Seaborn)	Содержание	22 / 14		
	1. Построение графиков, гистограмм, диаграмм рассеяния.	12/0	ПК 1.2, ОК 02	З 1.2.1, З 1.2.2, З 1.2.3
	2. Настройка визуализации (подписи, легенда, стили).	12/0	ПК 1.2, ОК 02	З 1.2.1, З 1.2.2, З 1.2.3
	В том числе лабораторных занятий	14 / 14		
	Лабораторное занятие №6. Построение графиков распределения данных.	14 / 14	ПК 1.2, ОК 02	У 1.2.1, У 1.2.2, У 1.2.3
Тема 1.5. Библиотеки машинного обучения (scikit-learn)	Содержание	56 / 44		
	1. Задачи регрессии и классификации. Обучение модели.	28/0	ПК 1.2, ПК 1.3	З 1.2.1, З 1.2.2, З 1.2.3
	2. Оценка качества модели (MSE, R2, accuracy, precision, recall).	28/0	ПК 1.2, ПК 1.3	З 1.2.1, З 1.2.2, З 1.2.3
	В том числе лабораторных занятий	44 / 44		
	Лабораторное занятие №7. Реализация линейной регрессии на scikit-learn.	22 / 22	ПК 1.2, ПК 1.3	У 1.2.1, У 1.2.2, У 1.2.3

	Лабораторное занятие №8. Классификация с помощью метода k-ближайших соседей.	22 / 22	ПК 1.2, ПК 1.3	У 1.2.1, У 1.2.2, У 1.2.3
Тема 1.6. Нейросетевые библиотеки (TensorFlow, PyTorch)	Содержание	66 / 48		
	1. Основы построения нейронных сетей. Слои, функции активации, обучение.	28/0	ПК 1.2, ОК 09	З 1.2.1, З 1.2.2, З 1.2.3, З 09.01
	2. Работа с TensorFlow/Keras и PyTorch.	38/0	ПК 1.2, ОК 09	З 1.2.1, З 1.2.2, З 1.2.3, З 09.01
	В том числе лабораторных занятий	48 / 48		
	Лабораторное занятие №9. Создание и обучение простой нейронной сети на TensorFlow/Keras.	24 / 24	ПК 1.2, ОК 09	У 1.2.1, У 1.2.2, У 1.2.3, З 09.01
	Лабораторное занятие №10. Использование PyTorch для построения нейросети.	24 / 24	ПК 1.2, ОК 09	У 1.2.1, У 1.2.2, У 1.2.3, З 09.01
	Самостоятельная работа	16		
	Подбор гиперпараметров нейронной сети, сравнение результатов.	16	ПК 1.2, ОК 09	У 1.2.1, У 1.2.2, У 1.2.3, З 1.2.1, З 1.2.2, З 1.2.3
Тема 1.7. Проектирование архитектуры ИИ-систем	Содержание	84 / 68		
	1. Модульность и масштабируемость ИИ-систем.	42/0	ПК 1.1, ПК 1.2	З 1.1.1, З 1.1.2, З 1.2.1, З 1.2.2, З 1.2.3
	2. Интеграция ИИ в реальные проекты (API, контейнеризация).	42/0	ПК 1.1, ПК 1.2	З 1.1.1, З 1.1.2, З 1.2.1, З 1.2.2, З 1.2.3

	В том числе лабораторных занятий	68 / 68		
	Лабораторное занятие №11. Проектирование архитектуры ИИ-системы с учетом модульности.	34 / 34	ПК 1.1, ПК 1.2	У 1.1.1, У 1.1.2, У 1.2.1, У 1.2.2, У 1.2.3
	Лабораторное занятие №12. Контейнеризация ИИ-модели с использованием Docker.	34 / 34	ПК 1.1, ПК 1.2	У 1.1.1, У 1.1.2, У 1.2.1, У 1.2.2, У 1.2.3
Тема 1.8. Стандарты оформления кода (PEP8, документация)	Содержание	42 / 34		
	1. Правила именования, отступы, комментарии.	22/0	ПК 1.3	З 1.3.1, З 1.3.2
	2. Генерация документации с помощью docstring и Sphinx.	20/0	ПК 1.3	З 1.3.1, З 1.3.2
	В том числе лабораторных занятий	34 / 34		
	Лабораторное занятие №13. Приведение кода к стандарту PEP8 с использованием линтеров (flake8, pylint).	20 / 20	ПК 1.3	У 1.3.1, У 1.3.2, З 1.3.1, З 1.3.2, Зо 04.01
	Лабораторное занятие №14. Написание документации к разработанному модулю.	14 / 14	ПК 1.3	У 1.3.1, У 1.3.2, З 1.3.1, З 1.3.2, Зо 04.01
Тема 1.9. Система контроля версий Git	Содержание	32 / 20		
	1. Основы Git: репозиторий, commit, ветвление, слияние.	12/0	ПК 1.4, ОК 04	Зд 3.2, Зо 04.02
	2. Удалённые репозитории (GitHub), pull request.	30/0	ПК 1.4, ОК 04	Зд 3.2, Зо 04.02

	В том числе лабораторных занятий	20 / 20		
	Лабораторное занятие №15. Создание локального репозитория, коммиты, работа с ветками.	20 / 20	ПК 1.4, ОК 04	У 1.4.1, У 1.4.2, Уо 04.01
	Самостоятельная работа	8		
	Регистрация на GitHub, загрузка проекта, создание pull request.	8	ПК 1.4, ОК 04	У 1.4.1, У 1.4.2, Уо 04.01
Тема 1.10. Отладка программного кода	Содержание	76 / 40		
	1. Инструменты отладки (breakpoints, watch).	38/0	ПК 1.5	З 1.5.1, З 1.5.2, З 1.5.3
	2. Логирование (logging).	38/0	ПК 1.5	З 1.5.1, З 1.5.2, З 1.5.3
	В том числе лабораторных занятий	40 / 40		
	Лабораторное занятие №16. Использование отладчика в PyCharm для поиска ошибок.	20 / 20	ПК 1.5	У 1.5.1, У 1.5.2, У 1.5.3
	Лабораторное занятие №17. Внедрение логирования в модуль ИИ.	20 / 20	ПК 1.5	У 1.5.1, У 1.5.2, У 1.5.3
	Самостоятельная работа	12		
	Поиск и устранение ошибок в чужом коде.	12	ПК 1.5	У 1.5.1, У 1.5.2, У 1.5.3
Тема 1.11. Тестирование программных модулей	Содержание	68 / 56		
	1. Виды тестирования: модульное, интеграционное.	34/0	ПК 1.6	З 1.6.1, З 1.6.2, З

				1.6.3, 3 1.6.4
	2. Написание тестов с unittest и pytest.	34/0	ПК 1.6	3 1.6.1, 3 1.6.2, 3 1.6.3, 3 1.6.4
	В том числе лабораторных занятий	56 / 56		
	Лабораторное занятие №18. Разработка модульных тестов для функций обработки данных (unittest).	28 / 28	ПК 1.6, ПК 1.7	У 1.6.1, У 1.6.2, У 1.6.3, У 1.6.4, 3 1.6.1, 3 1.6.2, 3 1.6.3, 3 1.6.4
	Лабораторное занятие №19. Написание тестов для нейросетевой модели (pytest).	28 / 28	ПК 1.6, ПК 1.7	У 1.6.1, У 1.6.2, У 1.6.3, У 1.6.4, 3 1.6.1, 3 1.6.2, 3 1.6.3, 3 1.6.4
	Самостоятельная работа	12		
	Анализ покрытия кода тестами.	12	ПК 1.6	У 1.6.1, У 1.6.2, У 1.6.3, У 1.6.4
Тема 1.12. Составление тестовых сценариев	Содержание	56 / 44	ПК 1.7	3 1.7.1, 3 1.7.2, 3 1.7.3
	1. Тест-кейсы, тест-планы, автоматизация тестирования.	56/0	ПК 1.7	3 1.7.1, 3 1.7.2, 3 1.7.3
	В том числе лабораторных занятий	44 / 44		
	Лабораторное занятие №20. Разработка тестового сценария для модуля классификации.	20 / 20	ПК 1.7, ПК 1.6	У 1.7.1, У 1.7.2, У 1.7.3
	Лабораторное занятие №21. Автоматизация запуска	24 / 24	ПК 1.7, ПК 1.6	У 1.7.1, У 1.7.2,

	тестов с использованием CI (GitHub Actions).			У 1.7.3
	Самостоятельная работа	4		
	Анализ результатов тестирования, составление отчёта.	4	ПК 1.7	У 1.7.1, У 1.7.2, У 1.7.3
МДК 01.02 Разработка мобильных приложений с поддержкой искусственного интеллекта		190 / 120		
Раздел 2. Разработка мобильных приложений с поддержкой искусственного интеллекта		190/ 120		
Тема 2.1. Введение в разработку мобильных приложений	Содержание	6/6		
	1. Платформы Android и iOS: сравнение, особенности, рыночные доли. История развития.	2	ПК 1.2, ОК 02	З 1.2.1, З 1.2.2, З 1.2.3, Зо 02.01
	2. Установка и настройка Android Studio. Создание первого приложения (Hello World).	2	ПК 1.2, ПК 1.3	З 1.2.1, З 1.2.2, З 1.2.3, Зо 02.01
	3. Структура Android-проекта: манифест, ресурсы, Gradle. Основные компоненты (Activity, Fragment, Service).	2	ПК 1.2, ОК 09	З 1.2.1, З 1.2.2, З 1.2.3, Зо 02.01
	В том числе лабораторных занятий	6/6		
	Лабораторное занятие №22. Создание первого Android-приложения. Настройка эмулятора. Разработка простого интерфейса (TextView, Button, Toast). Обработка нажатий.	6 / 6	ПК 1.2, ПК 1.5, ОК 02	У 1.2.1, У 1.2.2, У 1.2.3, У 1.5.1, У 1.5.2, У 1.5.3, Уо 02.01
Тема 2.2. Языки программирования для мобильной разработки	Содержание	8/6		
	1. Основы Kotlin: переменные, типы данных, строки, массивы. Null safety.	2	ПК 1.2, ОК 01	З 1.2.1, З 1.2.2, З 1.2.3, З 1.3.1, З 1.3.2, Зо 01.01, Зо 09.01
	2. Управляющие конструкции (if, when, loops). Функции: объявление, параметры, возвращаемые значения.	2	ПК 1.2, ПК 1.3	З 1.2.1, З 1.2.2, З 1.2.3, З 1.3.1, З 1.3.2, Зо 01.01, Зо 09.01
	3. Объектно-ориентированное программирование в	2	ПК 1.2, ОК 09	З 1.2.1, З 1.2.2, З

	Kotlin: классы, свойства, наследование, интерфейсы.			1.2.3, 3 1.3.1, 3 1.3.2, 3о 01.01, 3о 09.01
	4. Лямбда-выражения, функции высшего порядка. Работа с коллекциями (List, Map, Set).	2	ПК 1.2	3 1.2.1, 3 1.2.2, 3 1.2.3, 3 1.3.1, 3 1.3.2, 3о 01.01, 3о 09.01
	В том числе лабораторных занятий	6/6		
	Лабораторное занятие №23. Основы Kotlin. Написание функций, работа с коллекциями. Решение практических задач (сортировка, фильтрация, поиск).	6 / 6	ПК 1.2, ПК 1.3, ОК 01	У 1.2.1, У 1.2.2, У 1.2.3, У 1.3.1, У 1.3.2, Уо 01.01
Тема 2.3. Пользовательский интерфейс в Android	Содержание	10/8		
	1. XML-разметка. Типы Layout (LinearLayout, ConstraintLayout, FrameLayout, RelativeLayout).	2	ПК 1.2, ПК 1.3	3 1.2.1, 3 1.2.2, 3 1.2.3, 3 1.3.1, 3 1.3.2, 3 1.5.1, 3 1.5.2, 3 1.5.3, 3о 02.01, 3о 05.01
	2. UI-компоненты: TextView, EditText, Button, ImageView, ProgressBar. Обработчики событий.	2	ПК 1.2, ОК 05	3 1.2.1, 3 1.2.2, 3 1.2.3, 3 1.3.1, 3 1.3.2, 3 1.5.1, 3 1.5.2, 3 1.5.3, 3о 02.01, 3о 05.01
	3. Адаптеры и работа со списками: ListView, RecyclerView. ViewHolder.	2	ПК 1.2, ПК 1.5	3 1.2.1, 3 1.2.2, 3 1.2.3, 3 1.3.1, 3 1.3.2, 3 1.5.1, 3 1.5.2, 3 1.5.3, 3о 02.01, 3о 05.01
	4. Material Design: темы, стили, цвета. Анимации и переходы.	2	ПК 1.3, ОК 02	3 1.2.1, 3 1.2.2, 3 1.2.3, 3 1.3.1, 3 1.3.2, 3 1.5.1, 3 1.5.2, 3 1.5.3, 3о 02.01, 3о 05.01
	5. Навигация в приложении: Intent (явный и	2	ПК 1.2, ПК 1.4	3 1.2.1, 3 1.2.2, 3

