

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.02 Разработка и интеграция модулей программного обеспечения
«профессионального цикла»
программы подготовки специалистов среднего звена
по специальности 09.02.11 Разработка и управление программным обеспечением**

Квалификация: Программист

Форма обучения
очная на базе основного общего образования

Рабочая программа профессионального модуля «Разработка и интеграция модулей программного обеспечения» разработана на основе: ФГОС по специальности среднего профессионального образования 09.02.11 Разработка и управление программным обеспечением, утвержденного приказом Министерства просвещения Российской Федерации от «24» февраля 2025 г. №138

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

Разработчики:

преподаватель отделения №2 «Информационных технологий и транспорта»
Многопрофильного колледжа ФГБОУ ВО «МГТУ им. Г.И. Носова»

Кристина Дмитриевна Рассадникова

преподаватель отделения №2 «Информационных технологий и транспорта»
Многопрофильного колледжа ФГБОУ ВО «МГТУ им. Г.И. Носова»

Оксана Викторовна Кобыльская

преподаватель отделения №2 «Информационных технологий и транспорта»
Многопрофильного колледжа ФГБОУ ВО «МГТУ им. Г.И. Носова»

Светлана Владимировна Меркулова

преподаватель отделения №2 «Информационных технологий и транспорта»
Многопрофильного колледжа ФГБОУ ВО «МГТУ им. Г.И. Носова»

Иван Алексеевич Кривонос

ОДОБРЕНО

Предметно-цикловой комиссией «Информатики
и вычислительной техники»

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.....	9
1.4 Трудоемкость профессионального модуля	10
2 СТРУКТУРА И СОДЕРЖАНИЕ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ	11
2.1 Структура профессионального модуля.....	11
2.2 Тематический план и содержание профессионального модуля	12
2.3 Перечень практических и лабораторных занятий	35
3 УСЛОВИЯ РЕАЛИЗАЦИИ РАБОЧЕЙ ПРОГРАММЫ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ ..	47
3.1 Материально-техническое обеспечение	47
3.2 Учебно-методическое и информационное обеспечение реализации программы	47
3.3 Учебно-методическое обеспечение самостоятельной работы обучающихся.....	Ошибка!
Закладка не определена.	20
4 КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ .	49
4.1 Текущий контроль	49
4.2 Промежуточная аттестация.....	51
Приложение 1 Образовательные технологии.....	74
Приложение 2 Фонд оценочных средств по дисциплине	Ошибка! Закладка не определена.
Приложение 3 Методические указания	Ошибка! Закладка не определена.

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 09.02.11 Разработка и управление программным обеспечением. Рабочая программа составлена для очной формы обучения.

Цель профессионального модуля: овладение видом профессиональной деятельности «Разработка и интеграция модулей программного обеспечения».

Модуль «Разработка и интеграция модулей программного обеспечения» включен в обязательную часть образовательной программы по направленности «Веб-разработка»

1.2 Перечень планируемых результатов освоения профессионального модуля

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

Требования к результатам освоения модуля

Код	Наименование вида деятельности и профессиональных компетенций
ВД 2	Разработка и интеграция модулей программного обеспечения
ПК 2.1.	Проектировать модули программного обеспечения
ПК 2.2.	Разрабатывать модули программного обеспечения
ПК 2.3.	Выполнять интеграцию модулей и компонентов программного обеспечения
ПК 2.4.	Выполнять тестирование и отладку программного обеспечения
ПК 2.5.	Осуществлять документирование программных модулей программного обеспечения

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 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 ПК, ОК	Результаты освоения		
	Владеет навыками	Умеет	Знает
ПК 2.1. Проектировать модули программного обеспечения	Н 2.1.1 проектирования модулей ПО с учетом требований заказчика	У 2.1.1 разрабатывать и внедрять системы защиты баз данных от несанкционированного доступа	З 2.1.1 основные принципы проектирования модулей программного обеспечения
	Н 2.1.2 создания архитектурных диаграмм и спецификаций модулей	У 2.1.2 создавать архитектурные диаграммы и документацию	З 2.1.2 языки программирования и технологии для реализации модулей
	Н 2.1.3 определения интерфейсов и взаимодействия модулей в системе	У 2.1.3 определять структуру и интерфейсы модулей	З 2.1.3 паттерны проектирования и структуры данных для создания эффективных и масштабируемых модулей

		У 2.1.4 анализировать требования к модулю и определять его функциональность	З 2.1.4 методы анализа требований и способов определения функциональности модуля
		У 2.1.5 проектировать архитектуру модуля, включая выбор подходящих паттернов проектирования и структуры данных	З 2.1.5 принципы создания интерфейсов для взаимодействия с другими модулями и системами
		У 2.1.6 создавать диаграммы классов, последовательностей и прочих диаграмм для визуализации проектируемого модуля	З 2.1.6 принципы обеспечения безопасности, производительности и масштабируемости при проектировании модулей
		У 2.1.7 выбирать подходящие языки программирования и технологии для реализации модуля	З 2.1.7 методы анализа и оптимизации проектируемых модулей для повышения их эффективности и качества
		У 2.1.8 проектировать интерфейсы программного обеспечения для взаимодействия с другими модулями и системами	
		У 2.1.9 учитывать требования к масштабируемости, производительности и безопасности при проектировании модуля	
		У 2.1.10 проводить анализ и оптимизацию проектируемого модуля для повышения его эффективности и качества	
ПК 2.2. Разрабатывать модули программного обеспечения	Н 2.2.1 создания модулей программного обеспечения на различных языках программирования	У 2.2.1 разрабатывать модули программного обеспечения с использованием различных языков программирования и технологий	З 2.2.1 язык программирования, основные конструкции, синтаксис
	Н 2.2.2 отладки и тестирования разработанных модулей	У 2.2.2 применять паттерны проектирования и структуры данных для создания эффективных и масштабируемых модулей	З 2.2.2 паттерны проектирования
	Н 2.2.3 применения структурного и объектно-ориентированного программирования	У 2.2.3 анализировать требования и определять функциональность модуля	З 2.2.3 структуры данных
	Н 2.2.4 оптимизации кода и алгоритмов программных модулей для увеличения производительности	У 2.2.4 создавать интерфейсы для взаимодействия с другими модулями и системами	З 2.2.4 принципы создания интерфейсов для взаимодействия с другими модулями и системами, таких как REST API, SOAP
	Н 2.2.5 мониторинга и	У 2.2.5 обеспечивать	З 2.2.5 работу с

	анализа производительности приложений	безопасность, производительность и масштабируемость при разработке модулей	инструментальным программным обеспечением
		У 2.2.6 оптимизировать проектируемые модули для повышения их эффективности и качества	З 2.2.6 методы оптимизации кода и алгоритмов
		У 2.2.7 работать с системой контроля версий	З 2.2.7 эффективные алгоритмы и структуры данных для повышения производительности
		У 2.2.8 улучшать производительность модулей, выявляя и устраняя узкие места	З 2.2.8 многопоточность в программных модулях
		У 2.2.9 проводить анализ и мониторинг производительности приложений	З 2.2.9 методы оптимизации сетевых протоколов для ускорения обмена данными
		У 2.2.10 применять инструменты для рефакторинга и оптимизации программного кода	З 2.2.10 кэширование данных
			З 2.2.11 управление памятью
			З 2.2.12 техники повышения производительности программного обеспечения
ПК 2.3 Выполнять интеграцию модулей и компонентов программного обеспечения	Н 2.3.1 интеграции программных модулей и компонентов в единое программное решение	У 2.3.1 интегрировать модули и компоненты, обеспечивая их взаимодействие	З 2.3.1 общие принципы функционирования аппаратных, программных и программно-аппаратных средств администрируемой информационно-коммуникационной системы
	Н 2.3.2 работы с API и веб-сервисами для взаимодействия между модулями	У 2.3.2 работать с API и устанавливать соединения между компонентами	З 2.3.2 международные стандарты локальных вычислительных сетей
	Н 2.3.3 работы с интеграционными платформами и инструментами	У 2.3.3 отслеживать и устранять конфликты и ошибки интеграции	З 2.3.3 методы и подходы к интеграции модулей и компонентов
	Н 2.3.4 обеспечения совместимости и стабильности системы	У 2.3.4 анализировать и определять зависимости между модулями и компонентами	З 2.3.4 принципы версионирования и управления изменениями при интеграции
		У 2.3.5 работать с различными форматами данных и протоколами передачи данных	З 2.3.5 принципы безопасности при интеграции модулей и компонентов
ПК 2.4 Выполнять тестирование и отладку программного обеспечения	Н 2.4.1 отладки программного обеспечения на уровне программных модулей	У 2.4.1 анализировать требования к программному обеспечению и составлять планы тестирования	З 2.4.1 принципы и методы тестирования программного обеспечения
	Н 2.4.2 тестирования программного обеспечения	У 2.4.2 создавать тестовые сценарии и тест-кейсы для проверки	З 2.4.2 основы программирования и архитектуры

		функциональности и соответствия требованиям	программного обеспечения
	Н 2.4.3 формирования тестовых сценариев	У 2.4.3 выполнять тестирование программного обеспечения вручную и автоматизировать процесс тестирования	З 2.4.3 основы баз данных и SQL-запросов
	Н 2.4.4 подготовки тестовых платформ (установка операционной системы, дополнительного ПО и другого по необходимости)	У 2.4.4 анализировать результаты тестирования и документировать найденные ошибки	З 2.4.4 инструменты для автоматизации тестирования
	Н 2.4.5 оценки объема тестирования ПО с целью определения необходимых ресурсов для его выполнения	У 2.4.5 разрабатывать стратегии отладки и исправлять ошибки в программном обеспечении	З 2.4.5 основы разработки и отладки программного обеспечения на разных языках программирования
	Н 2.4.6 настройки тестовой среды и аппаратных средств для выполнения тестирования ПО в соответствии с заданием на тестирование в пределах своей компетенции	У 2.4.6 выполнять модульные тесты с использованием инструментов тестирования, в том числе автоматизированного тестирования	З 2.4.6 понятие дефекта программного обеспечения
	Н 2.4.7 формирования и представления отчетности о подготовке к выполнению задания на тестирование ПО в соответствии с установленными регламентами	У 2.4.7 использовать системы контроля дефектов ПО	З 2.4.7 критерии качества ПО
	Н 2.4.8 выполнения тестовых процедур на тестовых данных	У 2.4.8 составлять отчет о выполнении тестирования ПО	З 2.4.8 виды и типы тестирования ПО
			З 2.4.9 техники ручного тестирования
			З 2.4.10 техники автоматизированного тестирования
			З 2.4.11 жизненный цикл дефекта ПО
			З 2.4.12 принципы работы в системе контроля дефектов
			З 2.4.13 основные понятия о качестве ПО
ПК 2.5 Осуществлять документирование программных модулей программного обеспечения	Н 2.5.1 создания технической документации для модулей	У 2.5.1 описывать функциональность модулей в документации	З 2.5.1 стандарты технической документации
	Н 2.5.2 документирования кода, API и интерфейсов	У 2.5.2 создавать диаграммы для иллюстрации работы модулей	З 2.5.2 принципы документирования программного обеспечения
	Н 2.5.3 работы со специализированным ПО по документированию программного кода	У 2.5.3 программировать с использованием комментариев для документирования кода	З 2.5.3 инструменты для создания технической документации и комментирования кода
		У 2.5.4 использовать специальные метки/теги для отметки важных	

		частей кода в документации	
		У 2.5.5 вести журнал изменений и фиксировать обновления программных модулей	
		У 2.5.6 разбивать модули на логические блоки и описывать каждый блок отдельно	
		У 2.5.7 включать в документацию особенности модулей, такие как ограничения, уязвимости или оптимальные настройки	
		У 2.5.8 проводить регулярное обновление документации при изменении модулей или добавлении нового функционала	
ОК 01. Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам		Уо 01.01 распознавать и анализировать задачу и/или проблему в профессиональном и/или социальном контексте; выделять её составные части; определять этапы решения задачи; составлять план действий; определять необходимые ресурсы; реализовывать составленный план;	Зо 01.01 актуальный профессиональный и социальный контекст, в котором приходится работать и жить;
ОК 02. Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности		Уо 02.02 использовать современное программное обеспечение, различные цифровые средства для решения профессиональных задач; проявлять культуру информационной безопасности при использовании информационно-коммуникационных технологий;	Зо 02.02 современные средства и устройства информатизации, порядок их применения и программное обеспечение в профессиональной деятельности в том числе с использованием цифровых средств;
ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста		Уо 05.02 грамотно излагать свои мысли и оформлять документы по профессиональной тематике на государственном языке;	
ОК 09. Пользоваться профессиональной документацией на государственном и иностранном языках		Уо 09.05 читать, понимать и находить необходимые технические данные и инструкции в руководствах в любом доступном формате	