	неявный). Передача данных между Activity. Fragment и Navigation Component.			1.2.3, 3 1.3.1, 3 1.3.2, 3 1.5.1, 3 1.5.2, 3 1.5.3, 3o 02.01, 3o 05.01
	В том числе лабораторных занятий	8/8		
	Лабораторное занятие №24. Разработка пользовательского интерфейса. Создание экрана логина с валидацией полей. Навигация между экранами (Intent).	8 / 8	ПК 1.2, ПК 1.3, ПК 1.5	У 1.2.1, У 1.2.2, У 1.2.3, У 1.3.1, У 1.3.2, У 1.5.1, У 1.5.2, У 1.5.3, Уo 02.01
	Лабораторное занятие №25. Работа со списками (RecyclerView). Отображение списка данных. Кастомизация элементов списка. Добавление обработчиков нажатий.	6 / 6	ПК 1.2, ПК 1.5, ОК 02	У 1.2.1, У 1.2.2, У 1.2.3, У 1.3.1, У 1.3.2, У 1.5.1, У 1.5.2, У 1.5.3, Уo 02.01
Тема 2.4. Работа с данными и сетью	Содержание	8/10		
	1. SharedPreferences: сохранение настроек и простых данных (ключ-значение).	2	ПК 1.2	3 1.2.1, 3 1.2.2, 3 1.2.3, 3 1.5.1, 3 1.5.2, 3 1.5.3, 3 1.6.1, 3 1.6.2, 3 1.6.3, 3 1.6.4, 3o 02.01
	2. Работа с файловой системой устройства: чтение, запись, внешнее хранилище.	2	ПК 1.2, ПК 1.5	3 1.2.1, 3 1.2.2, 3 1.2.3, 3 1.5.1, 3 1.5.2, 3 1.5.3, 3 1.6.1, 3 1.6.2, 3 1.6.3, 3 1.6.4, 3o 02.01
	3. Room Database: ORM для SQLite. Создание Entity, DAO, Database.	2	ПК 1.2, ПК 1.6	3 1.2.1, 3 1.2.2, 3 1.2.3, 3 1.5.1, 3 1.5.2, 3 1.5.3, 3 1.6.1, 3 1.6.2, 3 1.6.3, 3 1.6.4, 3o 02.01

	4. HTTP-запросы: библиотеки Retrofit и OkHttp. Работа с REST API. Обработка ответов (JSON).	2	ПК 1.2, ОК 02	З 1.2.1, З 1.2.2, З 1.2.3, З 1.5.1, З 1.5.2, З 1.5.3, З 1.6.1, З 1.6.2, З 1.6.3, З 1.6.4, З 02.01
	В том числе лабораторных занятий	10/10		
	Лабораторное занятие №26. Работа с сетью и API. Загрузка данных с открытого REST API (Retrofit). Отображение загруженных данных в RecyclerView. Обработка ошибок сети.	10 / 10	ПК 1.2, ПК 1.5, ПК 1.6, ОК 02	У 1.2.1, У 1.2.2, У 1.2.3, У 1.5.1, У 1.5.2, У 1.5.3, У 1.6.1, У 1.6.2, У 1.6.3, У 1.6.4, У 02.01
Тема 2.5. Обзор инструментов ИИ для мобильных платформ	Содержание	6		
	1. ML Kit от Google: возможности (распознавание текста, лиц, классификация), настройка, интеграция.	2	ПК 1.2, ОК 02	З 1.2.1, З 1.2.2, З 1.2.3, З 02.01, З 09.01
	2. Core ML от Apple (обзор). Firebase ML. TensorFlow Lite для on-device inference.	2	ПК 1.2, ОК 09	З 1.2.1, З 1.2.2, З 1.2.3, З 02.01, З 09.01
	3. ONNX Runtime для кроссплатформенной разработки.	2	ПК 1.2	З 1.2.1, З 1.2.2, З 1.2.3, З 02.01, З 09.01
	В том числе лабораторных занятий	6/6		
	Лабораторное занятие №27. Настройка ML Kit. Подключение библиотек ML Kit. Тестирование готовых моделей (распознавание текста, классификация изображений).	6 / 6	ПК 1.2, ПК 1.5, ОК 02	У 1.2.1, У 1.2.2, У 1.2.3, У 1.5.1, У 1.5.2, У 1.5.3, У 02.01
Тема 2.6. Распознавание изображений и текста	Содержание	8		
	1. Распознавание текста (OCR) с помощью ML Kit: устройственная и облачная модели.	2	ПК 1.2, ПК 1.6	З 1.2.1, З 1.2.2, З 1.2.3, З 1.6.1, З 1.6.2, З 1.6.3, З 1.6.4, З 1.7.1, З

				1.7.2, 3 1.7.3, 3o 01.01
	2. Распознавание лиц и эмоций (Face Detection).	2	ПК 1.2, ОК 01	3 1.2.1, 3 1.2.2, 3 1.2.3, 3 1.6.1, 3 1.6.2, 3 1.6.3, 3 1.6.4, 3 1.7.1, 3 1.7.2, 3 1.7.3, 3o 01.01
	3. Классификация изображений (Image Labeling). Кастомизация меток.	2	ПК 1.2, ПК 1.7	3 1.2.1, 3 1.2.2, 3 1.2.3, 3 1.6.1, 3 1.6.2, 3 1.6.3, 3 1.6.4, 3 1.7.1, 3 1.7.2, 3 1.7.3, 3o 01.01
	4. Детекция объектов на изображении (Object Detection). Ограничивающие рамки.	2	ПК 1.2	3 1.2.1, 3 1.2.2, 3 1.2.3, 3 1.6.1, 3 1.6.2, 3 1.6.3, 3 1.6.4, 3 1.7.1, 3 1.7.2, 3 1.7.3, 3o 01.01
	В том числе лабораторных занятий	12/12		
	Лабораторное занятие №28. Распознавание текста и изображений. Реализация OCR-сканера с камеры и из галереи. Классификация изображений с камеры.	12 / 12	ПК 1.2, ПК 1.5, ПК 1.6, ОК 01	У 1.2.1, У 1.2.2, У 1.2.3, У 1.5.1, У 1.5.2, У 1.5.3, У 1.6.1, У 1.6.2, У 1.6.3, У 1.6.4, У 1.7.1, У 1.7.2, У 1.7.3, Уo 01.01
Тема 2.7. Интеграция облачных ИИ-сервисов (вариативная часть)	Содержание	8/14		
	1. Работа с Firebase ML (облачные модели). Аутентификация и настройка.	2	ПК 1.2, ОК 02	3 1.2.1, 3 1.2.2, 3 1.2.3, 3 1.6.1, 3 1.6.2, 3 1.6.3, 3 1.6.4, 3o 02.01, 3o 09.01

	2. Интеграция с Yandex Vision API (отечественное решение).	2	ПК 1.2, ОК 09	З 1.2.1, З 1.2.2, З 1.2.3, З 1.6.1, З 1.6.2, З 1.6.3, З 1.6.4, Зо 02.01, Зо 09.01
	3. Распознавание голоса и синтез речи (Speech-to-Text, Text-to-Speech).	2	ПК 1.2, ПК 1.6	З 1.2.1, З 1.2.2, З 1.2.3, З 1.6.1, З 1.6.2, З 1.6.3, З 1.6.4, Зо 02.01, Зо 09.01
	4. Перевод текста (Translation API).	2	ПК 1.2	З 1.2.1, З 1.2.2, З 1.2.3, З 1.6.1, З 1.6.2, З 1.6.3, З 1.6.4, Зо 02.01, Зо 09.01
	В том числе лабораторных занятий	14/14		
	Лабораторное занятие №29. Голосовой ассистент. Реализация Speech-to-Text и Text-to-Speech. Интеграция с Yandex Vision API.	14 / 14	ПК 1.2, ПК 1.5, ПК 1.7, ОК 02	У 1.2.1, У 1.2.2, У 1.2.3, У 1.5.1, У 1.5.2, У 1.5.3, У 1.7.1, У 1.7.2, У 1.7.3, Уо 02.01
Тема 2.8. Оптимизация моделей для мобильных платформ (вариативная часть)	Содержание	8/12		
	1. Конвертация моделей в TensorFlow Lite (из SavedModel, Keras).	2	ПК 1.2, ПК 1.3	З 1.2.1, З 1.2.2, З 1.2.3, З 1.3.1, З 1.3.2, З 1.5.1, З 1.5.2, З 1.5.3, Зо 01.01, Зо 02.01
	2. Квантование моделей (FP16, INT8, динамическое). Сжатие модели.	2	ПК 1.2, ОК 01	З 1.2.1, З 1.2.2, З 1.2.3, З 1.3.1, З 1.3.2, З 1.5.1, З 1.5.2, З 1.5.3, Зо 01.01, Зо 02.01
	3. Оптимизация размера модели и скорости инференса на устройстве.	2	ПК 1.5, ОК 02	З 1.2.1, З 1.2.2, З 1.2.3, З 1.3.1, З 1.3.2, З 1.5.1, З

				1.5.2, 3 1.5.3, 3o 01.01, 3o 02.01
	4. Использование GPU Delegation в TensorFlow Lite для ускорения.	2	ПК 1.2	3 1.2.1, 3 1.2.2, 3 1.2.3, 3 1.3.1, 3 1.3.2, 3 1.5.1, 3 1.5.2, 3 1.5.3, 3o 01.01, 3o 02.01
	В том числе лабораторных занятий	12/12		
	Лабораторное занятие №30. Оптимизация модели. Конвертация модели в TFLite. Квантование (INT8). Сравнение производительности (размер, скорость инференса).	12 / 12	ПК 1.2, ПК 1.5, ОК 01	У 1.2.1, У 1.2.2, У 1.2.3, У 1.5.1, У 1.5.2, У 1.5.3, Уо 01.01
Тема 2.9. Работа с сенсорами и камерой	Содержание	6/14		
	1. Доступ к камере устройства: CameraX (LifecycleOwner, ImageAnalysis).	2	ПК 1.2, ПК 1.5	3 1.2.1, 3 1.2.2, 3 1.2.3, 3 1.5.1, 3 1.5.2, 3 1.5.3, 3 1.7.1, 3 1.7.2, 3 1.7.3
	2. Работа с акселерометром, гироскопом, датчиком приближения (SensorManager).	2	ПК 1.2	3 1.2.1, 3 1.2.2, 3 1.2.3, 3 1.5.1, 3 1.5.2, 3 1.5.3, 3 1.7.1, 3 1.7.2, 3 1.7.3
	3. Интеграция ИИ с сенсорами (распознавание жестов, определение положения устройства).	2	ПК 1.2, ПК 1.7	3 1.2.1, 3 1.2.2, 3 1.2.3, 3 1.5.1, 3 1.5.2, 3 1.5.3, 3 1.7.1, 3 1.7.2, 3 1.7.3
	В том числе лабораторных занятий	14/14		
	Лабораторное занятие №31. Распознавание объектов с камеры в реальном времени. Интеграция камеры (CameraX) с моделью детекции объектов (TFLite). Отображение ограничивающих рамок на предпросмотре.	14 / 14	ПК 1.2, ПК 1.5, ПК 1.6, ОК 02	Уд 2, Уд 5, Уд 6, Уо 02.01

Тема 2.10. Отладка мобильных приложений	Содержание	6/6		
	1. Использование отладчика Android Studio: breakpoints, step over/into/out, watches.	2	ПК 1.5, ОК 02	З 1.5.1, З 1.5.2, З 1.5.3, Зо 01.01, Зо 02.01
	2. Логирование (Logcat). Уровни логирования (debug, info, error). Фильтрация логов.	2	ПК 1.5	З 1.5.1, З 1.5.2, З 1.5.3, Зо 01.01, Зо 02.01
	3. Профилирование производительности (Android Profiler): CPU, Memory, Network, Energy.	2	ПК 1.5, ОК 01	З 1.5.1, З 1.5.2, З 1.5.3, Зо 01.01, Зо 02.01
	В том числе лабораторных занятий	6/6		
	Лабораторное занятие №32. Отладка приложения. Поиск и исправление ошибок с помощью отладчика. Анализ логов Logcat.	6 / 6	ПК 1.5, ОК 01, ОК 02	У 1.5.1, У 1.5.2, У 1.5.3, Уо 01.01, Уо 02.01
Тема 2.11. Тестирование мобильных приложений	Содержание	6/8		
	1. Unit-тестирование (JUnit). Аннотации (@Test, @Before, @After). Assertions.	2	ПК 1.6, ОК 04	З 1.6.1, З 1.6.2, З 1.6.3, З 1.6.4, З 1.7.1, З 1.7.2, З 1.7.3, Зо 04.01
	2. UI-тестирование (Espresso). onView, perform, check.	2	ПК 1.6	З 1.6.1, З 1.6.2, З 1.6.3, З 1.6.4, З 1.7.1, З 1.7.2, З 1.7.3, Зо 04.01
	3. Интеграционное тестирование. Mock-объекты (Mockito).	2	ПК 1.6, ПК 1.7	З 1.6.1, З 1.6.2, З 1.6.3, З 1.6.4, З 1.7.1, З 1.7.2, З 1.7.3, Зо 04.01
	В том числе лабораторных занятий	8/8		
	Лабораторное занятие №33. Unit-тестирование. Написание тестов для критических функций приложения (валидация, вычисления). Запуск и анализ результатов.	8 / 8	ПК 1.6, ПК 1.7, ОК 04	У 1.6.1, У 1.6.2, У 1.6.3, У 1.6.4, У 1.7.1, У 1.7.2, У 1.7.3, Уо 04.01
Тема 2.12. Тестирование ИИ-компонентов (вариативная)	Содержание	6/8		
	1. Тестирование моделей на мобильном устройстве:	2	ПК 1.6, ПК 1.7	З 1.6.1, З 1.6.2, З

часть)	подготовка тестовых данных.			1.6.3, 3 1.6.4, 3 1.7.1, 3 1.7.2, 3 1.7.3, 3о 01.01, 3о 02.01
	2. Проверка качества распознавания при разных условиях (освещение, угол обзора).	2	ПК 1.6, ОК 01	3 1.6.1, 3 1.6.2, 3 1.6.3, 3 1.6.4, 3 1.7.1, 3 1.7.2, 3 1.7.3, 3о 01.01, 3о 02.01
	3. Автоматизация тестирования ИИ-функций (скрипты для Android Test Orchestrator).	2	ПК 1.7, ОК 02	3 1.6.1, 3 1.6.2, 3 1.6.3, 3 1.6.4, 3 1.7.1, 3 1.7.2, 3 1.7.3, 3о 01.01, 3о 02.01
	В том числе лабораторных занятий	8/8		
	Лабораторное занятие №34. Тестирование ИИ-компонентов. Разработка тестовых сценариев для проверки качества распознавания. Оценка точности при разных условиях.	8 / 8	ПК 1.6, ПК 1.7, ОК 01	У 1.6.1, У 1.6.2, У 1.6.3, У 1.6.4, У 1.7.1, У 1.7.2, У 1.7.3, Уо 01.01
Тема 2.13. Разработка итогового проекта	Содержание	8/22		
	1. Планирование проекта. Определение требований. Составление технического задания.	2	ПК 1.1, ОК 04	3 1.1.1, 3 1.1.2, 3 1.2.1, 3 1.2.2, 3 1.2.3, 3 1.3.1, 3 1.3.2, 3 1.4.1, 3 1.4.2, 3о 04.01, 3о 05.01
	2. Проектирование архитектуры приложения (MVP, MVVM). Схема взаимодействия компонентов.	2	ПК 1.2, ПК 1.4	3 1.1.1, 3 1.1.2, 3 1.2.1, 3 1.2.2, 3 1.2.3, 3 1.3.1, 3 1.3.2, 3 1.4.1, 3 1.4.2, 3о 04.01, 3о 05.01
	3. Распределение задач в команде (Git, GitHub). Правила оформления pull requests.	2	ПК 1.4, ОК 04	3 1.1.1, 3 1.1.2, 3 1.2.1, 3 1.2.2, 3 1.2.3, 3 1.3.1, 3

				1.3.2, 3 1.4.1, 3 1.4.2, 3o 04.01, 3o 05.01
	4. Оформление документации к проекту. Подготовка презентации для защиты.	2	ПК 1.3, ОК 05	3 1.1.1, 3 1.1.2, 3 1.2.1, 3 1.2.2, 3 1.2.3, 3 1.3.1, 3 1.3.2, 3 1.4.1, 3 1.4.2, 3o 04.01, 3o 05.01
	В том числе практических занятий	22/22		
	Практическое занятие №35. Разработка итогового проекта. Реализация мобильного приложения с интеграцией ИИ (распознавание изображений, голоса или текста). Полный цикл: от идеи до работающего приложения.	22 / 22	ПК 1.1-1.7, ОК 01,02,04,05,09	У 1.1.1, У 1.1.2, У 1.2.1, У 1.2.2, У 1.2.3, У 1.3.1, У 1.3.2, У 1.4.1, У 1.4.2, У 1.5.1, У 1.5.2, У 1.5.3, У 1.6.1, У 1.6.2, У 1.6.3, У 1.6.4, У 1.7.1, У 1.7.2, У 1.7.3, Уo 01.01, Уo 02.01, Уo 04.01, Уo 05.01, Уo 09.01
	Самостоятельная работа	14/0		
	Подготовка отчета по итоговому проекту. Анализ результатов. Оформление документации. Подготовка к защите проекта.	14 / 0	ПК 1.3, ПК 1.5, ОК 05	У 1.3.1, У 1.3.2, У 1.5.1, У 1.5.2, У 1.5.3, Уo 05.01
МДК.01.03 Отладка и тестирование программных модулей		152/112		
Раздел 3. Отладка и тестирование программных модулей				
Тема 3.1. Основы тестирования программного обеспечения	Содержание	24/16		
	1. Виды и уровни тестирования: модульное, интеграционное, системное, приёмочное.	12	ПК 1.5, ОК 02	3 1.5.1, 3 1.5.2, 3 1.5.3, 3o 02.01 / У 1.5.1, У 1.5.2, У 1.5.3

	2. Жизненный цикл дефекта, классификация ошибок, метрики качества тестов.	12	ПК 1.5, ОК 02	З 1.5.1, З 1.5.2, З 1.5.3, З 02.01 / У 1.5.1, У 1.5.2, У 1.5.3
	В том числе лабораторных занятий	16/16		
	Лабораторное занятие №36. Классификация дефектов и составление отчёта о тестировании	16/16	ПК 1.5	У 1.5.1, У 1.5.2, У 1.5.3
	Самостоятельная работа	4/0		
	Изучение документации по логированию, анализ типовых сценариев отказов.	4/0	ПК 1.5	У 1.5.1, У 1.5.2, У 1.5.3
Тема 3.2. Модульное и интеграционное тестирование	Содержание	32/24		
	1. Фреймворки тестирования: unittest, pytest.	16	ПК 1.6	З 1.6.1, З 1.6.2, З 1.6.3, З 1.6.4 / У 1.5.1, У 1.5.2, У 1.5.3
	2. Мокирование и стабы (unittest.mock, pytest-mock). Оценка покрытия кода (pytest-cov).	16	ПК 1.5, ОК 02	З 1.6.1, З 1.6.2, З 1.6.3, З 1.6.4 / У 1.5.1, У 1.5.2, У 1.5.3
	В том лабораторных занятий	24/24		
	Лабораторное занятие №37. Написание модульных тестов для функций обработки данных с использованием pytest.	12/12	ПК 1.6	У 1.6.1, У 1.6.2, У 1.6.3, У 1.6.4
	Лабораторное занятие №38. Интеграционное тестирование с использованием mock-объектов и заглушек внешних зависимостей.	12/12	ПК 1.6	У 1.6.1, У 1.6.2, У 1.6.3, У 1.6.4
	Самостоятельная работа	4/0		
	Разработка тестов для граничных значений, анализ отчёта о покрытии.	4/0	ПК 1.6	У 1.6.1, У 1.6.2, У 1.6.3, У 1.6.4
	Содержание	40/32		
Тема 3.3. Тестирование ИИ-моделей и нейросетевых компонентов	1. Метрики качества моделей: точность, полнота, F1-мера, ROC-AUC. Матрица ошибок.	20	ПК 1.6, 1.7	У 1.6.1, У 1.6.2, У 1.6.3, У 1.6.4, У 1.7.1, У 1.7.2, У 1.7.3
	2. Тестирование устойчивости моделей,	20	ПК 1.6, 1.7	У 1.6.1, У 1.6.2,

	аугментация данных, adversarial-тесты. Валидация и кросс-валидация.			У 1.6.3, У 1.6.4, У 1.7.1, У 1.7.2, У 1.7.3
	В том числе лабораторных занятий	32/32		
	Лабораторное занятие №39. Оценка качества модели классификации: построение матрицы ошибок и ROC-кривой.	16/16	ПК 1.6, 1.7	У 1.6.1, У 1.6.2, У 1.6.3, У 1.6.4, У 1.7.1, У 1.7.2, У 1.7.3
	Лабораторное занятие №40. Тестирование устойчивости нейросетевой модели к зашумлённым данным и выбросам.	16/16	ПК 1.6, 1.7	У 1.6.1, У 1.6.2, У 1.6.3, У 1.6.4, У 1.7.1, У 1.7.2, У 1.7.3
	Самостоятельная работа	4/0		
	Изучение документации по логированию, анализ типовых сценариев отказов.	4/0	ПК 1.6, 1.7	У 1.6.1, У 1.6.2, У 1.6.3, У 1.6.4, У 1.7.1, У 1.7.2, У 1.7.3
Тема 3.4. Автоматизация тестирования и CI/CD	Содержание	32/24		
	1. Непрерывная интеграция и доставка (CI/CD). Принципы автоматизации пайплайнов.	16	ПК 1.7	У 1.7.1, У 1.7.2, У 1.7.3
	2. Настройка GitHub Actions/GitLab CI. Генерация отчётов о покрытии и качестве кода в процессе сборки.	16	ПК 1.7	У 1.7.1, У 1.7.2, У 1.7.3
	В том числе лабораторных занятий	24/24		
	Лабораторное занятие №41. Настройка CI/CD-пайплайна для автоматического запуска тестов при push в репозиторий.	12/12	ПК 1.7	У 1.7.1, У 1.7.2, У 1.7.3
	Лабораторное занятие №42. Анализ покрытия кода и оптимизация тестового набора.	12/12	ПК 1.7	У 1.7.1, У 1.7.2, У 1.7.3
	Самостоятельная работа	4/0		
	Изучение синтаксиса YAML-конфигураций, анализ логов сборок CI.	4/0	ПК 1.6, 1.7	У 1.7.1, У 1.7.2, У 1.7.3
Тема 3.5. Разработка тестовой документации и сценариев	Содержание	24/16		
	1. Структура тест-плана, тест-кейса, чек-листа. Методы тест-дизайна (эквивалентное разбиение,	12	ПК 1.7	У 1.7.1, У 1.7.2, У 1.7.3

	граничные значения).			
	2. Применение промт-инжиниринга для генерации тестовых данных и автоматизации написания сценариев.	12	ПК 1.7	У 1.7.1, У 1.7.2, У 1.7.3
	В том числе лабораторных занятий	16/16		
	Лабораторное занятие №43. Разработка тестовых сценариев и чек-листов для ИИ-модуля с использованием ИИ-ассистента.	16/16	ПК 1.7	У 1.7.1, У 1.7.2, У 1.7.3
	Лабораторное занятие №44. Оформление итогового отчёта о тестировании.		ПК 1.7	У 1.7.1, У 1.7.2, У 1.7.3
	Самостоятельная работа	4/0		
	Оформление итогового отчёта о тестировании, анализ дефектов.	4/0	ПК 1.6, 1.7	У 1.7.1, У 1.7.2, У 1.7.3
Учебная практика раздела 3 Виды работ 1. Разработка и интеграция ИИ-модулей в коммерческие проекты. 2. Оптимизация моделей под production-среду. 3. Нагрузочное тестирование и профилирование. 4. Код-ревью и менторинг. 5. Подготовка технической документации и user-manuals. 6. Участие в agile-циклах разработки.		252/252	ПК 1.1-1.7, ОК 01,02,04,05,09	У 1.1.1–1.7.2, 3 1.1.1–1.7.2, Уо 01.01–09.05, 3о 01.01–09.05
Производственная практика Виды работ		72/72	ПК 1.1-1.7, ОК 01,02,04,05,09	У 1.1.1–1.7.2, 3 1.1.1–1.7.2, Уо 01.01–09.05, 3о 01.01–09.05
Промежуточная аттестация Экзамен по профессиональному модулю		12/0	ПК 1.1–1.7, ОК 01, 02, 04, 05, 09	Н 1.1.1–1.7.2, У 1.1.1–1.7.2, 3 1.1.1–1.7.2, Уо 01.01–09.05, 3о 01.01–09.05
Всего		1400/890		

2.3 Перечень практических и лабораторных занятий

Номенклатура практических и лабораторных занятий должна обеспечивать освоение названных в разделе 1.2 рабочей программы умений.