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 2.2.3, У 2.2.6, У 2.2.7, У 2.2.10, Уо 01.01, З 2.2.3, З 2.2.6, З 2.2.7, Зо 01.01	Тема 1.2 Ключевые алгоритмы и структуры данных для выполнения задач программных модулей	21	Тема обеспечивает формирование компетенций в области выбора и применения эффективных алгоритмов и структур данных для оптимизации программных модулей. Изучение методов сортировки, поиска, работы с графами, хеш-таблицами и динамическим программированием необходимо для разработки производительных и масштабируемых решений, соответствующих современным требованиям индустрии.
	У 2.3.1, У 2.3.2, У 2.3.3, У 2.3.4, У 2.3.5, Уо 02.02; З 2.3.1, З 2.3.2, З 2.3.3, З 2.3.4, З 2.3.5, Зо 02.02	Тема 2.1. Основы интеграции программных модулей	60	Тема формирует компетенции в области проектирования и реализации клиент-серверного взаимодействия, разработки REST API, WebSocket, микросервисной архитектуры. Изучение принципов маршрутизации, аутентификации (JWT, OAuth 2.0), обработки ошибок и фоновых задач необходимо для создания современных распределённых систем и обеспечения бесшовной интеграции компонентов программного обеспечения.
	У 2.2.1, У 2.2.2, У 2.2.4, У 2.2.5, У 2.4.1, У 2.4.3, У 2.5.1, У 2.5.2, Уо 02.02; З 2.2.1, З 2.2.2, З 2.2.4, З 2.2.5, З 2.4.1, З	Тема 6.2. Разработка безопасного ПО и прикладная криптография	30	Тема обеспечивает формирование компетенций в области безопасной разработки

	2.4.3, 3 2.5.1, 3 2.5.2, 3 02.02			программного обеспечения: применение криптографических протоколов, защита от уязвимостей (SQLi, XSS, CSRF), реализация аутентификации и шифрования данных.
Всего академических часов профессионального модуля в рамках вариативной части			111	

1.4 Трудоемкость профессионального модуля

Наименование составных частей профессионального модуля	Объем в часах	В т.ч. в форме практической подготовки
Теоретические занятия	191	-
Практические занятия	<i>Не предусмотрено</i>	-
Лабораторные занятия	418	418
Курсовая работа (проект)	36	36
Консультации	<i>Не предусмотрено</i>	-
Самостоятельная работа	<i>Не предусмотрено</i>	-
Практика, в т.ч.:	324	324
учебная	180	180
производственная	144	144
Промежуточная аттестация	54	-
Всего	1023	778

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	с преподавателем						Промежуточная аттестация	
		Всего	в практической подготовке	лекции, уроки	практические занятия	лабораторные занятия			курсовой проект (работа)	Консультации						
3	4						5	6			7	8	9	10	11	12
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
ПК 2.1, ПК 2.2, ОК 01, ОК 02	Раздел 1. Разработка программных модулей			6к			153		153	102	51		102			
ПК 2.3, ОК 01, ОК 02, ОК 05, ОК 09	Раздел 2. Осуществление интеграции программных модулей	6			6		216		204	154	50	118		36		12
ПК 2.4, ОК 01, ОК 02	Раздел 3. Поддержка и тестирование программных модулей			8			72		72	54	18		54			
ПК 2.2, ПК 2.5, ОК 01, ОК 02	Раздел 4. Математическое моделирование			6к			72		72	54	18		54			
ПК 2.2, ПК 2.5 ОК 01, ОК 2	Раздел 5. Численные методы	5					72		60	30	30		30			12
ПК 2.2, ПК 2.4, ПК 2.5, ОК 01, ОК 02	Раздел 6. Безопасность программного обеспечения	7					102		84	60	24		60			18
ПК 2.1 – ПК 2.5, ОК 01, ОК 02	Учебная практика		6к 7				180		180	180						
ПК 2.1 – ПК 2.5, ОК 01, ОК 02, ОК 05, ОК 09	Производственная практика		6к 8				144		144	144						
ПК 2.1 – ПК 2.5, ОК 01, ОК 02, ОК 05, ОК 09	Экзамен по профессиональному модулю	8					12									12
	Всего	4	4	3	1		1023		969	778	191		418	36		54

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), учебная и 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
Раздел 1. Разработка программных модулей		153/102		
МДК.02.01 Разработка программных модулей		153/102		
Тема 1.1 Использование принципов объектно-ориентированного программирования при разработке программных модулей	Содержание	35/23		
	1. Модульная архитектура веб-приложений: принципы, преимущества и примеры реализации	2/0	ПК 2.1	3 2.1.1
	2. Архитектурные шаблоны, применяемые при разработке программных модулей (MVC, MVVM, MVP)	2/0	ПК 2.1	У 2.1.2, 3 2.1.1
	3. Инструменты разработки приложений с модульной архитектурой. Системы контроля версий	2/0	ПК 2.1, ПК 2.2	У 2.2.7, У 2.2.3, У 2.1.5, 3 2.1.2, 3 2.2.1
	4. Работа с библиотеками и пакетами. Коллекции, строки, дата и время в веб-среде	1/0	ПК 2.2, ОК 02	3 2.2.5, Уо 02.02, 3о 02.02
	5. Паттерны проектирования: отношения между классами и объектами (наследование, реализация, ассоциация, композиция, агрегация), интерфейсы, абстрактные классы, порождающие паттерны, паттерны поведения, структурные паттерны, поведенческие паттерны, паттерны объектов	1/0	ПК 2.2, ОК 02	У 2.2.2, 3 2.1.3, 3 2.2.3, Уо 02.02, 3о 02.02
	6. Система ввода-вывода, средства доступа к файлам и папкам файловой системы, чтения/записи	1/0	ПК 2.2, ОК 02	У 2.2.1, 3 2.2.1, 3 2.2.11, Уо 02.02
	7. Работа со строками, регулярными выражениями, кодирование/декодирование текста	1/0	ПК 2.2, ОК 01	У 2.2.2, 3 2.2.3, У 2.2.10, 3 2.2.12, Уо 01.01, 3о 01.01
	8. Асинхронное программирование в веб-среде: Event Loop, управление параллельными задачами и координация состояний	2/0	ПК 2.2, ОК 02	У 2.2.8, У 2.2.9, 3 2.2.8, 3 2.2.10, Уо

				02.02, 3o 02.02, Уo 02.02, 3o 02.02
	В том числе лабораторных занятий	23/23		
	Лабораторное занятие №1. Разработка модулей обработки массивов в JavaScript. Работа через систему контроля версий	3/3	ПК 2.2, ОК 02	У 2.2.7, 3 2.2.7, У 2.2.1, 3 2.2.3, Уo 02.02
	Лабораторное занятие №2. Проектирование модулей работы с коллекциями и объектами. Работа через систему контроля версий	3/3	ПК 2.1, ОК 02, ОК 01	У 2.2.7, 3 2.2.7, У 2.1.5, 3 2.1.3, Уo 01.01, 3o 01.01, Уo 02.02
	Лабораторное занятие №3. Создание утилит для работы с датой и временем в веб-среде. Настройка Git-репозитория для модульных задач	3/3	ПК 2.2, ОК 02	У 2.2.7, 3 2.2.7, У 2.2.1, Уo 02.02, 3o 02.02
	Лабораторное занятие №4. Реализация веб-модулей с применением паттернов проектирования. Code Review и слияние веток в Git	4/4	ПК 2.2, ОК 02	У 2.2.2, 3 2.2.2, У 2.2.7, 3 2.2.7, Уo 02.02, 3o 02.02
	Лабораторное занятие №5. Работа с файлами и потоками в браузере и Node.js. Клиентские хранилища данных	3/3	ПК 2.2, ОК 02	У 2.2.1, 3 2.2.1, 3 2.2.11, Уo 02.02, 3o 02.02
	Лабораторное занятие №6. Обработка текстовых данных: регулярные выражения, кодирование и валидация	4/4	ПК 2.2, ОК 01	У 2.2.2, 3 2.2.3, У 2.2.10, 3 2.2.12, Уo 01.01, 3o 01.01
	Лабораторное занятие №7. Асинхронное программирование в JavaScript: управление состояниями и отмена операций	3/3	ПК 2.2, ОК 02	У 2.2.8, 3 2.2.8, У 2.2.9, 3 2.2.10, Уo 02.02, 3o 02.02
Тема 1.2 Ключевые алгоритмы и структуры данных для выполнения задач программных модулей	Содержание	27/17		
	1. Основные структуры данных (массив, связный список, стек, очередь; операции вставки, поиска и удаления; представление данных в памяти)	2/0	ПК 2.2	У 2.2.3, 3 2.2.3
	2. Алгоритмы сортировки и поиска. Основы рекурсии: примеры, преимущества и недостатки	1/0	ПК 2.2, ОК 01	У 2.2.7, 3 2.2.7, Уo 01.01
	3. Хеш-таблица и хеш-функция. Коллизии и разрешение коллизий. Методы хеширования и сжатия данных.	2/0	ПК 2.2	У 2.2.7, 3 2.2.3

	Эффективность и применение хеш-структур			
	4. Деревья и графы. Представление графов и деревьев. Поиск в глубину и ширину. Минимум затратный путь (алгоритм Дейкстры). Деревья поиска и обхода	2/0	ПК 2.2	У 2.2.7, З 2.2.3
	5. Жадные алгоритмы и динамическое программирование. Основные идеи динамического программирования	1/0	ПК 2.2, ОК 01	У 2.2.6, З 2.2.6, Уо 01.01
	6. Алгоритмы обработки текстовых данных в веб-приложениях: поиск, сравнение, хеширование и эффективные структуры для строк	1/0	ПК 2.2	У 2.2.2, З 2.2.12
	7. Кучи и очереди. Очереди с приоритетом и кучи. Куча и ее применение	1/0	ПК 2.2	У 2.2.3, З 2.2.3
	В том числе лабораторных занятий	17/17		
	Лабораторное занятие №8. Оценка сложности алгоритмов	2/2	ПК 2.2, ОК 02	У 2.2.6, З 2.2.7, Уо 02.02
	Лабораторное занятие №9. Применение рекурсивных алгоритмов	2/2	ПК 2.2	У 2.2.10, З 2.2.6
	Лабораторное занятие №10. Работа с алгоритмами сортировки и поиска	2/2	ПК 2.2	У 2.2.7, З 2.2.7
	Лабораторное занятие №11. Создание хеш-таблиц и их использование для ускорения поиска данных	2/2	ПК 2.2	У 2.2.3, З 2.2.3
	Лабораторное занятие №12. Нахождение кратчайших путей в графах с использованием алгоритма Дейкстры	2/2	ПК 2.2	У 2.2.7, З 2.2.3
	Лабораторное занятие №13. Решение задачи о рюкзаке с использованием метода динамического программирования	2/2	ПК 2.2, ОК 01	У 2.2.6, З 2.2.6, Уо 01.01
	Лабораторное занятие №14. Реализация строковых алгоритмов	2/2	ПК 2.2	У 2.2.2, З 2.2.12
	Лабораторное занятие №15. Реализация приоритетных очередей для планирования задач	3/3	ПК 2.2	У 2.2.3, З 2.2.3
Тема 1.3. Проектирование модулей	Содержание	22/14		
	1. Основные принципы проектирования веб-модулей: архитектура компонентов и сервисов	2/0	ПК 2.1, ПК 2.2	У 2.1.5, З 2.1.1, У 2.2.2
	2. Анализ требований и декомпозиция веб-задач: проектирование, оптимизация и специфицирование модулей	1/0	ПК 2.1	У 2.1.4, З 2.1.4, У 2.1.10
	3. Принципы обеспечения безопасности, производительности и масштабируемости при проектировании модулей	2/0	ПК 2.1, ПК 2.2	У 2.1.9, З 2.1.6, У 2.2.5
	4. Принципы проектирования классов. Проектирование классов с учётом инкапсуляции. Использование наследования:	1/0	ПК 2.1, ПК 2.2	У 2.1.5, З 2.1.3, У 2.2.2