Темы лабораторных и практических занятий	Содержание (краткое описание)	Специализированное оборудование, технические средства, программное обеспечение
МДК 01.01 Разработка программных модулей в системах искусственного интеллекта		
Лабораторные занятия		
Лабораторное занятие №1. Анализ технического задания на разработку модуля ИИ	Формирует умение изучать структуру технического задания, выделять функциональные требования, входные/выходные данные и ограничения, составлять план разработки модуля	ПК (Процессор: Intel(R) Core(TM) i5-9400 CPU @2.90GHz 2.90 GHz/RAM 8,00 Gb /HDD. Монитор IIYAMA 24 дюйма Монитор: PHILIPS 243V). Текстовый редактор, примеры ТЗ, средства совместной работы.
Лабораторное занятие №2. Разработка скрипта для обработки данных с использованием циклов и условных операторов	Формирует умение писать программный код на Python с использованием списков, условных конструкций и циклов для фильтрации, агрегации и трансформации данных	ПК (Процессор: Intel(R) Core(TM) i5-9400 CPU @2.90GHz 2.90 GHz/RAM 8,00 Gb /HDD. Монитор IIYAMA 24 дюйма Монитор: PHILIPS 243V). Python 3.10+, PyCharm Community / VS Code, Git Bash.
Лабораторное занятие №3. Создание функций для модульной обработки данных	Формирует умение определять функции с параметрами и возвратом значений, реализовывать модульную архитектуру, обрабатывать исключения (try-except), тестировать функции	ПК (Процессор: Intel(R) Core(TM) i5-9400 CPU @2.90GHz 2.90 GHz/RAM 8,00 Gb /HDD. Монитор IIYAMA 24 дюйма Монитор: PHILIPS 243V). Python 3.10+, PyCharm / VS Code, интерпретатор Python.
Лабораторное занятие №4. Загрузка и предобработка данных с помощью Pandas	Формирует умение читать CSV/Excel файлы, анализировать структуру DataFrame, работать с пропусками, фильтровать строки, сохранять очищенные	ПК (Процессор: Intel(R) Core(TM) i5-9400 CPU @2.90GHz 2.90 GHz/RAM 8,00 Gb /HDD. Монитор IIYAMA 24 дюйма Монитор: PHILIPS 243V). Python 3.10+, Pandas, NumPy,

	данные в новый файл	Jupyter Notebook, тестовые датасеты.
Лабораторное занятие №5. Векторизованные вычисления с NumPy	Формирует умение создавать многомерные массивы, выполнять индексацию и срезы, производить поэлементные арифметические операции, выполнять агрегацию (сумма, среднее, дисперсия)	ПК (Процессор: Intel(R) Core(TM) i5-9400 CPU @2.90GHz 2.90 GHz/RAM 8,00 Gb /HDD. Монитор IIYAMA 24 дюйма Монитор: PHILIPS 243V). Python 3.10+, NumPy, Jupyter Notebook, VS Code.
Лабораторное занятие №6. Построение графиков распределения данных	Формирует умение визуализировать данные: строить гистограммы, boxplot, scatter plot, настраивать оси, легенду, стили и сохранять графики в файлы	ПК (Процессор: Intel(R) Core(TM) i5-9400 CPU @2.90GHz 2.90 GHz/RAM 8,00 Gb /HDD. Монитор IIYAMA 24 дюйма Монитор: PHILIPS 243V). Python 3.10+, Matplotlib, Seaborn, Pandas, Jupyter Notebook.
Лабораторное занятие №7. Реализация линейной регрессии на scikit-learn	Формирует умение разделять выборку (train/test), обучать модель LinearRegression, выполнять прогнозирование, оценивать качество модели (MSE, R ²)	ПК (Процессор: Intel(R) Core(TM) i5-9400 CPU @2.90GHz 2.90 GHz/RAM 8,00 Gb /HDD. Монитор IIYAMA 24 дюйма Монитор: PHILIPS 243V). Python 3.10+, scikit-learn, Pandas, NumPy, Jupyter Notebook.
Лабораторное занятие №8. Классификация с помощью метода k-ближайших соседей	Формирует умение загружать набор Iris, обучать KNN (k=5), предсказывать классы, выводить матрицу ошибок, точность и отчёт классификации	ПК (Процессор: Intel(R) Core(TM) i5-9400 CPU @2.90GHz 2.90 GHz/RAM 8,00 Gb /HDD. Монитор IIYAMA 24 дюйма Монитор: PHILIPS 243V). Python 3.10+, scikit-learn, Matplotlib, Jupyter Notebook.
Лабораторное занятие №9. Создание и обучение нейронной сети на TensorFlow/Keras	Формирует умение строить Sequential-модель, компилировать, обучать на датасете MNIST, оценивать accuracy, сохранять веса модели	ПК (Процессор: Intel(R) Core(TM) i5-9400 CPU @2.90GHz 2.90 GHz/RAM 8,00 Gb /HDD. Монитор IIYAMA 24 дюйма Монитор: PHILIPS 243V). Python 3.10+, TensorFlow, Keras, NumPy, GPU-ускорение (опционально).
Лабораторное занятие №10. Использование PyTorch для построения нейросети	Формирует умение определять класс nn.Module, выбирать loss и optimizer, реализовывать цикл обучения на CIFAR-	ПК (Процессор: Intel(R) Core(TM) i5-9400 CPU @2.90GHz 2.90 GHz/RAM 8,00 Gb /HDD. Монитор IIYAMA 24 дюйма Монитор: PHILIPS 243V). Python 3.10+, PyTorch,

	10, выполнять валидацию, сохранять состояние модели	torchvision, Jupyter Notebook.
Лабораторное занятие №11. Приведение кода к стандарту PEP8 с использованием линтеров	Формирует умение устанавливать flake8, pylint, black, выполнять автоматическое форматирование, ручное исправление замечаний, проверять соответствие стилю	ПК (Процессор: Intel(R) Core(TM) i5-9400 CPU @2.90GHz 2.90 GHz/RAM 8,00 Gb /HDD. Монитор IIYAMA 24 дюйма Монитор: PHILIPS 243V). Python 3.10+, flake8, pylint, black, PyCharm/VS Code.
Лабораторное занятие №12. Написание документации к разработанному модулю	Формирует умение добавлять docstring к функциям и классам, настраивать conf.py, генерировать HTML/PDF документацию с помощью Sphinx	ПК (Процессор: Intel(R) Core(TM) i5-9400 CPU @2.90GHz 2.90 GHz/RAM 8,00 Gb /HDD. Монитор IIYAMA 24 дюйма Монитор: PHILIPS 243V). Python 3.10+, Sphinx, reStructuredText, текстовый редактор.
Лабораторное занятие №13. Создание локального репозитория, коммиты, работа с ветками	Формирует умение инициализировать git init, выполнять коммиты, создавать ветку develop, выполнять слияние (merge), разрешать конфликты, создавать pull request	ПК (Процессор: Intel(R) Core(TM) i5-9400 CPU @2.90GHz 2.90 GHz/RAM 8,00 Gb /HDD. Монитор IIYAMA 24 дюйма Монитор: PHILIPS 243V). Git Bash / терминал VS Code, GitHub/GitLab, Git.
Лабораторное занятие №14. Использование отладчика в PyCharm для поиска ошибок	Формирует умение устанавливать брейкпоинты, выполнять пошаговое выполнение (Step Over/Into), просматривать переменные в Watches, анализировать стек вызовов	ПК (Процессор: Intel(R) Core(TM) i5-9400 CPU @2.90GHz 2.90 GHz/RAM 8,00 Gb /HDD. Монитор IIYAMA 24 дюйма Монитор: PHILIPS 243V). PyCharm Professional/Community, Python 3.10+, отладочные скрипты.
Лабораторное занятие №15. Внедрение логирования в модуль ИИ	Формирует умение настраивать logging (уровни DEBUG/INFO/ERROR), форматировать записи, выводить в консоль и файл, настраивать ротацию логов	ПК (Процессор: Intel(R) Core(TM) i5-9400 CPU @2.90GHz 2.90 GHz/RAM 8,00 Gb /HDD. Монитор IIYAMA 24 дюйма Монитор: PHILIPS 243V). Python 3.10+, модуль logging, PyCharm/VS Code.
Лабораторное занятие №16. Разработка модульных тестов	Формирует умение писать тесты с использованием unittest, использовать	ПК (Процессор: Intel(R) Core(TM) i5-9400 CPU @2.90GHz 2.90 GHz/RAM 8,00 Gb /HDD. Монитор IIYAMA 24 дюйма

для функций обработки данных (unittest)	assertEqual, assertTrue, запускать тесты, анализировать отчёт о прохождении	Монитор: PHILIPS 243V). Python 3.10+, unittest, PyCharm/VS Code.
Лабораторное занятие №17. Написание тестов для нейросетевой модели (pytest)	Формирует умение использовать pytest, фикстуры, параметризацию, проверять точность модели, генерировать отчёт о покрытии (pytest-cov)	ПК (Процессор: Intel(R) Core(TM) i5-9400 CPU @2.90GHz 2.90 GHz/RAM 8,00 Gb /HDD. Монитор IIYAMA 24 дюйма Монитор: PHILIPS 243V). Python 3.10+, pytest, pytest-cov, PyCharm/VS Code.
Лабораторное занятие №18. Разработка тестового сценария для модуля классификации	Формирует умение составлять тест-кейсы (предусловия, шаги, ожидаемый результат), выполнять ручное тестирование, фиксировать дефекты, оформлять сценарий	ПК (Процессор: Intel(R) Core(TM) i5-9400 CPU @2.90GHz 2.90 GHz/RAM 8,00 Gb /HDD. Монитор IIYAMA 24 дюйма Монитор: PHILIPS 243V). Текстовый редактор, шаблоны тест-кейсов, документация.
Лабораторное занятие №19. Автоматизация запуска тестов с использованием CI (GitHub Actions)	Формирует умение настраивать workflow в GitHub Actions, запускать pytest при push, просматривать логи CI, интегрировать покрытие кода в процесс сборки	ПК (Процессор: Intel(R) Core(TM) i5-9400 CPU @2.90GHz 2.90 GHz/RAM 8,00 Gb /HDD. Монитор IIYAMA 24 дюйма Монитор: PHILIPS 243V). GitHub, Git, pytest, GitHub Actions, VS Code.
Лабораторное занятие №20. Разработка тестового сценария для модуля классификации	Формирует умение составлять тест-кейсы (предусловия, шаги, ожидаемый результат), выполнять ручное тестирование, фиксировать дефекты, оформлять сценарий	ПК (Процессор: Intel(R) Core(TM) i5-9400 CPU @2.90GHz 2.90 GHz/RAM 8,00 Gb /HDD. Монитор IIYAMA 24 дюйма Монитор: PHILIPS 243V). Текстовый редактор, шаблоны тест-кейсов, документация.
Лабораторное занятие №21. Автоматизация запуска тестов с использованием CI (GitHub Actions)	Формирует умение настраивать workflow в GitHub Actions, запускать pytest при push, просматривать логи CI, интегрировать покрытие кода в процесс сборки	ПК (Процессор: Intel(R) Core(TM) i5-9400 CPU @2.90GHz 2.90 GHz/RAM 8,00 Gb /HDD. Монитор IIYAMA 24 дюйма Монитор: PHILIPS 243V). GitHub, Git, pytest, GitHub Actions, VS Code.
МДК 01.02 Разработка мобильных приложений с поддержкой искусственного интеллекта		

Лабораторные занятия		
Лабораторное занятие №22. Создание первого Android-приложения	Формирует умение настраивать среду разработки, создавать интерфейс и обрабатывать события нажатия кнопок	ПК, Android Studio, эмулятор Android
Лабораторное занятие №23. Основы Kotlin	Формирует умение писать функции и работать с коллекциями (List, Map, Set) на языке Kotlin	ПК, IntelliJ IDEA / Android Studio
Лабораторное занятие №24. Разработка пользовательского интерфейса	Формирует умение создавать экран логина с валидацией полей и навигацию между экранами (Intent)	ПК, Android Studio
Лабораторное занятие №25. Работа со списками (RecyclerView)	Формирует умение отображать список данных, кастомизировать элементы и обрабатывать нажатия	ПК, Android Studio
Лабораторное занятие №26. Работа с сетью и API	Формирует умение загружать данные с REST API с использованием Retrofit и отображать их в RecyclerView	ПК, Android Studio, доступ в интернет
Лабораторное занятие №27. Настройка ML Kit	Формирует умение подключать библиотеки ML Kit и тестировать готовые модели распознавания	ПК, Android Studio, ML Kit
Лабораторное занятие №28. Распознавание текста и изображений	Формирует умение реализовывать OCR-сканер и классифицировать изображения с камеры	ПК, Android Studio, ML Kit, камера
Лабораторное занятие №29. Голосовой ассистент	Формирует умение реализовывать Speech-to-Text, Text-to-Speech и интегрироваться с облачными ИИ-сервисами	ПК, Android Studio, микрофон, динамики, доступ в интернет
Лабораторное занятие №30. Оптимизация модели	Формирует умение конвертировать модель в TensorFlow Lite, выполнять квантование и сравнивать производительность	ПК, Android Studio, TensorFlow Lite
Лабораторное занятие №31. Распознавание объектов с камеры в реальном времени	Формирует умение интегрировать камеру (CameraX) с моделью детекции объектов и отображать ограничивающие рамки	ПК, Android Studio, TensorFlow Lite, камера
Лабораторное занятие №32. Отладка приложения	Формирует умение искать и исправлять ошибки с помощью отладчика Android Studio и анализировать логи Logcat	ПК, Android Studio

Лабораторное занятие №33. Unit-тестирование	Формирует умение писать Unit-тесты с использованием JUnit и анализировать результаты тестирования	ПК, Android Studio, JUnit
Лабораторное занятие №34. Тестирование ИИ-компонентов	Формирует умение разрабатывать тестовые сценарии для проверки качества распознавания в различных условиях	ПК, Android Studio, камера, тестовые изображения
Лабораторное занятие №35. Планирование и проектирование итогового проекта	Формирует умение составлять техническое задание, проектировать архитектуру и распределять задачи в команде	ПК, Android Studio, Git, GitHub
МДК.01.03 Отладка и тестирование программных модулей		
Лабораторные занятия		
Лабораторное занятие №36 Диагностика ошибок с использованием отладчика и внедрение логирования в ИИ-модуль	Формирует умение использовать отладчики (PyCharm Debugger, VS Code, pdb), устанавливать точки останова, выполнять пошаговую отладку, настраивать многоуровневое логирование	ПК, Python 3.10+, PyCharm/VS Code, pdb, модуль logging
Лабораторное занятие №37 Написание модульных тестов для функций обработки данных с использованием pytest	Формирует умение создавать модульные тесты с использованием pytest, работать с фикстурами, параметризацией тестов, assertions	ПК, Python 3.10+, pytest, pytest-mock, PyCharm/VS Code
Лабораторное занятие №38 Интеграционное тестирование с использованием mock-объектов и заглушек внешних зависимостей	Формирует умение мокировать внешние API, базы данных, файловую систему с использованием unittest.mock и pytest-mock	ПК, Python 3.10+, pytest, pytest-mock, requests-mock, PyCharm/VS Code
Лабораторное занятие №39 Оценка качества модели классификации: построение матрицы ошибок и ROC-кривой	Формирует умение рассчитывать метрики качества (accuracy, precision, recall, F1-score), строить confusion matrix, ROC-кривую и рассчитывать AUC	ПК, Python 3.10+, scikit-learn, matplotlib, seaborn, Jupyter Notebook
Лабораторное занятие №40 Тестирование устойчивости нейросетевой модели к зашумлённым данным и выбросам	Формирует умение проводить adversarial-тестирование, аугментацию данных, проверять устойчивость моделей к аномалиям	ПК, Python 3.10+, TensorFlow/PyTorch, numpy, adversarial-robustness-toolbox

Лабораторное занятие №41 Формирование умений настройки GitHub Actions/GitLab CI, создания workflow файлов, автоматизации запуска тестов и проверки покрытия	Формирует умение настраивать GitHub Actions/GitLab CI, создавать workflow файлы, автоматизировать запуск тестов и проверять покрытие	ПК, GitHub/GitLab, Git, pytest, GitHub Actions, YAML, VS Code
Лабораторное занятие №42 Анализ покрытия кода и оптимизация тестового набора	Формирует умение оценивать покрытие кода тестами с помощью pytest-cov/Coverage.py, выявлять непокрытые участки, оптимизировать тесты	ПК, Python 3.10+, pytest-cov, Coverage.py, HTML-отчёты о покрытии
Лабораторное занятие №43 Разработка тестовых сценариев и чек-листов для ИИ-модуля с использованием ИИ-ассистента	Формирует умение проектировать тест-кейсы, создавать чек-листы, использовать промт-инжиниринг для генерации тестовых данных	ПК, текстовый редактор, шаблоны тест-кейсов, LLM (ChatGPT/Claude), доступ к интернету
Лабораторное занятие №44 Оформление итогового отчёта о тестировании	Формирует умение документировать результаты тестирования, анализировать дефекты, подготавливать отчётность с визуализацией метрик	ПК, текстовый редактор, средства визуализации (matplotlib, seaborn), шаблоны отчётности
Практические занятия		
Практическое занятие №1. Разработка приложения с сетевым взаимодействием (вариативная часть)	Формирует умение разрабатывать полноценное приложение с загрузкой данных из сети, их обработкой и отображением	ПК, Android Studio, доступ в интернет
Практическое занятие №2. Разработка приложения с распознаванием изображений (вариативная часть)	Формирует умение разрабатывать приложение с функцией распознавания объектов в реальном времени с камеры	ПК, Android Studio, камера
Практическое занятие №3. Разработка тестового набора для ИИ-функций (вариативная часть)	Формирует умение создавать автоматизированные тесты для проверки качества работы ИИ-компонентов	ПК, Android Studio, Android Test Orchestrator
Практическое занятие №4. Разработка итогового проекта	Формирует умение реализовывать полноценное мобильное приложение с интеграцией ИИ (распознавание	ПК, Android Studio, доступ в интернет, камера/микрофон

	изображений, голоса или текста)	
--	---------------------------------	--

3 УСЛОВИЯ РЕАЛИЗАЦИИ РАБОЧЕЙ ПРОГРАММЫ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

3.1 Материально-техническое обеспечение

Для реализации программы учебной дисциплины предусмотрены следующие специальные помещения:

Лаборатория «Программного обеспечения и сопровождения компьютерных и информационных систем», оснащенная в соответствии с приложением 3 образовательной программы.

Лаборатория «По обслуживанию и управлению беспилотными летательными аппаратами и сопровождению компьютерных систем», оснащенная в соответствии с приложением 3 образовательной программы.

3.2 Учебно-методическое и информационное обеспечение реализации программы

Основные источники:

1. Маккинли, У. Python и анализ данных / У. Маккинли. — 2-е изд. — М. : ДМК Пресс, 2023. — 544 с. — ISBN 978-5-97060-987-1. — URL: <https://dmk-press.ru/catalog/programming/python/python-and-data-analysis/> (посещался 20.04.2026)
2. Раска, Д. Введение в машинное обучение с помощью Python / Д. Раска, С. Гвидо. — СПб. : Питер, 2022. — 384 с. — ISBN 978-5-4461-1845-3. — URL: <https://www.piter.com/product/detail.php?ID=108567> (посещался 20.04.2026)
3. Бизли, Д. Python. Подробный справочник / Д. Бизли. — 4-е изд. — СПб. : Питер, 2023. — 2. (посещался 20.04.2026) https://rusbuk.ru/offer/python_podrobnyi_spravochniks_1
4. Гриффитс, И. Android. Программирование для профессионалов / И. Гриффитс, Д. Гриффитс. — 4-е изд. — СПб.: Питер, 2024. — 864 с. — ISBN 978-5-4461-2100-1 (посещался 20.04.2026) https://rusbuk.ru/offer/python_podrobnyi_spravochniks_1
5. Хортон, Д. Android. Полное руководство по разработке приложений / Д. Хортон. — М.: Эксмо, 2023. — 624 с. — ISBN 978-5-04-122348-7. (посещался 20.04.2026) <https://www.piter.com/collection/programmirovanie-pod-android-i-ios/product/android-programmirovanie-dlya-professionalov-4-e-izdanie>
6. Смит, К. TensorFlow Lite для мобильных разработчиков / К. Смит. — М.: ДМК Пресс, 2023. — 312 с. — ISBN 978-5-97060-980-4 (посещался 20.04.2026) <https://www.litres.ru/book/dzhon-horton-21544363/razrabotka-android-prilozheniy-s-nulya-69842353/>
7. Морозов, Д.А. Разработка мобильных приложений с искусственным интеллектом [Электронный ресурс] : учебное пособие для СПО / Д.А. Морозов. — Саратов: Профобразование, 2024. — 186 с. — Режим доступа: <https://profspo.ru/books/141234> (дата обращения: 15.05.2026). — ISBN 978-5-4488-2156-1 (посещался 20.04.2026)
8. Лебедев, И.С. Kotlin для разработчиков Android [Электронный ресурс] : учебное пособие / И.С. Лебедев. — М.: КноРус, 2025. — 278 с. — Режим доступа: <https://book.ru/book/956712> (дата обращения: 15.05.2026). — ISBN 978-5-406-14567-8 (посещался 20.04.2026)

Дополнительные источники:

1. Галеева, Л.И. Основы тестирования мобильных приложений [Электронный ресурс] : практикум для СПО / Л.И. Галеева. — Саратов: Профобразование, 2024. — 112 с. — Режим доступа: <https://profspo.ru/books/141456> (дата обращения: 15.05.2026). — ISBN 978-5-4488-2189-9 (посещался 20.04.2026)
2. Дорофеев, А.В. Git и GitHub для командной разработки [Электронный ресурс] : учебное пособие / А.В. Дорофеев. — М.: Ай Пи Ар Медиа, 2024. — 156 с. — Режим доступа: <https://profspo.ru/books/138892> (дата обращения: 15.05.2026). — ISBN 978-5-4497-2967-5 (посещался 20.04.2026)

3. Николаев, В.П. ML Kit: практическое руководство для Android-разработчиков / В.П. Николаев. — М.: Лаборатория знаний, 2024. — 198 с. — ISBN 978-5-00101-367-9 (посещался 20.04.2026) https://znanium.ru/?spm=a2ty_o01.29997173.0.0.73af55fbcT6cj7
4. Воробьев, Е.А. Firebase для мобильных приложений / Е.А. Воробьев. — СПб.: БХВ-Петербург, 2023. — 342 с. — ISBN 978-5-9775-6866-7. https://www.flip.kz/descript?spm=a2ty_o01.29997173.0.0.73af55fbcT6cj7&cat=publish&id=242
5. Комарова, Т.Н. Java и Kotlin: сравнительный анализ для мобильной разработки [Электронный ресурс] : учебное пособие / Т.Н. Комарова. — Саратов: Профобразование, 2025. — 98 с. — Режим доступа: <https://profspo.ru/books/143011> (дата обращения: 15.05.2026). — ISBN 978-5-4488-2367-1.

Периодические издания:

1. Журнал «Открытые системы. СУБД»
2. Журнал «Информационные технологии и вычислительные системы»

Интернет-ресурсы:

1. Официальная документация Python 3 [Электронный ресурс] – Режим доступа: <https://docs.python.org/3/>, свободный. – Загл. с экрана. – Яз. англ (посещался 20.04.2026)
2. Scikit-learn: Machine Learning in Python [Электронный ресурс] – Режим доступа: <https://scikit-learn.org/stable/>, свободный. – Загл. с экрана. – Яз. англ (посещался 20.04.2026)
3. TensorFlow Core Documentation [Электронный ресурс] – Режим доступа: https://www.tensorflow.org/api_docs, свободный. – Загл. с экрана. – Яз. англ (посещался 20.04.2026)
4. PyTorch Documentation [Электронный ресурс] – Режим доступа: <https://pytorch.org/docs/stable/index.html>, свободный. – Загл. с экрана. – Яз. англ (посещался 20.04.2026)
5. Git Documentation [Электронный ресурс] – Режим доступа: <https://git-scm.com/doc>, свободный. – Загл. с экрана. – Яз. англ (посещался 20.04.2026)
6. Kaggle: Datasets and Competitions [Электронный ресурс] – Режим доступа: <https://www.kaggle.com/>, свободный. – Загл. с экрана. – Яз. Англ (посещался 20.04.2026)
7. Журнал «Android Developer Magazine». — Режим доступа: <https://android-developers.googleblog.com> (посещался 20.04.2026)
8. Журнал «Программист». — Издательство «Открытые системы». — Режим доступа: <https://www.programmer.ru> (посещался 20.04.2026)
9. «Искусственный интеллект и принятие решений» (ФИЦ ИУ РАН). — Режим доступа: <https://www.iaas-journal.ru> (посещался 20.04.2026)

3.3 Учебно-методическое обеспечение самостоятельной работы обучающихся

Самостоятельная работа является обязательной для каждого обучающегося. Самостоятельная работа может осуществляться индивидуально или группами в зависимости от цели, объема, конкретной тематики самостоятельной работы, уровня сложности, уровня умений обучающихся.