	создание иерархий классов. Полиморфизм: перегрузка методов и интерфейсов			
	5. Визуальное проектирование архитектуры веб-модулей: диаграммы компонентов, типов данных и потоков состояний	1/0	ПК 2.1	У 2.1.6, З 2.1.2
	6. Применение диаграмм компонентов для визуализации организации компонентов проектируемого модуля	1/0	ПК 2.1	У 2.1.6, З 2.1.2
	В том числе лабораторных занятий	14/14		
	Лабораторное занятие №16. Анализ требований и определение функциональных границ модуля	2/2	ПК 2.1	У 2.1.4, З 2.1.4
	Лабораторное занятие №17. Создание спецификации программного модуля	2/2	ПК 2.1	У 2.1.4, З 2.1.1
	Лабораторное занятие №18. Проектирование внутренней архитектуры веб-модуля: моделирование состояний, типов и применение паттернов	3/3	ПК 2.1, ПК 2.2	У 2.1.5, У 2.2.2, З 2.1.3
	Лабораторное занятие №19. Проектирование требований к организации компонентов модуля средствами диаграммы компонентов	2/2	ПК 2.1	У 2.1.6, З 2.1.2
	Лабораторное занятие №20. Проектирование интерфейсов взаимодействия веб-модулей и внешних сервисов	2/2	ПК 2.1	У 2.1.8, З 2.1.5
	Лабораторное занятие №21. Анализ метрик качества и оптимизация веб-модуля: производительность, бандл-сайз и рефакторинг	3/3	ПК 2.2	У 2.2.6, З 2.2.7
Тема 1.4. Создание программных модулей для взаимодействия с пользователем	Содержание	30/21		
	1. Виды пользовательского интерфейса. Основные этапы и принципы разработки графического пользовательского интерфейса	2/0	ПК 2.1, ОК 02	У 2.1.5, З 2.1.2, Уо 02.02
	2. Технологии и инструменты разработки графического пользовательского интерфейса.	1/0	ПК 2.2, ОК 02	У 2.2.1, З 2.2.5, Уо 02.02
	3. Компоненты графического пользовательского интерфейса. Типы элементов управления. Компонировка элементов управления. События. Обработчики событий.	1/0	ПК 2.2	У 2.2.1, З 2.2.1
	4. Работа с окнами. Основные методы работы с окнами. Создание окна: функции и классы. Открытие и закрытие окон. Взаимодействие с окнами. Примеры валидации. Сообщения об ошибках и уведомления пользователя. Использование регулярных выражений для валидации.	2/0	ПК 2.2, ОК 01	У 2.2.2, З 2.2.3, Уо 01.01

	5. Многопоточность и асинхронная работа окон. Многопоточность в GUI-приложениях. Проблемы синхронизации потоков. Использование асинхронных вызовов для долго выполнимых операций.	1/0	ПК 2.2, ОК 02	У 2.2.8, З 2.2.8, Уо 02.02
	6. Значение стиля в UX/UI дизайне. Основы теории цвета. Работа с цветом и шрифтами. Стилизация.	1/0	ПК 2.1, ОК 02	У 2.1.5, З 2.1.2, Уо 02.02
	7. Работа с текстом, изображениями. Построение графиков и диаграмм. Библиотеки для построения графиков и диаграмм. Работа с мультимедиа	1/0	ПК 2.2	У 2.2.1, З 2.2.5
	В том числе лабораторных занятий	21/21		
	Лабораторное занятие №22. Проектирование главного окна приложения с несколькими панелями и элементами управления.	2/2	ПК 2.2	У 2.2.1, З 2.2.1
	Лабораторное занятие №23. Разработка модулей многооконного приложения	2/2	ПК 2.2	У 2.2.1, З 2.2.1
	Лабораторное занятие №24. Разработка стилей для приложения для улучшения взаимодействия с пользователем	2/2	ПК 2.2, ОК 02	У 2.2.1, З 2.2.5, Уо 02.02
	Лабораторное занятие №25. Разработка модулей для представления текстовой информации	3/3	ПК 2.2, ОК 01	У 2.2.2, З 2.2.3, Уо 01.01
	Лабораторное занятие №26. Разработка модулей для работы с изображениями	2/2	ПК 2.2	У 2.2.1, З 2.2.5
	Лабораторное занятие №27. Разработка модулей для представления информации в виде графиков и диаграмм	2/2	ПК 2.2	У 2.2.1, З 2.2.5
	Лабораторное занятие №28. Разработка модулей для работы аудио и видео	2/2	ПК 2.2	У 2.2.1, З 2.2.5
	Лабораторное занятие №29. Реализация загрузки данных из интернета в фоновом режиме без блокировки основного потока приложения.	2/2	ПК 2.2, ОК 02	У 2.2.8, З 2.2.8, Уо 02.02
	Лабораторное занятие №30. Разработка формы регистрации с элементами ввода и проверкой корректности введенных данных.	4/4	ПК 2.2, ОК 01	У 2.2.2, З 2.2.3, Уо 01.01
Тема 1.5. Создание модулей для взаимодействия с базами данных	Содержание	17/12		
	1. Взаимодействие приложения с базой данных. Технологии доступа к данным. Безопасность при работе с базами данных.	2/0	ПК 2.2, ОК 02	У 2.2.5, З 2.2.1, Уо 02.02

	2. Понятие и преимущества ORM. Концепцией объектно-реляционного отображения и использование ORM-библиотек. Применение ORM для работы с базами данных.	2/0	ПК 2.2	У 2.2.1, 3 2.2.1
	3. Реализация CRUD-операций в приложении. Выполнение запросов к базе данных.	1/0	ПК 2.2	У 2.2.1, 3 2.2.1
	В том числе лабораторных занятий	12/12		
	Лабораторное занятие №31. Разработка программных модулей для работы с базами данных	6/6	ПК 2.2	У 2.2.1, 3 2.2.1
	Лабораторное занятие №32. Разработка программных модулей для работы с запросами к базе данных	6/6	ПК 2.2	У 2.2.1, 3 2.2.1
Тема 1.6. Принципы безопасности, производительности и масштабируемости программных модулей	Содержание	22/15		
	1. Основные понятия: Факторы, влияющие на производительность. Техники повышения производительности программного обеспечения Масштабируемость: горизонтальная и вертикальная масштабируемость; принципы проектирования для масштабируемости; использование облачных технологий для масштабирования.	4/0	ПК 2.1, ПК 2.2	У 2.1.9, У 2.2.5, 3 2.1.6
	2. Понятие оптимизации кода. Основные цели оптимизации.	3/0	ПК 2.2	У 2.2.6, 3 2.2.6
	В том числе лабораторных занятий	15/15		
	Лабораторное занятие №33. Оптимизация проектируемых модулей для повышения их эффективности и качества	4/4	ПК 2.2	У 2.2.6, 3 2.2.6
	Лабораторное занятие №34. Решение задач на оптимизацию алгоритмов	4/4	ПК 2.2, ОК 01	У 2.2.6, 3 2.2.7, Уо 01.01
	Лабораторное занятие №35. Анализ и мониторинг производительности приложений. Обеспечение производительности и масштабируемости при разработке модулей программного обеспечения	4/4	ПК 2.2	У 2.2.9, 3 2.2.9
	Лабораторное занятие №36. Улучшение производительности модулей посредством выявления и устранения узких мест	3/3	ПК 2.2	У 2.2.8, 3 2.2.11
Раздел 2. Осуществление интеграции программных модулей		216/154		
МДК.02.02 Осуществление интеграции программных модулей		216/154		
Тема 2.1. Основы интеграции	Содержание	105/79		
	1. Разработка REST API. Клиент-серверное взаимодействие. Особенности передачи информации по HTTP протоколу.	3/0	ПК 2.3, ОК 02	У 2.3.2, 3 2.3.3, Уо 02.02, 3о

программных модулей	Структура HTTP запроса. HTTP методы: GET, POST, DELETE, PUT, PATCH. HTTP заголовки. Тело запроса.			02.02
	2. Маршрутизация запросов. Группировка маршрутов. Статические ресурсы.	3/0	ПК 2.3	У 2.3.1, З 2.3.3
	3. Парсинг и валидация входящих веб-запросов: работа с параметрами, телом запроса и защита от некорректных данных	3/0	ПК 2.3, ОК 02	У 2.3.4, З 2.3.5, Уо 02.02, Зо 02.02
	4. Формирование HTTP-ответов: коды состояний, заголовки, cookies, сериализация данных и управление файловой отдачей	3/0	ПК 2.3	У 2.3.5, З 2.3.3
	5. Создание и управление фоновыми задачами	3/0	ПК 2.3, ОК 02	У 2.3.1, З 2.3.3, Уо 02.02
	6. Системы аутентификации и авторизации в веб-приложениях: JWT, OAuth 2.0, сессии и ролевой доступ (RBAC)	3/0	ПК 2.3, ОК 02	У 2.3.5, З 2.3.5, Уо 02.02, Зо 02.02
	7. Разработка WebSocket API. Взаимодействие клиента и сервера по WebSocket протоколу. Настройки соединения. Открытие и закрытие соединения. Передача сообщения серверу.	3/0	ПК 2.3, ОК 02	У 2.3.2, З 2.3.2, Уо 02.02
	8. Разработка микросервисов. Микросервисная и монолитная архитектура.	3/0	ПК 2.3, ОК 01	У 2.3.4, З 2.3.1, Уо 01.01, Зо 01.01
	9. Синхронное (REST, gRPC) и асинхронное (брокеры сообщений) взаимодействие между микросервисами.	2/0	ПК 2.3, ОК 02	У 2.3.5, З 2.3.3, З 2.3.4
	В том числе лабораторных занятий	79/79		
	Лабораторное занятие №37. Создание клиентского приложения для работы с публичным API	7/7	ПК 2.3, ОК 02	У 2.3.2, З 2.3.2, Уо 02.02
	Лабораторное занятие №38. Создание REST API приложения с реализацией: добавления, удаления, изменения и создания данных	7/7	ПК 2.3	У 2.3.1, З 2.3.3
	Лабораторное занятие №39. Расширение функционала REST API приложения: работа с удаленным источником данных	7/7	ПК 2.3	У 2.3.5, З 2.3.3
	Лабораторное занятие №40. Расширение функционала REST API приложения: работа со статическими изображениями (ресурсами) - загрузка, передача, удаление.	7/7	ПК 2.3	У 2.3.5, З 2.3.3
	Лабораторное занятие №41. Расширение функционала REST API приложения: обработка path и query параметров	6/6	ПК 2.3	У 2.3.4, З 2.3.3
	Лабораторное занятие №42. Расширение функционала REST API приложения: обработка ошибок, передача сообщений об	6/6	ПК 2.3	У 2.3.3, З 2.3.4