Контроль результатов внеаудиторной самостоятельной работы осуществляется в пределах времени, отведенного на обязательные учебные занятия и внеаудиторную самостоятельную работу обучающихся по учебной дисциплине, проходит как в письменной, так и устной или смешанной форме, с представлением изделия или продукта самостоятельной деятельности.

В качестве форм и методов контроля внеаудиторной самостоятельной работы используются: проверка выполненной работы преподавателем, семинарские занятия, тестирование, самоотчеты, контрольные работы, защита творческих работ, проверка репозиторий с кодом и др.

№	Наименование раздела/темы	Оценочные средства (задания) для самостоятельной внеаудиторной работы
1	Раздел 1. МДК.01.01 Тема 1.1. Введение в искусственный	Вид задания: Аналитический обзор (реферат). Текст задания: Подготовить реферат на тему «Современные системы ИИ и их применение». Проанализировать не менее 3-х

	интеллект и постановка задач	направлений ИИ (машинное обучение, компьютерное зрение, NLP) и привести примеры их использования в реальных задачах. Цель: сформировать умение анализировать информацию и выделять ключевые тенденции в области ИИ. Рекомендации: Использовать открытые источники, статьи на Habr, документацию библиотек. Оформить по ГОСТ (титульный лист, содержание, список источников). Критерии оценки: «5» — глубокий анализ, ≥ 5 источников, корректное оформление; «4» — поверхностный анализ или незначительные ошибки в оформлении; «3» — анализ не полный, мало источников; «2» — работа не выполнена.
2	Раздел 1. МДК.01.01 Тема 1.2. Базовые конструкции языка Python	Вид задания: Решение задач на программирование. Текст задания: Составить 5 программ на Python, демонстрирующих использование: условного оператора, цикла for, цикла while, функции с возвратом значения, обработки исключений try-except. Цель: закрепить навыки работы с базовыми конструкциями языка и алгоритмизации. Рекомендации: Использовать среду PyCharm или VS Code, соблюдать стандарт PEP8. Сохранить все программы в одной папке «СР_Раздел1». Критерии оценки: «5» — все программы работают, код по PEP8, есть комментарии; «4» — есть 1–2 нарушения стиля; «3» — программы работают с ошибками; «2» — менее 3 программ.
3	Раздел 1. МДК.01.01 Тема 1.3. Нейросетевые библиотеки (TensorFlow, PyTorch)	Вид задания: Эксперимент по подбору гиперпараметров. Текст задания: Для модели из лабораторной работы №9 провести эксперименты с эпохами (10, 20, 30), размером пакета (16, 32, 64) и функцией активации. Построить таблицу точности, выбрать лучшую комбинацию. Цель: научить настраивать нейросетевые модели и оценивать качество обучения. Рекомендации: Использовать Jupyter Notebook, фиксировать историю обучения, построить графики зависимости точности от параметров. Критерии оценки: «5» — все параметры протестированы, таблица полная, вывод обоснован; «4» — протестированы 2 из 3 параметров; «3» — протестирован 1 параметр; «2» — не выполнено.
4	Раздел 1. МДК.01.01 Тема 1.4. Система контроля версий Git	Вид задания: Работа с удалённым репозиторием GitHub. Текст задания: Создать репозиторий на GitHub, загрузить код лабораторной работы №7, создать ветку 'develop', сделать pull request в 'main'. В README.md описать проект. Цель: освоить базовые операции Git и навыки командной разработки. Рекомендации: Использовать Git Bash или терминал PyCharm. Команды: 'git init', 'add', 'commit', 'branch', 'push', 'pull request'. Критерии оценки: «5» — выполнены все пункты, PR создан; «4» — нет pull request; «3» — нет ветки develop; «2» — репозиторий не создан.
5	Раздел 3. МДК.01.03 Тема 3.1. Основы отладки и диагностики программного кода	Вид задания: Практикум по отладке. Текст задания: Найти и устранить 5 ошибок в предоставленном коде с использованием отладчика PyCharm/VS Code. Внедрить многоуровневое логирование (DEBUG, INFO, ERROR). Цель: закрепить навыки пошаговой отладки и диагностики ошибок.

		Рекомендации: Использовать breakpoints, watch variables, call stack. Настроить ротацию логов. Критерии оценки: «5» — все ошибки найдены, логирование работает; «4» — 1-2 ошибки пропущены; «3» — найдено менее 3 ошибок; «2» — задание не выполнено.
6	Раздел 3. МДК.01.03 Тема 3.2. Модульное и интеграционное тестирование	Вид задания: Разработка тестов. Текст задания: Написать 10 unit-тестов с использованием pytest для модуля обработки данных. Использовать фикстуры, параметризацию, mock-объекты. Цель: освоить техники модульного тестирования. Рекомендации: Применять pytest-mock для изоляции зависимостей. Обеспечить покрытие $\geq 80\%$. Критерии оценки: «5» — все тесты проходят, покрытие $\geq 80\%$; «4» — покрытие 60-79%; «3» — покрытие $< 60\%$; «2» — тесты не работают.
7	Раздел 3. МДК.01.03 Тема 3.3. Тестирование ИИ-моделей и нейросетевых компонентов	Вид задания: Анализ качества модели. Текст задания: Рассчитать метрики качества (accuracy, precision, recall, F1, ROC-AUC) для модели классификации. Построить confusion matrix и ROC-кривую. Цель: научиться оценивать качество ИИ-моделей. Рекомендации: Использовать scikit-learn.metrics, matplotlib для визуализации. Критерии оценки: «5» — все метрики рассчитаны, визуализация качественная; «4» — есть незначительные ошибки; «3» — рассчитаны базовые метрики; «2» — задание не выполнено.
8	Раздел 3. МДК.01.03 Тема 3.4. Автоматизация тестирования и CI/CD	Вид задания: Настройка CI-пайплайна. Текст задания: Создать GitHub Actions workflow для автоматического запуска тестов при push. Настроить генерацию отчёта о покрытии. Цель: освоить автоматизацию тестирования. Рекомендации: Использовать pytest-cov, actions/checkout, setup-python. Критерии оценки: «5» — workflow работает, тесты запускаются автоматически; «4» — есть ошибки в конфигурации; «3» — workflow создан, но не работает; «2» — задание не выполнено.
9	Раздел 3. МДК.01.03 Тема 3.5. Разработка тестовой документации и сценариев	Вид задания: Создание тестовой документации. Текст задания: Разработать тест-план, 5 тест-кейсов и чек-лист для модуля ИИ. Использовать промт-инжиниринг для генерации тестовых данных. Цель: научиться проектировать тестовую документацию. Рекомендации: Применять шаблоны тест-кейсов, ChatGPT/Claude для генерации данных. Критерии оценки: «5» — документация полная, соответствует стандартам; «4» — есть недочёты в оформлении; «3» — документация неполная; «2» — задание не выполнено.

4 КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

Контроль и оценка результатов освоения профессионального модуля осуществляется преподавателем в процессе текущего контроля и промежуточной аттестации.

Формой промежуточной аттестации по профессиональному модулю является экзамен по профессиональному модулю.

4.1 Текущий контроль

Контролируемые разделы (темы) учебной дисциплины	Наименование оценочного средства	Критерии оценки
МДК.01.01 Разработка программных модулей в системах искусственного интеллекта		
ПК 1.1, ПК 1.2, ОК 01, ОК 02	Лабораторное занятие	См. ниже
ПК 1.2, ПК 1.3, ОК 02, ОК 09	Лабораторное занятие	См. ниже
ПК 1.3, ПК 1.4, ОК 04, ОК 05	Лабораторное занятие	См. ниже
ПК 1.5, ПК 1.6, ПК 1.7, ОК 02, ОК 05	Лабораторное занятие	См. ниже
МДК.01.02 Разработка мобильных приложений с поддержкой искусственного интеллекта		
ПК 1.2, ПК 1.3, ПК 1.5, ПК 1.6, ОК 01, ОК 02, ОК 05	Лабораторное занятие	См. ниже
ПК 1.2, ПК 1.5, ПК 1.6, ПК 1.7, ОК 01, ОК 02, ОК 09	Лабораторное занятие	См. ниже
ПК 1.5, ПК 1.6, ПК 1.7, ОК 01, ОК 02, ОК 04	Лабораторное занятие	См. ниже
ПК 1.1-1.7, ОК 01, ОК 02, ОК 04, ОК 05, ОК 09	Виды работ по практике	См. ниже
МДК.01.01 Разработка программных модулей в системах искусственного интеллекта		
ПК 1.5, ОК 02	Лабораторное занятие	См. ниже
ПК 1.6, ОК 04	Лабораторное занятие	См. ниже
ПК 1.6, ПК 1.7, ОК 01	Лабораторное занятие	См. ниже
ПК 1.7, ОК 02	Лабораторное занятие	См. ниже
ПК 1.7, ОК 05	Лабораторное занятие	См. ниже
ПК 1.1-1.7, ОК 01, ОК 02, ОК 04, ОК 05, ОК 09	Виды работ по практике	См. ниже
УП.01 Учебная практика		
	Отчёт по практике	См. ниже

Критерии оценки лабораторного занятия:

«5» (отлично): выполнены все задания лабораторной работы, студент четко и без ошибок ответил на все контрольные вопросы.

«4» (хорошо): выполнены все задания лабораторной работы; студент ответил на все контрольные вопросы с замечаниями.

«3» (удовлетворительно): выполнены все задания лабораторной работы с замечаниями; студент ответил на все контрольные вопросы с замечаниями.

«2» (неудовлетворительно): студент не выполнил или выполнил неправильно задания лабораторной работы; студент ответил на контрольные вопросы с ошибками или не ответил на контрольные вопросы.

4.2 Промежуточная аттестация

Код	Структурный элемент профессионального модуля	Форма промежуточной аттестации	Семестр
МДК.01.01	Разработка программных модулей в системах искусственного интеллекта	Экзамен	3,4,6
МДК.01.02	Разработка мобильных приложений с поддержкой искусственного интеллекта	Экзамен	6
МДК 01.03	Отладка и тестирование программных модулей	Зачет с оценкой	6
УП.01	Учебная практика	Зачет	4,5,6
ПП.01	Производственная практика	Зачет	6
ПМ.01.Э	Экзамен по профессиональному модулю	Экзамен	6

4.2.1 Оценочные средства для зачета, экзамена по МДК, практике

Оценочные средства для экзамена по МДК.01.01 Разработка программных модулей в системах искусственного интеллекта 3 семестр

Результаты обучения	Оценочные средства для промежуточной аттестации
ПК 1.1–1.4, ОК 01, 02, 04, 05, 09	1. Теоретическая часть (устный опрос по билетам / тест из 20 вопросов) 1. Что такое жизненный цикл программного модуля? Перечислите основные этапы. 2. Какие инструменты используются для контроля версий? Опишите базовые команды Git. 3. В чём разница между модульным и интеграционным тестированием? 4. Какие правила оформления кода предусмотрены стандартом PEP8? 5. Опишите методы отладки программ в PyCharm/VS Code. 6. Какие библиотеки Python используются для задач машинного обучения? 7. Что такое техническое задание и какие разделы оно содержит? 8. Как организовать групповую разработку с использованием Git Flow? 9. Какие виды ошибок могут возникать при выполнении программы? 10. Как документировать код с помощью docstring и Sphinx?

	2. Практическая часть (кейс-задание) Задание: Разработать программный модуль на Python для решения задачи классификации данных (например, набор данных Iris) с использованием библиотеки scikit-learn. Требования: 1. Загрузить данные из CSV-файла, выполнить предобработку (удаление пропусков, нормализация). 2. Разделить выборку на обучающую и тестовую, обучить модель классификации. 3. Оценить качество модели (точность, матрица ошибок). 4. Оформить код в соответствии с PEP8, добавить комментарии и docstring. 5. Сохранить код в локальный репозиторий Git, сделать коммит с осмысленным сообщением. 6. Написать не менее 2 юнит-тестов для ключевых функций модуля.
ПК 1.1–1.7, ОК 01, 02, 04, 05, 09	Комплексное практическое задание (проектная работа) Задание: Разработать и представить полноценный программный модуль ИИ с полным циклом: от ТЗ до тестирования и документирования. Требования: 1. Проанализировать ТЗ, выделить функциональные требования, составить план разработки. 2. Реализовать модуль на Python с использованием библиотек ИИ (NumPy, Pandas, scikit-learn / TensorFlow). 3. Оформить код по стандарту PEP8, сгенерировать документацию через Sphinx. 4. Организовать репозиторий на GitHub, продемонстрировать историю коммитов и работу с ветками. 5. Выполнить отладку модуля, внедрить логирование, устранить ошибки. 6. Написать модульные тесты (покрытие $\geq 70\%$), оценить качество модели метриками. 7. Составить тестовый сценарий, автоматизировать запуск тестов через CI (GitHub Actions). 8. Подготовить отчёт и презентацию для защиты проекта. Форма представления: – Ссылка на репозиторий GitHub с кодом, тестами, документацией; – Отчёт в формате PDF с описанием решения, метриками, скриншотами; – Устная защита (5–7 минут) с ответами на вопросы.

Оценочные средства для экзамена по МДК.01.01 Разработка программных модулей в системах искусственного интеллекта 4 семестр

Результаты обучения	Оценочные средства для промежуточной аттестации
ПК 1.5–1.7, ОК 01, 02, 04, 05, 09	1. Теоретическая часть (устный опрос / тест из 15 вопросов) 1. Какие виды тестирования программного обеспечения

	вы знаете? 2. В чём разница между модульным и интеграционным тестированием? 3. Что такое отладка? Какие инструменты отладки вы использовали? 4. Какие метрики качества моделей ИИ вы знаете? 5. Что такое confusion matrix и как её интерпретировать? 6. Объясните разницу между precision и recall. 7. Что такое ROC-кривая и AUC? 8. Какие инструменты контроля версий вы знаете? 9. Что такое CI/CD? Для чего он нужен? 10. Как настроить автоматический запуск тестов в GitHub Actions? 11. Что такое логирование? Какие уровни логирования вы знаете? 12. Как оценить покрытие кода тестами? 13. Что такое тест-кейс? Из каких элементов он состоит? 14. Какие виды ошибок могут возникать при выполнении программы? 15. Как документировать код с помощью docstring? 2. Практическая часть (кейс-задание) Задание: Провести комплексное тестирование и отладку программного модуля ИИ. Требования: 1. Найти и устранить ошибки в предоставленном коде с использованием отладчика. 2. Написать ≥ 5 модульных тестов с использованием pytest/unittest. 3. Оценить покрытие кода тестами (цель: $\geq 70\%$) (ПК 1.6). 4. Рассчитать метрики качества модели (precision, recall, F1). 5. Оформить отчёт о тестировании с выводами и рекомендациями. Форма представления: – Ссылка на репозиторий с кодом и тестами; – Файл TEST_REPORT.md с результатами; – Скриншот отчёта о покрытии кода.
ПК 1.1–1.7, ОК 01, 02, 04, 05, 09	Комплексное практическое задание (проектная работа) Задание: Разработать систему обеспечения качества для программного модуля ИИ с полным циклом тестирования и отладки. Требования: 1. Проанализировать модуль, выделить критические функции для тестирования. 2. Разработать тестовую документацию (тест-план, тест-

	кейсы, чек-листы). 3. Написать модульные и интеграционные тесты (покрытие $\geq 80\%$). 4. Провести отладку модуля, внедрить логирование (ПК 1.5). 5. Рассчитать метрики качества модели, построить ROC-кривую (ПК 1.7). 6. Настроить CI/CD-пайплайн с автоматическим запуском тестов (ПК 1.4). 7. Сгенерировать отчёт о покрытии и качестве кода. 8. Подготовить презентацию для защиты проекта. Форма представления: – Ссылка на репозиторий GitHub с кодом, тестами, документацией; – Отчёт о тестировании в формате PDF; – Устная защита (5–7 минут) с демонстрацией работы CI/CD. Критерии оценки: «Отлично» — все задания выполнены, код работает стабильно, тесты покрывают $\geq 80\%$, защита полная, ответы на вопросы верные. «Хорошо» — задания выполнены, есть незначительные недочёты в коде или тестах. «Удовлетворительно» — основные задания выполнены, но есть ошибки в реализации или тестировании. «Неудовлетворительно» — задания не выполнены или код не работает.
--	---

Оценочные средства для зачета с оценкой по МДК.01.01 Разработка программных модулей в системах искусственного интеллекта 6 семестр

Результаты обучения	Оценочные средства для промежуточной аттестации
ПК 1.1–1.7, ОК 01, 02, 04, 05, 09	Комплексное практическое задание (итоговый проект) Задание: Разработать и защитить программный модуль ИИ с полным циклом: от ТЗ до тестирования и документирования. Требования: 1. Проанализировать ТЗ, выделить функциональные требования, составить план разработки). 2. Реализовать модуль на Python/Kotlin с использованием библиотек ИИ (NumPy, Pandas, scikit-learn, TensorFlow/PyTorch, ML Kit). 3. Оформить код по стандарту PEP8/Kotlin conventions, сгенерировать документацию через Sphinx/KDoc (ПК 1.3). 4. Организовать репозиторий на GitHub, продемонстрировать историю коммитов и работу с ветками.

	5. Выполнить отладку модуля, внедрить логирование, устранить ошибки. 6. Написать модульные тесты (покрытие $\geq 70\%$), оценить качество модели метриками (accuracy, precision, recall, F1). 7. Составить тестовый сценарий, автоматизировать запуск тестов через CI (GitHub Actions). 8. Подготовить отчёт и презентацию для защиты проекта. Форма представления: – Ссылка на репозиторий GitHub с кодом, тестами, документацией; – Отчёт в формате PDF с описанием решения, метриками, скриншотами; – Устная защита (5–7 минут) с ответами на вопросы. Критерии оценки: «5» — все требования выполнены, код работает стабильно, тесты покрывают $\geq 70\%$, защита полная; «4» — задания выполнены с незначительными замечаниями; «3» — основные требования выполнены, но есть ошибки в реализации; «2» — задания не выполнены или код не работает.
--	---

Оценочные средства для экзамена по МДК.01.02 Разработка мобильных приложений с поддержкой искусственного интеллекта

Результаты обучения	Оценочные средства для промежуточной аттестации
ПК 1.1-1.7, ОК 01, ОК 02, ОК 04, ОК 05, ОК 09	Наименование оценочного средства: Защита итогового проекта (кейс). Текст типового оценочного средства: 1. Представить разработанное мобильное приложение (демонстрация работы). 2. Презентовать архитектуру приложения и использованные технологии. 3. Продемонстрировать интеграцию ИИ-функций (распознавание изображений, текста, голоса). 4. Обосновать выбор ИИ-инструментов и методов оптимизации. 5. Представить результаты тестирования приложения (unit-тесты, тестирование ИИ-компонентов). 6. Ответить на вопросы преподавателя по коду и архитектуре. Критерии оценки: «Отлично» — все задания выполнены, приложение работает стабильно, ИИ-функции реализованы качественно, защита полная, ответы на вопросы верные. «Хорошо» — задания выполнены, есть незначительные недочеты в работе приложения или ИИ-функций. «Удовлетворительно» — основные задания выполнены,

	но приложение работает с ошибками или ИИ-функции работают неточно. «Неудовлетворительно» — задания не выполнены или приложение не работает.
--	--

Оценочные средства для зачета с оценкой по МДК.01.03 Отладка и тестирование программных модулей

Результаты обучения	Оценочные средства для промежуточной аттестации
ПК 1.5–1.7, ОК 01, 02, 04, 05, 09	1. Теоретическая часть (устный опрос / тест из 15 вопросов) 1. Какие виды тестирования вы знаете? В чем их различие? 2. Что такое модульное тестирование? Приведите примеры. 3. Какие метрики качества моделей ИИ вы знаете? 4. Что такое confusion matrix и как её интерпретировать? 5. Объясните разницу между precision и recall. 6. Что такое ROC-кривая и AUC? 7. Какие инструменты отладки вы использовали? 8. Что такое CI/CD? Для чего он нужен? 9. Как настроить автоматический запуск тестов в GitHub Actions? 10. Что такое mock-объекты и когда их применять? 11. Как оценить покрытие кода тестами? 12. Что такое тест-кейс? Из каких элементов он состоит? 13. Какие уровни логирования вы знаете? 14. Что такое интеграционное тестирование? 15. Как проводить тестирование ИИ-моделей? 2. Практическая часть (кейс-задание) Задание: Провести комплексное тестирование программного модуля ИИ. Требования: 1. Написать ≥ 5 модульных тестов с использованием pytest. 2. Оценить покрытие кода (цель: $\geq 70\%$) (ПК 1.6). 3. Рассчитать метрики качества модели (precision, recall, F1). 4. Настроить GitHub Actions для автоматического запуска тестов. 5. Оформить отчёт о тестировании с выводами и рекомендациями. Форма представления: – Ссылка на репозиторий с кодом и тестами; – Файл TEST_REPORT.md с результатами; – Скриншот успешного запуска workflow в GitHub Actions. Критерии оценки: «Отлично» — все задания выполнены, приложение

	работает стабильно, ИИ-функции реализованы качественно, защита полная, ответы на вопросы верные. «Хорошо» — задания выполнены, есть незначительные недочеты в работе приложения или ИИ-функций. «Удовлетворительно» — основные задания выполнены, но приложение работает с ошибками или ИИ-функции работают неточно. «Неудовлетворительно» — задания не выполнены или приложение не работает.
ПК 1.5–1.7, ОК 01, 02, 04, 05, 09	Комплексное практическое задание (проектная работа) Задание: Разработать систему обеспечения качества для программного модуля ИИ с полным циклом тестирования. Требования: 1. Проанализировать модуль, выделить критические функции для тестирования. 2. Разработать тестовую документацию (тест-план, тест-кейсы, чек-листы). 3. Написать модульные и интеграционные тесты (покрытие $\geq 80\%$). 4. Провести тестирование ИИ-модели: рассчитать метрики качества, построить ROC-кривую. 5. Внедрить логирование для диагностики ошибок. 6. Настроить CI/CD-пайплайн с автоматическим запуском тестов. 7. Сгенерировать отчёт о покрытии и качестве кода (ОК 02). 8. Использовать промт-инжиниринг для генерации тестовых данных. Форма представления: – Ссылка на репозиторий GitHub с кодом, тестами, документацией; – Отчёт о тестировании в формате PDF; – Устная защита (5–7 минут) с демонстрацией работы CI/CD. Критерии оценки: «Отлично» — все задания выполнены, приложение работает стабильно, ИИ-функции реализованы качественно, защита полная, ответы на вопросы верные. «Хорошо» — задания выполнены, есть незначительные недочеты в работе приложения или ИИ-функций. «Удовлетворительно» — основные задания выполнены, но приложение работает с ошибками или ИИ-функции работают неточно. «Неудовлетворительно» — задания не выполнены или приложение не работает.

Оценочные средства для зачета по Учебной практике

Результаты обучения	Оценочные средства
---------------------	--------------------

(ПК, ОК)	для промежуточной аттестации
ПК 1.1 – ПК 1.7, ОК 01, 02, 04, 05, 09	Виды работ и задание на производственную практику

Оценочные средства для зачета по Производственной практике

Результаты обучения (ПК, ОК)	Оценочные средства для промежуточной аттестации
ПК 1.1 – ПК 1.7, ОК 01, 02, 04, 05, 09	Виды работ и задание на производственную практику

Критерии зачета с оценкой экзамена

«Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.

«Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.

«Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.

«Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.

Для оценки образовательных достижений обучающихся применяется универсальная шкала, представленная в приложении 2.