	ошибке пользователю			
	Лабораторное занятие №43. Расширение функционала REST API приложения: валидация полученных данных	7/7	ПК 2.3, ОК 02	У 2.3.4, З 2.3.5, Уо 02.02, Зо 02.02
	Лабораторное занятие №44. Расширение функционала REST API приложения: добавление фоновых задач	6/6	ПК 2.3, ОК 02	У 2.3.1, З 2.3.3, Уо 02.02
	Лабораторное занятие №45. Расширение функционала REST API приложения: добавление аутентификации и авторизации, создание ролевой системы	7/7	ПК 2.3, ОК 02	У 2.3.5, З 2.3.5, Уо 02.02, Зо 02.02
	Лабораторное занятие №46. Создание клиентского приложения для работы с публичным WebSocket.	6/6	ПК 2.3, ОК 02	У 2.3.2, З 2.3.2, Уо 02.02
	Лабораторное занятие №47. Создание серверного приложения для работы по websocket протоколу.	7/7	ПК 2.3	У 2.3.2, З 2.3.3
	Лабораторное занятие №48. Создание микросервисного приложения с взаимодействием по REST	6/6	ПК 2.3, ОК 01	У 2.3.1, У 2.3.4, З 2.3.3, Уо 01.01
Тема 2.2. Управление и мониторинг интегрированной системы	Содержание	22/13		
	1. Настройка конфигурации и сборки приложения.	3/0	ПК 2.3, ОК 02	У 2.3.1, З 2.3.4, Уо 02.02, Зо 02.02
	2. Мониторинг приложения: нагрузка, ошибки, сбор статистики. Внедрение сборщика метрик.	3/0	ПК 2.3, ОК 02	У 2.3.3, З 2.3.1, Уо 02.02
	3. Инструменты контейнеризации. Контейнеризация приложения. Средства доставки и средства развертывания решения.	3/0	ПК 2.3, ОК 02, ОК 09	У 2.3.1, З 2.3.4, Уо 02.02, Уо 09.05
	В том числе лабораторных занятий	13/13		
	Лабораторное занятие №49. Настройка конфигурации REST API приложения	7/7	ПК 2.3, ОК 02	У 2.3.1, З 2.3.4, Уо 02.02
	Лабораторное занятие №50. Упаковка REST API приложения в контейнер и доставка на другое устройство	6/6	ПК 2.3, ОК 02, ОК 05	У 2.3.1, З 2.3.4, Уо 02.02, Уо 05.02
Тема 2.3. Безопасность при интеграции	Содержание	19/13		
	1. Протоколы с использованием безопасного соединения: HTTPS, WSS (WebSocket Secure).	3/0	ПК 2.3, ОК 02	У 2.3.5, З 2.3.5, Уо 02.02, Зо 02.02
	2. Предотвращение угроз безопасности: SQL инъекции, CSRF, XSS. Хеширование чувствительных данных, применение	3/0	ПК 2.3, ОК 02,	З 2.3.5, Уо 02.02, Зо 02.02,

	алгоритмов хеширования паролей с солью.		ОК 01	Уо 01.01
	В том числе лабораторных занятий	13/13		
	Лабораторное занятие №51. Добавление SSL сертификата в приложение	7/7	ПК 2.3, ОК 02	У 2.3.5, З 2.3.5, Уо 02.02, Зо 02.02
	Лабораторное занятие №52. Настройка конфигурации безопасности приложения	6/6	ПК 2.3, ОК 02	У 2.3.3, З 2.3.5, Уо 02.02, Зо 02.02
Тема 2.4. Оптимизация и масштабируемость интегрированных решений	Содержание	22/13		
	1. Масштабирование интегрированных решений. Горизонтальное и вертикальное масштабирование.	3/0	ПК 2.3, ОК 01	У 2.3.4, З 2.3.1, Уо 01.01, Зо 01.01
	2. Оптимизации производительности. Кэширование данных. Оптимизация запросов к базам данных.	3/0	ПК 2.3, ОК 02	У 2.3.5, З 2.3.3, Уо 02.02
	3. Профилирование кода. Уменьшение времени отклика.	3/0	ПК 2.3, ОК 02	У 2.3.3, З 2.3.1, Уо 02.02
	В том числе лабораторных занятий	13/13		
	Лабораторное занятие №53. Реализация кэширования данных в REST API	7/7	ПК 2.3	У 2.3.5, З 2.3.3
	Лабораторное занятие №54. Оптимизация производительности REST API через профилирование	6/6	ПК 2.3, ОК 02	У 2.3.3, З 2.3.1, Уо 02.02
Курсовой проект. Тематика курсовых проектов.				
1. Развитие и интеграция HR-модуля в кадровую систему предприятия для автоматизации процессов найма и управления персоналом 2. Создание системы управления проектами с использованием модульной архитектуры 3. Разработка и интеграция модуля управления проектами в CRM-систему 4. Создание и интеграция платежного модуля для электронной коммерции 5. Создание платформы для обмена сообщениями 6. Интеграция разных баз данных в единую систему 7. Модульная архитектура и интеграция модулей в распределенной системе управления складскими запасами 8. Создание модуля аутентификации 9. Интеграция базы данных с модулем обработки данных 10. Создания и интеграция аналитического модуля для обработки данных в медицинской информационной системе 11. Разработка и интеграция образовательного модуля в LMS-систему				

12. Разработка и интеграция геолокационного модуля в систему управления транспортом 13. Разработка и интеграция нового модуля для системы управления складом				
Обязательные аудиторные учебные занятия по курсовому проекту (работе): 1. Ознакомление обучающихся с целями и задачами курсового проектирования с тематикой курсового проекта, выбор темы. 2. Ознакомление обучающихся с графиком выполнения курсового проекта, выдача задания и методических рекомендаций для выполнения курсового проекта. 3. Рекомендации по содержанию теоретической и практической частей курсового проекта. 4. Рекомендации по последовательности курсового проекта, требования к выполнению. 5. Рекомендации при анализе нормативно-правовых источников, ГОСТ, научных статей, монографий и т.п. 6. Анализ предметной области и разработка технического задания. Сбор и анализ требований к разрабатываемому программному продукту. 7. Разработка технического задания. 8. Проектирование программного продукта. 9. Разработка внешнего интерфейса и структуры меню. 10. Разработка связей между объектами. 11. Тестирование и отладка программного продукта. 12. Оформление основных разделов пояснительной записки. 13. Консультации по оформлению курсового проекта в соответствии с нормоконтролем. 14. Требования к презентационным материалам курсового проекта. 15. Защита курсового проекта.		36/36	ПК 2.1, ПК 2.2, ПК 2.3, ПК 2.4, ПК 2.5, ОК 01, ОК 02, ОК 05, ОК 09	У 2.1.4, У 2.1.5, У 2.1.6, У 2.1.8, У 2.1.10, У 2.2.1, У 2.2.2, У 2.2.6, У 2.2.7, У 2.3.1, У 2.3.2, У 2.3.4, У 2.3.5, У 2.4.3, У 2.4.6, У 2.5.1, У 2.5.2, У 2.5.3, З 2.1.1, З 2.1.2, З 2.1.3, З 2.1.4, З 2.1.5, З 2.2.1, З 2.2.2, З 2.2.3, З 2.2.6, З 2.2.7, З 2.3.3, З 2.3.4, З 2.3.5, З 2.4.1, З 2.4.2, З 2.5.1, З 2.5.2, Уо 01.01, Уо 02.02, Уо 02.02, Уо 05.02, Уо 09.05, Зо 01.01, Зо 02.02, Зо 02.02
Промежуточная аттестация МДК.02.02 Осуществление интеграции программных модулей Экзамен		12/12		
Раздел 3. Поддержка и тестирование программных модулей (72 часа)		72/54		
МДК.02.03 Поддержка и тестирование программных модулей		72/54		
Тема 3.1 Основы обеспечения качества ПО	Содержание	10/7		
	1. Понятие качества программного обеспечения. Стандарты качества (ISO 25010, ГОСТ Р ИСО/МЭК 9126). Модели качества.	1/0	ПК 2.4, ОК 02	З 2.4.1, З 2.4.7, З 2.4.13, Уо 02.02

	2. Метрики качества ПО: надежность, функциональность, удобство использования, эффективность, сопровождаемость, переносимость.	1/0	ПК 2.4	У 2.4.4, З 2.4.6
	3. Жизненный цикл дефекта. Классификация и типы дефектов. Способы документирования дефектов.	1/0	ПК 2.4	З 2.4.11, У 2.4.7
	В том числе лабораторных занятий	7/7		
	Лабораторное занятие №55. Анализ качества информационной системы с использованием метрик качества.	2/2	ПК 2.4, ОК 02	У 2.4.4, З 2.4.7, Уо 02.02
	Лабораторное занятие №56. Составление отчетов о дефектах. Работа с баг-трекингowymi системами (Jira, Redmine).	3/3	ПК 2.4	У 2.4.7, З 2.4.12
	Лабораторное занятие №57. Документирование дефектов и формирование отчета о тестировании.	2/2	ПК 2.4, ОК 01	У 2.4.8, З 2.4.8, Уо 01.01
Тема 3.2. Тест-дизайн и тестирование ПО	Содержание	13/11		
	1. Тестовый сценарий, тестовый план, чек-лист, тестовый пакет, задание на тестирование	1/0	ПК 2.4	З 2.4.2, З 2.4.8
	2. Классификация видов тестирования. Техники тест-дизайна: эквивалентное разбиение, анализ граничных значений, таблицы решений	1/0	ПК 2.4, ОК 01	У 2.4.2, З 2.4.9, З 2.4.10, Уо 01.01
	В том числе лабораторных занятий	11/11		
	Лабораторное занятие №58. Проектирование тест-кейсов для модульного тестирования.	2/2	ПК 2.4	У 2.4.2, З 2.4.8
	Лабораторное занятие №59. Разработка и выполнение автоматизированных тестов с использованием Selenium.	3/3	ПК 2.4, ОК 02	У 2.4.3, З 2.4.4, Уо 02.02
	Лабораторное занятие №60. Тестирование документации на ПО.	2/2	ПК 2.4	У 2.4.4, З 2.4.13
	Лабораторное занятие №61. Выполнение модульных тестов с использованием инструментов тестирования.	2/2	ПК 2.4	У 2.4.6, З 2.4.6
	Лабораторное занятие №62. Разработка тестов для проверки подключения баз данных	2/2	ПК 2.4, ОК 02	У 2.4.1, З 2.4.3, Уо 02.02
Тема 3.3 Регрессионное тестирование и восстановление после сбоев	Содержание	14/11		
	1. Регрессионное тестирование: цели, методы отбора тестов	1/0	ПК 2.4	З 2.4.8, У 2.4.5
	2. Восстановление тестов после сбоев. Причины сбоев, методы восстановления	1/0	ПК 2.4, ОК 02	У 2.4.5, З 2.4.5, Уо 02.02
	3. Проверка исправленных дефектов. Повторное тестирование	1/0	ПК 2.4	У 2.4.4, З 2.4.11
	В том числе лабораторных занятий	11/11		
	Лабораторное занятие №63. Выполнение регрессионного	2/2	ПК 2.4	У 2.4.3, У 2.4.4,

	тестирования и анализ результатов.			З 2.4.8
	Лабораторное занятие №64. Имитация сбоя и восстановление автоматизированных тестов.	3/3	ПК 2.4, ОК 02	У 2.4.5, З 2.4.10, Уо 02.02
	Лабораторное занятие №65. Проверка исправленных дефектов и оформление отчета.	2/2	ПК 2.4	У 2.4.8, З 2.4.11
	Лабораторное занятие №66. Настройка системы логирования и анализ логов.	2/2	ПК 2.4, ОК 02	У 2.4.7, З 2.4.4, Уо 02.02
	Лабораторное занятие №67. Составление сценариев поведения пользователей для регрессионного тестирования.	2/2	ПК 2.4	У 2.4.2, З 2.4.9
Тема 3.4 Модульное тестирование	Содержание	19/13		
	1. Средства модульного тестирования. Синтаксис языков программирования для модульных тестов.	1/0	ПК 2.4, ОК 02	З 2.4.4, З 2.4.10, У 2.4.6, Уо 02.02
	2. Отчет о тестировании. Системы контроля дефектов. Логирование. Сбор статистики.	1/0	ПК 2.4, ОК 02	З 2.4.4, З 2.4.10, У 2.4.6, Уо 02.02
	3. Область применения модульных автотестов. Библиотека тестов.	1/0	ПК 2.4, ОК 02	З 2.4.4, З 2.4.10, У 2.4.6, Уо 02.02
	4. Средства развертывания и интеграции автотестов	1/0	ПК 2.4, ОК 02	З 2.4.4, З 2.4.10, У 2.4.6, Уо 02.02
	5. Виды технической документации. Стандарты оформления.	1/0	ПК 2.4, ОК 02	З 2.4.4, З 2.4.10, У 2.4.6, Уо 02.02
	6. Инструменты и методы тестирования документации. Альфа- и бета-тестирование.	1/0	ПК 2.4, ОК 02	З 2.4.4, З 2.4.10, У 2.4.6, Уо 02.02
	В том числе лабораторных занятий	13/13		
	Лабораторное занятие №68. Разработка модульных автотестов для настольных приложений.	2/2	ПК 2.4	У 2.4.6, З 2.4.6
	Лабораторное занятие №69. Оформление отчетов о тестировании (создание библиотеки тестов).	3/3	ПК 2.4	У 2.4.8, З 2.4.8
	Лабораторное занятие №70. Запуск автотестов и сбор статистики. Оформление отчета.	2/2	ПК 2.4, ОК 02	У 2.4.3, У 2.4.8, Уо 02.02
	Лабораторное занятие №71. Сборка и запуск тестов из консоли.	2/2	ПК 2.4	У 2.4.3, З 2.4.4