4.2.2 Экзамен по профессиональному модулю

Оценочные средства промежуточной аттестации по профессиональному модулю – экзамену профессиональному по модулю

Код и наименование компетенций	Перечень оцениваемых умений, навыков (практического опыта)	Максимальный балл
ОК 01. Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам	Умение анализировать техническое задание, выделять функциональные требования и составлять план разработки ИИ-модуля	2
ОК 02. Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности	Умение применять современные ИТ-инструменты (IDE, библиотеки, эмуляторы, промт-инжиниринг) для решения профессиональных задач	2
ОК 04. Эффективно взаимодействовать и работать в коллективе и команде	Умение организовывать командную разработку, распределять задачи, проводить код-ревью и использовать инструменты совместной работы	2
ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста	Умение грамотно оформлять техническую документацию, отчёты и презентовать результаты разработки	2
ОК 09. Пользоваться профессиональной документацией на государственном и иностранном языках	Умение читать, понимать и применять англоязычную документацию (API, библиотеки, фреймворки) при реализации проекта	2
ПК 1.1. Формировать алгоритмы разработки программных модулей	Навык проектирования алгоритмов обработки данных и построения логики работы ИИ-модулей	2
ПК 1.2. Разрабатывать программные модули	Навык написания рабочего кода на Python и Kotlin/Java, интеграции библиотек ИИ (NumPy, Pandas, scikit-learn, TensorFlow/PyTorch, ML Kit)	2
ПК 1.3. Оформлять программный код	Навык соблюдения стандартов оформления (PEP8, Kotlin conventions), написания docstring и генерации документации	2
ПК 1.4. Использовать систему контроля версий	Навык работы с Git: создание веток, осмысленные коммиты, merge, разрешение	2

	конфликтов, pull request	
ПК 1.5. Выполнять отладку программных модулей	Навык использования отладчиков (PyCharm, Android Studio), брейкпоинтов, логирования и анализа стека вызовов	2
ПК 1.6. Выполнять тестирование программного кода	Навык написания модульных и интеграционных тестов (unittest, pytest, JUnit), оценки покрытия кода	3
ПК 1.7. Составлять тестовые сценарии	Навык проектирования тест-кейсов, автоматизации запуска тестов через CI/CD (GitHub Actions) и анализа результатов	2
ИТОГО		25

Задание 1.

Текст задания: Разработать программный модуль для предобработки и классификации табличных данных с использованием библиотеки scikit-learn.

Исходные данные: CSV-файл с датасетом (предоставляется преподавателем), содержащий пропуски, категориальные и числовые признаки.

Требования:

- Загрузить данные, выполнить очистку (заполнение пропусков, кодирование категорий).
- Реализовать алгоритм классификации (LogisticRegression или RandomForest).
- Разделить выборку, обучить модель, вывести метрики (accuracy, precision, recall, F1).
- Код должен соответствовать PEP8, содержать docstring и комментарии.
- Разместить код в Git-репозитории с историей не менее 5 коммитов и веткой develop.

Форма представления: Ссылка на репозиторий, файл requirements.txt, скриншоты вывода метрик и структуры проекта.

Задание 2.

Текст задания: Создать Android-приложение на Kotlin с функцией распознавания текста или изображений в реальном времени.

Исходные данные: Android Studio, эмулятор или реальное устройство с камерой, документация ML Kit / TensorFlow Lite.

Требования:

- Реализовать интерфейс с кнопкой запуска и областью вывода результата.
- Интегрировать ML Kit (Text Recognition или Image Labeling) или TFLite-модель.
- Обработать результат распознавания и вывести его на экран.
- Настроить логирование основных этапов работы (запуск модели, получение данных, ошибка).
- Применить промт-инжиниринг для генерации/оптимизации фрагментов кода (приложить скриншот диалога с ИИ-ассистентом).

Форма представления: Архив проекта или ссылка на репозиторий, APK-файл, скриншоты работы приложения на устройстве/эмуляторе.

Задание 3.

Текст задания: Провести комплексное тестирование одного из разработанных модулей (из Задания 1 или 2) и настроить CI-пайплайн.

Исходные данные: Готовый код модуля, доступ к GitHub/GitLab.

Требования:

- Написать не менее 3 unit-тестов (pytest или JUnit) для проверки граничных значений и основного функционала.
- Оценить покрытие кода тестами (цель $\geq 70\%$).
- Разработать тестовый сценарий (чек-лист) на 5 шагов с предусловиями и ожидаемым результатом.
- Настроить GitHub Actions workflow для автоматического запуска тестов при push в ветку main.
- Оформить краткий отчёт о результатах тестирования и найденных дефектах.

Форма представления: Файлы тестов, скриншот отчёта о покрытии, ссылка на успешный запуск workflow в GitHub Actions, файл TEST_REPORT.md.

Критерии оценивания:

Оцениваемая ПК, ОК	Критерий оценивания (умения, навыки)	Описание оценки критерия		Максимальный балл оценки подкритерия - 2 балла	Вес подкритерия: не менее 1; - шаг 0,5; не более 3	Итоговый максимальный балл подкритерия
		Конкретные оцениваемые действия или набор действий для оценки подкритерия	Описание результата выполнения конкретного действия подкритерия в баллах			
ПК 1.1. Формировать алгоритмы разработки программных модулей	анализировать ТЗ и выделять функциональные требования	выделяет входные/выходные данные, составляет план разработки	0 – не выделяет требования; 1 – выделяет частично; 2 – полно и точно формулирует требования и план	2	1	2
ПК 1.2. Разрабатывать программные модули	реализовывать работающий модуль на Python с использованием библиотек ИИ	код работает, решает поставленную задачу, использует соответствующие	0 – код не работает; 1 – работает с ошибками; 2 – работает корректно,	2	1	2

		библиотеки	код оптимален			
ПК 1.3. Оформлять программный код	соблюдать стандарты оформления (PEP8), писать документацию	код соответствует стилю, содержит docstring, сгенерирована документация	0 – стиль не соблюден; 1 – есть замечания; 2 – полное соответствие стандартам	2	1	2
ПК 1.4. Использовать систему контроля версий	работать с Git: коммиты, ветки, pull request	репозиторий содержит осмысленную историю, использованы ветки, оформлен PR	0 – нет репозитория; 1 – базовые операции; 2 – полноценная работа с ветками и PR	2	1	2
ПК 1.5. Выполнять отладку программных модулей	использовать отладчик и логирование для поиска ошибок	ошибки найдены и устранены, внедрено логирование	0 – отладка не выполнена; 1 – частично; 2 – полностью, с обоснованием	2	1	2
ПК 1.6. Выполнять тестирование программного кода	писать модульные тесты, оценивать покрытие	тесты покрывают $\geq 70\%$ кода, проходят успешно	0 – тестов нет; 1 – покрытие $< 70\%$; 2 – покрытие	2	2	4

			≥70%, тесты релевантн ы			
ПК 1.7. Составля ть тестовые сценарии	разрабатывать тест-кейсы, автоматизиро вать тестирование	тест- сценарий оформлен, интеграция с CI (GitHub Actions)	0 – сценарий отсутствует; 1 – оформлен частично; 2 – полный сценарий + CI	2	1,5	3
ОК 02. Используй вать современ ные средства поиска информа ции	находить и применять актуальную документаци ю	использует официальн ые источники (Python docs, scikit- learn)	0 – непроверен ные источники; 1 – с ошибками; 2 – эффективн о и точно	2	1	2
ОК 04. Работать в коллекти ве и команде	использовать навыки управления проектами, распределять задачи	соблюдает график, распределяе т нагрузку в группе (если проект групповой)	0 – не следует графику; 1 – с отклонения ми; 2 – грамотно планирует и выполняет	2	1	2
ОК 05. Осущест влять коммуни кацию на	оформлять отчёт и презентацию по профессионал	отчёт структурир ован, презентаци я	0 – оформлени е с нарушения	2	1	2

русском языке	ьной тематике	информативна, язык грамотный	ми; 1 – частично; 2 – полностью соответствует требованиям			
ОК 09. Пользоваться профессиональной документацией на иностранном языке	читать и применять англоязычную документацию	корректно интерпретирует API docs, терминологию	0 – не применяет; 1 – с трудом; 2 – свободно и точно	2	1	2

Для оценки образовательных достижений обучающихся применяется универсальная шкала, представленная в приложении 2.

ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

При проведении теоретических и практических/лабораторных занятий используются следующие педагогические технологии:

№ п/п	Название образовательной технологии (с указанием автора)	Цель использования образовательной технологии	Планируемый результат использования образовательной технологии	Описание порядка использования (алгоритм применения) технологии в практической профессиональной деятельности
1	Проектное обучение (Дж. Дьюи, У. Килпатрик)	Формирование навыков комплексной разработки программных модулей для ИИ, от анализа ТЗ до тестирования и документирования	Разработанный и защищённый программный модуль (классификатор, регрессор, нейросеть), умение работать в команде, использовать Git, оформлять код по стандартам	На протяжении семестра студенты выполняют индивидуальный или групповой (2–3 чел.) проект. Этапы: анализ ТЗ → проектирование алгоритма → реализация кода → отладка → тестирование → документирование → защита. Преподаватель выступает в роли эксперта-заказчика
2	Проблемное обучение (Дж. Дьюи, М.И. Махмутов)	Развитие аналитических способностей, умения выбирать оптимальный алгоритм и библиотеку под конкретную задачу ИИ	Умение анализировать проблему, обосновывать выбор структуры данных, алгоритма и метрик качества, интерпретировать результаты выполнения кода	На лабораторных занятиях предлагаются реальные кейсы: «Оптимизация обработки больших данных», «Выбор между scikit-learn и TensorFlow для задачи», «Устранение утечки памяти в цикле». Студенты в малых группах за 15–20 минут предлагают

				решение, затем защищают и тестируют его
3	Перевернутое обучение (Flipped classroom, J. Bergmann, A. Sams)	Повышение самостоятельности и, углубление понимания теоретических основ алгоритмов, архитектур нейросетей и принципов тестирования	Более качественная подготовка к лабораторным работам, активное участие в обсуждении сложных вопросов, сокращение времени на объяснение базовых понятий	Теоретический материал (видеолекции, статьи из официальной документации Python, scikit-learn, TensorFlow) студенты изучают дома. На аудиторных занятиях — разбор вопросов, отладка кода, выполнение лабораторных работ под руководством преподавателя
4	Обучение в сотрудничестве (кооперативное обучение, Д. Джонсон, Р. Джонсон)	Формирование навыков командной работы (ОК 04), распределения ролей, совместной разработки и код-ревью через систему контроля версий	Умение эффективно взаимодействовать в команде, проводить ревью кода, использовать общие ветки Git, разрешать конфликты слияния, документировать изменения	При выполнении лабораторных работ №10–13 и итогового проекта студенты разбиваются на пары/группы. Каждый участник отвечает за свой модуль (preprocessing, model, tests), но все пользуются общим репозиторием. В конце каждого занятия — ретроспектива и обмен лучшими практиками
5	Применение ИКТ и цифровых образовательных ресурсов (Г. Ливитт, Т. Уислер)	Обеспечение доступа к современным инструментам разработки (среды, библиотеки, CI/CD) и	Использование актуальных инструментов (VS Code, PyCharm, Jupyter, Git, GitHub Actions, Google Colab) для разработки,	Все занятия проводятся в компьютерном классе с выходом в интернет. Для ресурсоёмких задач (обучение нейросетей) используется Google

		облачным платформам для обучения моделей	тестирования и развёртывания программных модулей ИИ	Colab / Yandex Cloud. Код, тесты и отчёты загружаются в GitHub Classroom, проверяются через CI-пайплайны
6	Технология критического мышления через чтение и письмо (РКМЧП, Дж. Стил, К. Мередит, Ч. Темпл)	Развитие навыков анализа документации, формулирования точных требований к коду, критической оценки результатов тестирования и отладки	Умение грамотно читать техническую документацию, формулировать комментарии к коду, выявлять логические ошибки, аргументированно корректировать алгоритмы для достижения целевых метрик	На этапах изучения документации студенты используют приёмы: «Инсерт» (пометки на полях), «Кластер» (структурирование алгоритмов), «RAFT» (роль-аудитория-форма-тема) для написания комментариев, docstring и технической документации

Универсальная шкала перевода баллов в отметки по пятибалльной системе

Оценка	«2»	«3»	«4»	«5»
Отношение полученного количества баллов к максимально возможному (в %)	0,00 % - 49,99 %	50,00 % - 64,99 %	65,00 % – 89,99 %	90,00 % - 100,00 %
Количество баллов, полученных при защите курсового проекта (работы) (максимальный балл 25)	0 – 12,4	12,5 – 16,2	16,3 – 22,4	22,5 - 25
Количество баллов, полученных при сдаче экзамена по профессиональному модулю, в т.ч. в форме демонстрационного экзамена в рамках промежуточной аттестации / квалификационного экзамена (по каждой профессии) (максимальный балл 25)	0 – 12,4	12,5 – 16,2	16,3 – 22,4	22,5 - 25