	Лабораторное занятие №72. Выполнение приемочного тестирования и оформление отчета.	2/2	ПК 2.4	У 2.4.4, З 2.4.13
	Лабораторное занятие №73. Тестирование пользовательской документации и оформление отчета.	2/2	ПК 2.4, ОК 01	У 2.4.4, З 2.4.13, Уо 01.01
Тема 3.5 Основы интеграционного и системного тестирования	Содержание	16/10		
	1. Инструменты интеграционного тестирования. Особенности системного тестирования.	1/0	ПК 2.4, ОК 02	З 2.4.1, З 2.4.3, У 2.4.1, У 2.4.3, Уо 02.02
	2. Автоматизация тестирования UI. Инструменты API-тестирования.	1/0	ПК 2.4, ОК 02	З 2.4.1, З 2.4.3, У 2.4.1, У 2.4.3, Уо 02.02
	3. REST и SOAP. Типовые SQL-запросы для тестирования.	1/0	ПК 2.4, ОК 02	З 2.4.1, З 2.4.3, У 2.4.1, У 2.4.3, Уо 02.02
	4. Понятие заглушки. Имитаторы (Mock). Настройка тестового окружения.	1/0	ПК 2.4, ОК 02	З 2.4.1, З 2.4.3, У 2.4.1, У 2.4.3, Уо 02.02
	5. Фреймворки для тестирования кода. Настройка автоматического сбора логов.	1/0	ПК 2.4, ОК 02	З 2.4.1, З 2.4.3, У 2.4.1, У 2.4.3, Уо 02.02
	6. Регрессионное тестирование в жизненном цикле ПО.	1/0	ПК 2.4, ОК 02	З 2.4.1, З 2.4.3, У 2.4.1, У 2.4.3, Уо 02.02
	В том числе лабораторных занятий	10/10		
	Лабораторное занятие №74. Разработка и запуск тестов пользовательского интерфейса.	2/2	ПК 2.4, ОК 02	У 2.4.3, З 2.4.10, Уо 02.02
	Лабораторное занятие №75. Настройка автоматического запуска API-тестов.	2/2	ПК 2.4	У 2.4.3, З 2.4.4
	Лабораторное занятие №76. Разработка тестов для проверки подключения баз данных.	2/2	ПК 2.4, ОК 02	У 2.4.1, З 2.4.3, Уо 02.02
	Лабораторное занятие №77. Разработка автотеста с заглушками и имитаторами (моками).	2/2	ПК 2.4	У 2.4.2, З 2.4.10
	Лабораторное занятие №78. Настройка выбранной системы логирования с учетом ротации файлов.	1/1	ПК 2.4, ОК 02	У 2.4.7, З 2.4.4, Уо 02.02
	Лабораторное занятие №79. Анализ логов и подготовка отчета о результатах мониторинга.	1/1	ПК 2.4	У 2.4.4, У 2.4.8, З 2.4.7

Раздел 4. Математическое моделирование		72/54		
МДК.02.04 Математическое моделирование		72/54		
Тема 4.1. Математическое моделирование как методология решения практических задач	Содержание	40/30		
	1. Понятие решения. Множество решений, оптимальное решение. Показатель эффективности решения	1/0	ОК 01, ОК 02	Уо 01.01, Уо 02.02, Уо 02.02; Зо 01.01, Зо 02.02, Зо 02.02
	2. Математические модели, принципы их построения, виды моделей	1/0	ОК 01, ОК 02	Уо 01.01, Уо 02.02; Зо 01.01, Зо 02.02
	3. Задачи: классификация, методы решения, граничные условия. Общий вид и основная задача линейного программирования. Симплекс – метод.	2/0	ОК 01, ОК 02	Уо 01.01, Уо 02.02; Зо 01.01, Зо 02.02
	4. Транспортная задача. Методы нахождения начального решения транспортной задачи. Метод потенциалов.	1/0	ОК 01, ОК 02	Уо 01.01, Уо 02.02; Зо 01.01, Зо 02.02
	5. Общий вид задач нелинейного программирования. Графический метод решения задач нелинейного программирования. Метод множителей Лагранжа.	1/0	ОК 01, ОК 02	Уо 01.01, Уо 02.02; Зо 01.01, Зо 02.02
	6. Основные понятия динамического программирования: шаговое управление, управление операцией в целом, оптимальное управление, выигрыш на данном шаге, выигрыш за всю операцию, аддитивный критерий, мультипликативный критерий.	2/0	ОК 01, ОК 02	Уо 01.01, Уо 02.02; Зо 01.01, Зо 02.02
	7. Простейшие задачи, решаемые методом динамического программирования.	1/0	ОК 01, ОК 02	Уо 01.01, Уо 02.02; Зо 01.01, Зо 02.02
	8. Методы хранения графов в памяти ЭВМ. Задача о нахождении кратчайших путей в графе и методы ее решения. Задача о максимальном потоке и алгоритм Форда Фалкерсона.	1/0	ОК 01, ОК 02	Уо 01.01, Уо 02.02; Зо 01.01, Зо 02.02
	В том числе лабораторных занятий	30/30		
	Лабораторное занятие №80. Построение простейших математических моделей. Построение простейших статистических моделей	3/3	ПК 2.2, ОК 02	У 2.2.1, У 2.2.3, Уо 02.02; З 2.2.1, З 2.2.3, Зо 02.02

	Лабораторное занятие №81. Решение простейших однокритериальных задач	3/3	ПК 2.2, ОК 02	У 2.2.1, У 2.2.6, Уо 02.02; З 2.2.1, З 2.2.6, Зо 02.02
	Лабораторное занятие №82. Задача Коши для уравнения теплопроводности	3/3	ПК 2.2, ОК 02	У 2.2.1, У 2.2.5, Уо 02.02; З 2.2.1, З 2.2.5, Зо 02.02
	Лабораторное занятие №83. Сведение произвольной задачи линейного программирования к основной задаче линейного программирования»	3/3	ПК 2.2, ОК 02	У 2.2.1, У 2.2.3, Уо 02.02; З 2.2.1, З 2.2.3, Зо 02.02
	Лабораторное занятие №84. Решение задач линейного программирования симплекс–методом	3/3	ПК 2.2, ОК 02	У 2.2.1, У 2.2.6, Уо 02.02; З 2.2.1, З 2.2.6, Зо 02.02
	Лабораторное занятие №85. Нахождение начального решения транспортной задачи. Решение транспортной задачи методом потенциалов	3/3	ПК 2.2, ОК 02	У 2.2.1, У 2.2.8, Уо 02.02; З 2.2.1, З 2.2.8, Зо 02.02
	Лабораторное занятие №86. Применение метода стрельбы для решения линейной краевой задачи	3/3	ПК 2.2, ПК 2.5	У 2.2.4, У 2.5.3, У 2.5.5; З 2.2.4, З 2.5.2, З 2.5.3
	Лабораторное занятие №87. Задача о распределении средств между предприятиями	3/3	ПК 2.2, ПК 2.5	У 2.2.4, У 2.5.6, У 2.5.8; З 2.2.4, З 2.5.1, З 2.5.2
	Лабораторное занятие №88. Задача о замене оборудования	3/3	ПК 2.2, ПК 2.5	У 2.2.4, У 2.5.5, У 2.5.7; З 2.2.4, З 2.5.1, З 2.5.3
	Лабораторное занятие №89. Нахождение кратчайших путей в графе. Решение задачи о максимальном потоке	3/3	ПК 2.2, ПК 2.5	У 2.2.4, У 2.5.4, У 2.5.6; З 2.2.4, З 2.5.2, З 2.5.3
Тема 4.2 Задачи в условиях неопределенности	Содержание	32/24		
	1. Системы массового обслуживания: понятия, примеры, модели	2/0	ОК 01, ОК 02	Уо 01.01, Уо 02.02; Зо 01.01, Зо 02.02
	2. Основные понятия теории марковских процессов: случайный процесс, марковский процесс, граф состояний, поток событий,	2/0	ОК 01, ОК 02	Уо 01.01, Уо 02.02; Зо 01.01,

	вероятность состояния, уравнения Колмогорова, финальные вероятности состояний.			Зо 02.02
	3. Метод имитационного моделирования. Единичный жребий и формы его организации. Примеры задач.	1/0	ОК 01, ОК 02	Уо 01.01, Уо 02.02; Зо 01.01, Зо 02.02
	4. Предмет и задачи теории игр. Основные понятия теории игр: игра, игроки, партия, выигрыш, проигрыш, ход, личные и случайные ходы, стратегические игры, стратегия, оптимальная стратегия.	1/0	ОК 01, ОК 02	Уо 01.01, Уо 02.02; Зо 01.01, Зо 02.02
	5. Методы решения конечных игр: сведение игры mхn к задаче линейного программирования, численный метод – метод итераций.	1/0	ОК 01, ОК 02	Уо 01.01, Уо 02.02; Зо 01.01, Зо 02.02
	6. Область применимости теории принятия решений. Принятие решений в условиях определенности, в условиях риска, в условиях неопределенности.	1/0	ОК 01, ОК 02	Уо 01.01, Уо 02.02; Зо 01.01, Зо 02.02
	В том числе лабораторных занятий	24/24		
	Лабораторное занятие №90. Составление систем уравнений Колмогорова. Нахождение финальных вероятностей. Нахождение характеристик простейших систем массового обслуживания	4/4	ПК 2.2, ПК 2.5, ОК 01	У 2.2.8, У 2.5.2, Уо 01.01; 3 2.2.10, 3 2.5.1, Зо 01.01
	Лабораторное занятие №91. Решение задач массового обслуживания методами имитационного моделирования	4/4	ПК 2.2, ПК 2.5, ОК 01	У 2.2.8, У 2.5.3, Уо 01.01; 3 2.2.9, 3 2.5.3, Зо 01.01
	Лабораторное занятие №92. Построение прогнозов	4/4	ПК 2.2, ПК 2.5, ОК 01	У 2.2.8, У 2.5.5, Уо 01.01; 3 2.2.11, 3 2.5.1, Зо 01.01
	Лабораторное занятие №93. Решение матричной игры методом итераций	4/4	ПК 2.2, ПК 2.5, ОК 01	У 2.2.8, У 2.5.6, Уо 01.01; 3 2.2.12, 3 2.5.2, Зо 01.01
	Лабораторное занятие №94. Моделирование прогноза	4/4	ПК 2.2, ПК 2.5, ОК 01	У 2.2.8, У 2.5.7, Уо 01.01; 3 2.2.10, 3 2.5.1, Зо 01.01
	Лабораторное занятие №95. Выбор оптимального решения с помощью дерева решений	4/4	ПК 2.2, ПК 2.5, ОК 01	У 2.2.8, У 2.5.8, Уо 01.01; 3

				2.2.7, 3 2.5.2, 3o 01.01
Раздел 5. Численные методы		72/30		
МДК.02.05 Численные методы		72/30		
Тема 5.1 Актуализация основных понятий высшей математики	Содержание	8/4		
	1. Введение. Численные методы в программирование. Повторение основных понятий высшей математики	4/0	ОК 01, ОК 02	Уо 01.01, Уо 02.02, Уо 02.02; 3o 01.01, 3o 02.02, 3o 02.02
	В том числе лабораторных занятий	4/4		
	Лабораторное занятие №96 «Решение задач из основных разделов высшей математики»	4/4	ПК 2.2, ОК 02	У 2.2.1, У 2.2.3, Уо 02.02; 3 2.2.1, 3 2.2.3, 3o 02.02
Тема 5.2 Основные понятия теории погрешностей вычислений	Содержание	8/4		
	1. Типы погрешностей. Статистический и технический подходы к учету погрешностей.	4/0	ОК 01, ОК 02	Уо 01.01, Уо 02.02; 3o 01.01, 3o 02.02
	В том числе лабораторных занятий	4/4		
	Лабораторное занятие №97 «Решение простейших задач на вычисление погрешностей»	4/4	ПК 2.2, ОК 02	У 2.2.1, У 2.2.6, Уо 02.02; 3 2.2.1, 3 2.2.6, 3o 02.02
Тема 5.3 Численное решение СЛАУ	Содержание	10/5		
	1. Прямые методы (LU-метод, метод прогонки) Итерационные методы. Метод простой итерации, метод Зейделя	5/0	ОК 01, ОК 02	Уо 01.01, Уо 02.02; 3o 01.01, 3o 02.02
	В том числе лабораторных занятий	5/5		
	Лабораторное занятие №98 «Решение систем линейных уравнений»	5/5	ПК 2.2, ОК 02	У 2.2.1, У 2.2.4, Уо 02.02; 3 2.2.1, 3 2.2.4, 3o 02.02
Тема 5.4 Алгоритмы и методы поиска корней уравнения и решения нелинейных систем	Содержание	10/5		
	1. Поиск корней уравнения методом половинного деления, методом касательных, итерационным методом Решение систем нелинейных уравнений методом Ньютона, методом спуска.	5/0	ОК 01, ОК 02	Уо 01.01, Уо 02.02; 3o 01.01, 3o 02.02

	В том числе лабораторных занятий	5/5		
	Лабораторное занятие №99 «Решение алгебраических и трансцендентных уравнений»	5/5	ПК 2.2, ОК 02	У 2.2.1, У 2.2.6, Уо 02.02; З 2.2.1, З 2.2.6, Зо 02.02
Тема 5.5 Методы аналитического представления таблично заданной функции	Содержание	8/4		
	1. Интерполирование функции многочленами Лагранжа и Ньютона. Интерполирование функции многочленами Чебышева, тригонометрическая интерполяция, интерполяция сплайнами циклов.	4/0	ОК 01, ОК 02	Уо 01.01, Уо 02.02; Зо 01.01, Зо 02.02
	В том числе лабораторных занятий	4/4		
	Лабораторное занятие №100 «Интерполирование функции»	4/4	ПК 2.2, ПК 2.5	У 2.2.1, У 2.5.2; З 2.2.1, З 2.5.2
Тема 5.6 Алгоритмы и методы численного интегрирования и дифференцирования	Содержание	8/4		
	1. Численное дифференцирование Формулы дифференцирования первого и второго порядка Численное интегрирование Квадратурные формулы Ньютона-Котеса, Гаусса	4/0	ОК 01, ОК 02	Уо 01.01, Уо 02.02; Зо 01.01, Зо 02.02
	В том числе лабораторных занятий	4/4		
	Лабораторное занятие №101 «Численное интегрирование и дифференцирование»	4/4	ПК 2.2, ПК 2.5	У 2.2.1, У 2.5.4; З 2.2.1, З 2.5.3
Тема 5.7 Численные методы решения обыкновенных дифференциальных уравнений	Содержание	8/4		
	1. Метод Эйлера, метод Рунге-Кутты, решение задачи Коши Метод прогонки решения краевой задачи для ОДУ	4/0	ОК 01, ОК 02	Уо 01.01, Уо 02.02; Зо 01.01, Зо 02.02
	В том числе лабораторных занятий	4/4		
	Лабораторное занятие №102 «Численное решение дифференциальных уравнений»	4/4	ПК 2.2, ПК 2.5	У 2.2.1, У 2.5.6, У 2.5.8; З 2.2.1, З 2.5.1, З 2.5.2
Промежуточная аттестация МДК.02.05 Численные методы				
Экзамен				
Раздел 6. Безопасность программного обеспечения		102/60		
МДК.02.06 Безопасность программного обеспечения		102/60		
Тема 6.1	Содержание	54/40		

Основы безопасности программного обеспечения	1. Введение в кибербезопасность и уязвимости ПО	2/0	ОК 01, ОК 02	Уо 01.01, Уо 02.02; Зо 01.01, Зо 02.02
	2. Модели угроз и анализ рисков.	3/0	ОК 01, ОК 02	Уо 01.01, Уо 02.02, Уо 02.02; Зо 01.01, Зо 02.02, Зо 02.02
	3. Уязвимости веб-приложений: OWASP Top 10	3/0	ОК 01, ОК 02	Уо 01.01, Уо 02.02; Зо 01.01, Зо 02.02
	4. Криптография для разработчиков	3/0	ОК 01, ОК 02	Уо 01.01, Уо 02.02; Зо 01.01, Зо 02.02
	5. Безопасная аутентификация и авторизация	3/0	ОК 01, ОК 02	Уо 01.01, Уо 02.02; Зо 01.01, Зо 02.02
	В том числе лабораторных занятий	40/40		
	Лабораторное занятие №103 SQL инъекции - эксплуатация и защита уязвимого приложения	3/3	ПК 2.2, ПК 2.4	У 2.2.3, У 2.4.3; З 2.2.3, З 2.4.3
	Лабораторное занятие №104 XSS атаки - создание и предотвращение межсайтового скриптинга	3/3	ПК 2.2, ПК 2.4	У 2.2.5, У 2.4.5; З 2.2.5, З 2.4.5
	Лабораторное занятие №105 CSRF защита - реализация токенов и проверки Origin/Referer	2/2	ПК 2.2, ПК 2.5	У 2.2.4, У 2.5.2; З 2.2.4, З 2.5.2
	Лабораторное занятие №106 Составление модели угроз для типового веб-приложения	3/3	ПК 2.4, ПК 2.5	У 2.4.1, У 2.5.7; З 2.4.1, З 2.5.3
	Лабораторное занятие №107 Настройка безопасной аутентификации с JWT и refresh токенами	3/3	ПК 2.2, ОК 02	У 2.2.1, Уо 02.02; З 2.2.1, Зо 02.02
	Лабораторное занятие №108 Шифрование данных с использованием AES и RSA	3/3	ПК 2.2, ПК 2.4	У 2.2.6, У 2.4.2; З 2.2.6, З 2.4.2
	Лабораторное занятие №109 Хэширование паролей с salt и adaptive functions (bcrypt, Argon2)	2/2	ПК 2.2, ПК 2.5	У 2.2.8, У 2.5.3; З 2.2.8, З

				2.5.1
	Лабораторное занятие №110 Анализ сетевого трафика с помощью Wireshark	3/3	ПК 2.4, ОК 01	У 2.4.4, Уо 01.01; З 2.4.4, Зо 01.01
	Лабораторное занятие №111 Настройка HTTPS и создание самоподписанных сертификатов	3/3	ПК 2.2, ПК 2.4	У 2.2.2, У 2.4.6; З 2.2.2, З 2.4.6
	Лабораторное занятие №112 Защита от brute-force атак с ограничением попыток входа	3/3	ПК 2.2, ПК 2.4	У 2.2.5, У 2.4.7; З 2.2.5, З 2.4.7
	Лабораторное занятие №113 Безопасная работа с файлами	2/2	ПК 2.2, ПК 2.5	У 2.2.7, У 2.5.4; З 2.2.7, З 2.5.2
	Лабораторное занятие №114 Реализация безопасной десериализации данных	2/2	ПК 2.2, ПК 2.4	У 2.2.9, У 2.4.3; З 2.2.9, З 2.4.3
	Лабораторное занятие №115 Настройка CORS политик для веб-приложений	2/2	ПК 2.2, ПК 2.5	У 2.2.4, У 2.5.1; З 2.2.4, З 2.5.1
	Лабораторное занятие №116 Защита от DDOS атак с помощью rate limiting	2/2	ПК 2.2, ПК 2.4	У 2.2.6, У 2.4.8; З 2.2.6, З 2.4.8
	Лабораторное занятие №117 Безопасная работа с памятью в приложениях	2/2	ПК 2.2, ПК 2.4	У 2.2.10, У 2.4.5; З 2.2.11, З 2.4.5
	Лабораторное занятие №118 Создание безопасного API с валидацией всех входных данных	2/2	ПК 2.2, ПК 2.4	У 2.2.3, У 2.4.2; З 2.2.3, З 2.4.2
Тема 6.2. Разработка безопасного ПО и прикладная криптография	Содержание	30/20		
	Принципы безопасного проектирования архитектуры.	4/0	ОК 01, ОК 02	Уо 01.01, Уо 02.02; Зо 01.01, Зо 02.02
	Криптографические протоколы и их реализация.	3/0	ОК 01, ОК 02	Уо 01.01, Уо 02.02; Зо 01.01, Зо 02.02
	Криптография в веб-приложениях	3/0	ОК 01, ОК 02	Уо 01.01, Уо 02.02; Зо 01.01, Зо 02.02

	В том числе лабораторных занятий	20/20		
	Лабораторное занятие №119 Создание secure OAuth 2.0 провайдера с PKCE и защитой от атак	2/2	ПК 2.2, ПК 2.4	У 2.2.4, У 2.4.2; 3 2.2.4, 3 2.4.2
	Лабораторное занятие №120 Имплементация JWE (JSON Web Encryption) для защищённых токенов	2/2	ПК 2.2, ПК 2.5	У 2.2.1, У 2.5.2; 3 2.2.1, 3 2.5.2
	Лабораторное занятие №121 Разработка безопасного voting system с homomorphic encryption	2/2	ПК 2.2, ПК 2.4	У 2.2.3, У 2.4.3; 3 2.2.3, 3 2.4.3
	Лабораторное занятие №122 Реализация secure password manager с client-sWebStorm encryption	2/2	ПК 2.2, ПК 2.5	У 2.2.5, У 2.5.3; 3 2.2.5, 3 2.5.3
	Лабораторное занятие №123 Настройка HSM эмулятора для аппаратной защиты ключей	2/2	ПК 2.4, ПК 2.5	У 2.4.6, У 2.5.5; 3 2.4.6, 3 2.5.1
	Лабораторное занятие №124 Разработка secure API gateway с JWT verification и rate limiting	2/2	ПК 2.2, ПК 2.4	У 2.2.6, У 2.4.7; 3 2.2.6, 3 2.4.7
	Лабораторное занятие №125 Создание hardware-backed key storage для мобильного приложения	2/2	ПК 2.2, ПК 2.5	У 2.2.7, У 2.5.4; 3 2.2.7, 3 2.5.2
	Лабораторное занятие №126 Настройка certificate transparency logs для мониторинга SSL сертификатов	2/2	ПК 2.4, ПК 2.5	У 2.4.8, У 2.5.6; 3 2.4.8, 3 2.5.3
	Лабораторное занятие №127 Разработка secure session management с защитой от hijacking	2/2	ПК 2.2, ПК 2.4	У 2.2.8, У 2.4.4; 3 2.2.8, 3 2.4.4
	Лабораторное занятие №128 Создание cryptographically secure RNG (random number generator)	2/2	ПК 2.2, ПК 2.4	У 2.2.9, У 2.4.5; 3 2.2.10, 3 2.4.5
Промежуточная аттестация МДК.02.06 Безопасность программного обеспечения Экзамен		18/18		
Учебная практика Виды работ 1. Проектирование модулей программного обеспечения с учетом технического задания 2. Визуализации и описания архитектурных решений 3. Определение интерфейсов и взаимодействия модулей в системе		180/180	ПК 2.1 – ПК 2.5 ОК 01, ОК 02	Н 2.1.1 - Н 2.1.3, Н 2.2.1 – Н 2.2.5, Н 2.3.1 – Н 2.3.4, Н 2.4.1 – Н 2.4.8, Н 2.5.1 – Н 2.5.3,

4. Создание модулей программного обеспечения 5. Работа с API и веб-сервисами для взаимодействия между модулями 6. Работа с интеграционными платформами и инструментами 7. Отладка программного обеспечения на уровне программных модулей 8. Тестирование программного обеспечения 9. Формирование тестовых сценариев 10. Подготовка тестовых платформ (установка операционной системы, дополнительного программного обеспечения и другого по необходимости) 11. Оценка объема тестирования программного обеспечения с целью определения необходимых ресурсов для его выполнения 12. Формирование и представление отчетности о подготовке к выполнению задания на тестирование программного обеспечения в соответствии с установленными регламентами 13. Выполнение тестовых процедур на тестовых данных 14. Создание технической документации для модулей 15. Документирование кода, API и интерфейсов 16. Работа со специализированным программным обеспечением по документированию программного кода			Уо 01.01, Уо 02.02, Уо 02.02
Производственная практика Виды работ 1. Проектирование модулей программного обеспечения с учетом технического задания 2. Визуализации и описания архитектурных решений 3. Определение интерфейсов и взаимодействия модулей в системе 4. Создание модулей программного обеспечения 5. Оптимизация кода и алгоритмов программных модулей для увеличения производительности 6. Мониторинг и анализ производительности приложений 7. Интеграция программных модулей и компонентов в единое программное решение 8. Работа с API и веб-сервисами для взаимодействия между модулями 9. Работа с интеграционными платформами и инструментами 10. Обеспечение совместимости и стабильности системы 11. Отладка программного обеспечения на уровне программных модулей 12. Тестирование программного обеспечения 13. Формирование тестовых сценариев 14. Подготовка тестовых платформ (установка операционной системы, дополнительного программного обеспечения и другого по необходимости) 15. Оценка объема тестирования программного обеспечения с целью определения	144/144	ПК 2.1 – ПК 2.5 ОК 01, ОК 02	Н 2.1.1 - Н 2.1.3, Н 2.2.1 – Н 2.2.5, Н 2.3.1 – Н 2.3.4, Н 2.4.1 – Н 2.4.8, Н 2.5.1 – Н 2.5.3, Уо 01.01, Уо 02.02, Уо 02.02

необходимых ресурсов для его выполнения 16. Настройки тестовой среды и аппаратных средств для выполнения тестирования программного обеспечения в соответствии с заданием на тестирование в пределах своей компетенции 17. Формирование и представление отчетности о подготовке к выполнению задания на тестирование программного обеспечения в соответствии с установленными регламентами 18. Выполнение тестовых процедур на тестовых данных 19. Создание технической документации для модулей 20. Документирование кода, API и интерфейсов 21. Работа со специализированным программным обеспечением по документированию программного кода			
Промежуточная аттестация	12/0		
Экзамен по профессиональному модулю			
Всего	1023/778		

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.02.01 Разработка программных модулей		
Лабораторные занятия		
Лабораторное занятие №1. Разработка модулей обработки массивов в JavaScript. Работа через систему контроля версий	Формирование умений создавать и обрабатывать массивы в JavaScript, использовать Git для версионирования кода	ПК с доступом в интернет, WEBSTORM (VS Code/WebStorm), Node.js, Git, репозитории GitHub/GitLab
Лабораторное занятие №2. Проектирование модулей работы с коллекциями и объектами. Работа через систему контроля версий	Формирование умений работать с объектами и коллекциями, проектировать модульную архитектуру, использовать ветвление в Git	ПК, WEBSTORM, Git, Node.js, браузеры
Лабораторное занятие №3. Создание утилит для работы с датой и временем в веб-среде. Настройка Git-репозитория для модульных задач	Формирование умений разрабатывать утилиты для обработки дат/времени, настраивать удаленные репозитории и ветки	ПК, WEBSTORM, браузеры, Git, JS-библиотеки работы со временем
Лабораторное занятие №4. Реализация веб-модулей с применением паттернов проектирования. Code Review и слияние веток в Git	Формирование умений применять паттерны проектирования (MVC, Singleton и др.), проводить ревью кода и выполнять merge	ПК, WEBSTORM, Git
Лабораторное занятие №5. Работа с файлами и потоками в браузере и Node.js. Клиентские хранилища данных	Формирование умений читать/записывать файлы, работать с файловыми потоками и localStorage/sessionStorage	ПК, Node.js, браузеры, WEBSTORM, файловая система ОС
Лабораторное занятие №6. Обработка текстовых данных: регулярные выражения, кодирование и валидация	Формирование умений применять регулярные выражения, выполнять кодирование/декодирование и валидацию текста	ПК, WEBSTORM, онлайн-тестеры RegEx, браузеры
Лабораторное занятие №7. Асинхронное программирование в JavaScript: управление состояниями и отмена операций	Формирование умений работать с Promise, async/await, управлять асинхронными потоками и отменять операции	ПК, Node.js/браузер, WEBSTORM, инструменты отладки
Лабораторное занятие №8. Оценка сложности алгоритмов	Формирование умений анализировать временную и пространственную сложность алгоритмов (Big O notation)	ПК, WEBSTORM, инструменты профилирования, визуализаторы алгоритмов

Лабораторное занятие №9. Применение рекурсивных алгоритмов	Формирование умений проектировать и реализовывать рекурсивные решения задач, отслеживать стек вызовов	ПК, WEBSTORM, отладчик, языки программирования
Лабораторное занятие №10. Работа с алгоритмами сортировки и поиска	Формирование умений реализовывать и сравнивать алгоритмы сортировки и поиска, анализировать их эффективность	ПК, WEBSTORM, тестовые наборы данных, визуализаторы
Лабораторное занятие №11. Создание хеш-таблиц и их использование для ускорения поиска данных	Формирование умений реализовывать хеш-таблицы, обрабатывать коллизии, применять их для оптимизации поиска	ПК, WEBSTORM, языки программирования, профайлеры
Лабораторное занятие №12. Нахождение кратчайших путей в графах с использованием алгоритма Дейкстры	Формирование умений представлять графы, реализовывать алгоритм Дейкстры и находить оптимальные пути	ПК, WEBSTORM, графические редакторы/визуализаторы графов
Лабораторное занятие №13. Решение задачи о рюкзаке с использованием метода динамического программирования	Формирование умений применять динамическое программирование для решения оптимизационных задач	ПК, WEBSTORM, инструменты отладки, таблицы
Лабораторное занятие №14. Реализация строковых алгоритмов	Формирование умений реализовывать алгоритмы поиска и обработки подстрок (KMP, Z-алгоритм и др.)	ПК, WEBSTORM, тестовые наборы строк
Лабораторное занятие №15. Реализация приоритетных очередей для планирования задач	Формирование умений создавать и использовать приоритетные очереди (кучи) для управления задачами	ПК, WEBSTORM, языки программирования
Лабораторное занятие №16. Анализ требований и определение функциональных границ модуля	Формирование умений анализировать технические требования, декомпозировать задачи и определять границы модуля	ПК, системы управления требованиями (Jira/Confluence), WEBSTORM
Лабораторное занятие №17. Создание спецификации программного модуля	Формирование умений разрабатывать техническую спецификацию модуля согласно стандартам оформления	ПК, офисное ПО, системы документации (Swagger/Markdown)
Лабораторное занятие №18. Проектирование внутренней архитектуры веб-модуля: моделирование состояний, типов и применение паттернов	Формирование умений проектировать архитектуру модуля, описывать типы данных и состояния системы	ПК, CASE-средства, UML-редакторы, WEBSTORM
Лабораторное занятие №19. Проектирование требований к организации компонентов модуля средствами диаграммы компонентов	Формирование умений визуализировать структуру модуля с помощью диаграмм компонентов UML	ПК, UML-инструменты (Draw.io/PlantUML), WEBSTORM
Лабораторное занятие №20. Проектирование	Формирование умений проектировать API-	ПК, Postman, Swagger/OpenAPI, WEBSTORM

интерфейсов взаимодействия веб-модулей и внешних сервисов	контракты и схемы взаимодействия модулей	
Лабораторное занятие №21. Анализ метрик качества и оптимизация веб-модуля: производительность, бандл-сайз и рефакторинг	Формирование умений анализировать метрики кода, уменьшать размер сборки и проводить рефакторинг	ПК, сборщики (Webpack/Vite), профилировщики, WEBSTORM
Лабораторное занятие №22. Проектирование главного окна приложения с несколькими панелями и элементами управления	Формирование умений верстать и программировать многопанельные интерфейсы с элементами управления	ПК, браузеры, WEBSTORM, UI-фреймворки
Лабораторное занятие №23. Разработка модулей многооконного приложения	Формирование умений управлять окнами, обрабатывать события открытия/закрытия и обмена данными между окнами	ПК, WEBSTORM, фреймворки для десктопных/веб-приложений
Лабораторное занятие №24. Разработка стилей для приложения для улучшения взаимодействия с пользователем	Формирование умений применять CSS/SCSS, реализовывать адаптивный дизайн и улучшать UX	ПК, браузеры с DevTools, WEBSTORM, графические редакторы
Лабораторное занятие №25. Разработка модулей для представления текстовой информации	Формирование умений форматировать, отображать и валидировать текстовые данные в интерфейсе	ПК, WEBSTORM, браузеры, UI-библиотеки
Лабораторное занятие №26. Разработка модулей для работы с изображениями	Формирование умений загружать, отображать, масштабировать и оптимизировать изображения в приложениях	ПК, WEBSTORM, графические редакторы, браузеры
Лабораторное занятие №27. Разработка модулей для представления информации в виде графиков и диаграмм	Формирование умений интегрировать библиотеки визуализации данных и строить интерактивные графики	ПК, WEBSTORM, библиотеки (Chart.js/D3.js), браузеры
Лабораторное занятие №28. Разработка модулей для работы аудио и видео	Формирование умений внедрять медиаплееры, управлять потоками и обрабатывать аудио/видео события	ПК, WEBSTORM, медиафайлы, браузеры
Лабораторное занятие №29. Реализация загрузки данных из интернета в фоновом режиме без блокировки основного потока приложения	Формирование умений использовать Web Workers, асинхронные запросы и управление фоновыми задачами	ПК, WEBSTORM, браузеры, инструменты мониторинга сети
Лабораторное занятие №30. Разработка формы регистрации с элементами ввода и проверкой корректности введенных данных	Формирование умений создавать формы, валидировать данные на клиенте и сервере, обрабатывать ошибки	ПК, WEBSTORM, браузеры, серверная среда (Node.js/PHP)
Лабораторное занятие №31. Разработка	Формирование умений устанавливать	ПК, СУБД (PostgreSQL/MySQL),

программных модулей для работы с базами данных	соединения с СУБД, выполнять базовые CRUD-операции через ORM/драйверы	WEBSTORM, ORM-библиотеки
Лабораторное занятие №32. Разработка программных модулей для работы с запросами к базе данных	Формирование умений составлять оптимизированные SQL-запросы, использовать транзакции и индексы	ПК, СУБД, клиенты БД (DBeaver), WEBSTORM
Лабораторное занятие №33. Оптимизация проектируемых модулей для повышения их эффективности и качества	Формирование умений выявлять узкие места, рефакторить код и применять паттерны оптимизации	ПК, профилировщики, WEBSTORM, линтеры
Лабораторное занятие №34. Решение задач на оптимизацию алгоритмов	Формирование умений применять алгоритмические техники для снижения вычислительной сложности	ПК, WEBSTORM, алгоритмические тренажеры
Лабораторное занятие №35. Анализ и мониторинг производительности приложений. Обеспечение производительности и масштабируемости при разработке модулей программного обеспечения	Формирование умений настраивать мониторинг, анализировать метрики и проектировать масштабируемые модули	ПК, АРМ-инструменты (Grafana/Prometheus), WEBSTORM, облачные среды
Лабораторное занятие №36. Улучшение производительности модулей посредством выявления и устранения узких мест	Формирование умений проводить профилирование, устранять утечки памяти и оптимизировать сетевые запросы	ПК, профилировщики, браузер DevTools, WEBSTORM
МДК.02.02 Осуществление интеграции программных модулей		
Лабораторные занятия		
Лабораторное занятие №37. Создание клиентского приложения для работы с публичным API	Формирование умений выполнять HTTP-запросы, парсить JSON и отображать данные из внешних API	ПК, браузер/WEBSTORM, Postman, публичные API
Лабораторное занятие №38. Создание REST API приложения с реализацией: добавления, удаления, изменения и создания данных	Формирование умений проектировать REST-эндпоинты и реализовывать CRUD-операции на сервере	ПК, серверная среда (Node.js/Python/Java), WEBSTORM, Postman
Лабораторное занятие №39. Расширение функционала REST API приложения: работа с удаленным источником данных	Формирование умений интегрировать API с внешними сервисами и обрабатывать асинхронные ответы	ПК, WEBSTORM, HTTP-клиенты, СУБД
Лабораторное занятие №40. Расширение функционала REST API приложения: работа со статическими изображениями (ресурсами) - загрузка, передача, удаление	Формирование умений настраивать обработку multipart/form-data, хранить и отдавать медиафайлы	ПК, серверная среда, файловые хранилища, WEBSTORM

Лабораторное занятие №41. Расширение функционала REST API приложения: обработка path и query параметров	Формирование умений валидировать и использовать параметры маршрутизации и запроса	ПК, WEBSTORM, Postman, фреймворки маршрутизации
Лабораторное занятие №42. Расширение функционала REST API приложения: обработка ошибок, передача сообщений об ошибке пользователю	Формирование умений проектировать единую схему обработки ошибок и возвращать стандартные HTTP-коды	ПК, WEBSTORM, серверная среда, клиентские приложения
Лабораторное занятие №43. Расширение функционала REST API приложения: валидация полученных данных	Формирование умений валидировать входные данные с использованием библиотек и схем	ПК, WEBSTORM, библиотеки валидации, Postman
Лабораторное занятие №44. Расширение функционала REST API приложения: добавление фоновых задач	Формирование умений внедрять очереди задач и асинхронные обработчики (cron, bull, celery)	ПК, серверная среда, брокеры сообщений (Redis/RabbitMQ), WEBSTORM
Лабораторное занятие №45. Расширение функционала REST API приложения: добавление аутентификации и авторизации, создание ролевой системы	Формирование умений реализовывать JWT/OAuth, проверять токены и управлять доступом по ролям	ПК, WEBSTORM, серверная среда, СУБД
Лабораторное занятие №46. Создание клиентского приложения для работы с публичным WebSocket	Формирование умений устанавливать WebSocket-соединения, обрабатывать события в реальном времени	ПК, браузеры/WEBSTORM, WebSocket-тестеры
Лабораторное занятие №47. Создание серверного приложения для работы по websocket протоколу	Формирование умений настраивать WebSocket-сервер, управлять комнатами и рассылкой сообщений	ПК, серверная среда, WEBSTORM, клиенты тестирования
Лабораторное занятие №48. Создание микросервисного приложения с взаимодействием по REST	Формирование умений разделять монолит на микросервисы и настраивать межсервисное взаимодействие	ПК, Docker, WEBSTORM, API-шлюзы, сервисы обнаружения
Лабораторное занятие №49. Настройка конфигурации REST API приложения	Формирование умений управлять конфигурационными файлами, переменными окружения и профилями запуска	ПК, WEBSTORM, серверная среда, системы конфигурации (.env/YAML)
Лабораторное занятие №50. Упаковка REST API приложения в контейнер и доставка на другое устройство	Формирование умений писать Dockerfile, собирать образы и деплоить приложение в изолированном окружении	ПК, Docker/Docker Compose, регистры контейнеров, терминал
Лабораторное занятие №51. Добавление SSL сертификата в приложение	Формирование умений генерировать самоподписанные сертификаты, настраивать HTTPS и перенаправления	ПК, OpenSSL, веб-серверы (Nginx/Apache), WEBSTORM

Лабораторное занятие №52. Настройка конфигурации безопасности приложения	Формирование умений применять заголовки безопасности (CORS, CSP, HSTS) и настраивать политики доступа	ПК, серверная среда, WEBSTORM, инструменты сканирования безопасности
Лабораторное занятие №53. Реализация кэширования данных в REST API	Формирование умений внедрять кэширование на уровне сервера и клиента, управлять инвалидацией кэша	ПК, Redis/Memcached, WEBSTORM, серверная среда
Лабораторное занятие №54. Оптимизация производительности REST API через профилирование	Формирование умений анализировать время отклика, оптимизировать запросы БД и настраивать пулы соединений	ПК, профилировщики, АРМ-инструменты, WEBSTORM
МДК.02.03 Поддержка и тестирование программных модулей		
Лабораторные занятия		
Лабораторное занятие №55. Анализ качества информационной системы с использованием метрик качества	Формирование умений измерять метрики надежности, сопровождаемости и эффективности ПО	ПК, системы статического анализа (SonarQube), WEBSTORM, отчетные инструменты
Лабораторное занятие №56. Составление отчетов о дефектах. Работа с баг-трекинговыми системами (Jira, Redmine)	Формирование умений регистрировать, классифицировать и отслеживать жизненный цикл дефектов	ПК, баг-трекеры, браузеры, системы управления проектами
Лабораторное занятие №57. Документирование дефектов и формирование отчета о тестировании	Формирование умений составлять тестовые отчеты, фиксировать шаги воспроизведения и ожидаемые результаты	ПК, офисное ПО, системы тест-менеджмента
Лабораторное занятие №58. Проектирование тест-кейсов для модульного тестирования	Формирование умений применять техники тест-дизайна и создавать покрывающие тест-кейсы	ПК, тест-фреймворки, WEBSTORM, чек-листы
Лабораторное занятие №59. Разработка и выполнение автоматизированных тестов с использованием Selenium	Формирование умений писать UI-автотесты, настраивать браузерные драйверы и запускать сценарии	ПК, Selenium WebDriver, браузеры, WEBSTORM
Лабораторное занятие №60. Тестирование документации на ПО	Формирование умений проверять полноту, актуальность и соответствие документации требованиям	ПК, офисное ПО, системы контроля версий, чек-листы
Лабораторное занятие №61. Выполнение модульных тестов с использованием инструментов тестирования	Формирование умений запускать юнит-тесты, анализировать покрытие кода и устранять сбои	ПК, тест-фреймворки (JUnit/pytest/Jest), WEBSTORM, CI/CD
Лабораторное занятие №62. Разработка тестов для проверки подключения баз данных	Формирование умений тестировать соединения, транзакции и целостность данных в СУБД	ПК, СУБД, тестовые фреймворки, WEBSTORM

Лабораторное занятие №63. Выполнение регрессионного тестирования и анализ результатов	Формирование умений отбирать регрессионные сценарии, запускать их и анализировать отклонения	ПК, тестовые среды, CI-системы, отчетные инструменты
Лабораторное занятие №64. Имитация сбоя и восстановление автоматизированных тестов	Формирование умений моделировать отказы сервисов и проверять устойчивость тестовых сценариев	ПК, инструменты мокирования, тестовые фреймворки, WEBSTORM
Лабораторное занятие №65. Проверка исправленных дефектов и оформление отчета	Формирование умений проводить повторное тестирование, подтверждать исправления и фиксировать результаты	ПК, баг-трекеры, тестовые среды, офисное ПО
Лабораторное занятие №66. Настройка системы логирования и анализ логов	Формирование умений настраивать уровни логирования, агрегировать журналы и искать аномалии	ПК, лог-менеджеры (ELK/Graylog), терминал, WEBSTORM
Лабораторное занятие №67. Составление сценариев поведения пользователей для регрессионного тестирования	Формирование умений моделировать пользовательские сценарии и интегрировать их в регрессионные наборы	ПК, тест-менеджмент системы, WEBSTORM, браузеры
Лабораторное занятие №68. Разработка модульных автотестов для настольных приложений	Формирование умений писать и запускать автотесты для GUI-компонентов десктопных приложений	ПК, тестовые фреймворки для десктопа, WEBSTORM, эмуляторы
Лабораторное занятие №69. Оформление отчетов о тестировании (создание библиотеки тестов)	Формирование умений структурировать тестовую базу, формировать отчеты и вести архивы тестов	ПК, системы отчетности, репозитории тестов, офисное ПО
Лабораторное занятие №70. Запуск автотестов и сбор статистики. Оформление отчета	Формирование умений автоматизировать запуск тестов, собирать метрики покрытия и формировать дашборды	ПК, CI/CD (Jenkins/GitHub Actions), тестовые фреймворки, браузеры
Лабораторное занятие №71. Сборка и запуск тестов из консоли	Формирование умений использовать CLI-инструменты для сборки проекта и запуска тестовых наборов	ПК, терминал, CLI-утилиты, WEBSTORM
Лабораторное занятие №72. Выполнение приемочного тестирования и оформление отчета	Формирование умений проверять соответствие ПО бизнес-требованиям и оформлять акты приемки	ПК, тестовые среды, чек-листы, офисное ПО
Лабораторное занятие №73. Тестирование пользовательской документации и оформление отчета	Формирование умений оценивать понятность, полноту и точность пользовательских руководств	ПК, офисное ПО, чек-листы, системы управления контентом
Лабораторное занятие №74. Разработка и запуск тестов пользовательского интерфейса	Формирование умений создавать автоматизированные UI-тесты, валидировать	ПК, UI-фреймворки (Playwright/Cypress), WEBSTORM, браузеры

	элементы и события	
Лабораторное занятие №75. Настройка автоматического запуска API-тестов	Формирование умений интегрировать API-тесты в пайплайн CI и настраивать триггеры запуска	ПК, Postman/Newman, CI-системы, WEBSTORM
Лабораторное занятие №76. Разработка тестов для проверки подключения баз данных	Формирование умений тестировать ORM-слой, валидировать схемы и проверять транзакционную целостность	ПК, СУБД, тестовые фреймворки, WEBSTORM
Лабораторное занятие №77. Разработка автотеста с заглушками и имитаторами (моками)	Формирование умений изолировать тестируемые модули с помощью mock-объектов и заглушек сервисов	ПК, библиотеки мокирования, WEBSTORM, тестовые среды
Лабораторное занятие №78. Настройка выбранной системы логирования с учетом ротации файлов	Формирование умений настраивать ротацию логов, уровни записи и политики хранения	ПК, лог-фреймворки (Log4j/Winston), серверная среда, терминал
МДК.02.04 Математическое моделирование		
Лабораторные занятия		
Лабораторное занятие №80. Построение простейших математических моделей. Построение простейших статистических моделей	Формирование умений формализовать задачи, выбирать модели и строить статистические распределения	ПК, математические пакеты (Python/NumPy, MATLAB, Excel), WEBSTORM
Лабораторное занятие №81. Решение простейших однокритериальных задач	Формирование умений применять методы оптимизации для задач с одним критерием эффективности	ПК, математическое ПО, WEBSTORM, визуализаторы
Лабораторное занятие №82. Задача Коши для уравнения теплопроводности	Формирование умений численно решать дифференциальные уравнения в частных производных	ПК, математические среды, WEBSTORM, графические пакеты
Лабораторное занятие №83. Сведение произвольной задачи линейного программирования к основной задаче линейного программирования	Формирование умений преобразовывать ограничения и целевые функции к каноническому виду	ПК, решатели ЛП, WEBSTORM, таблицы
Лабораторное занятие №84. Решение задач линейного программирования симплекс–методом	Формирование умений реализовывать симплекс-алгоритм и анализировать оптимальные решения	ПК, математическое ПО, WEBSTORM, алгоритмические тренажеры
Лабораторное занятие №85. Нахождение начального решения транспортной задачи. Решение транспортной задачи методом	Формирование умений строить опорные планы и оптимизировать перевозки методом потенциалов	ПК, Excel/математические пакеты, WEBSTORM, визуализаторы графов

потенциалов		
Лабораторное занятие №86. Применение метода стрельбы для решения линейной краевой задачи	Формирование умений применять метод стрельбы для решения краевых задач ОДУ	ПК, математические среды, WEBSTORM, инструменты визуализации
Лабораторное занятие №87. Задача о распределении средств между предприятиями	Формирование умений применять динамическое программирование для оптимизации распределения ресурсов	ПК, математическое ПО, WEBSTORM, таблицы
Лабораторное занятие №88. Задача о замене оборудования	Формирование умений моделировать процессы замены оборудования и находить оптимальные стратегии	ПК, математические пакеты, WEBSTORM, системы визуализации
Лабораторное занятие №89. Нахождение кратчайших путей в графе. Решение задачи о максимальном потоке	Формирование умений реализовывать алгоритмы поиска путей и расчета потоков в сетях	ПК, алгоритмические среды, WEBSTORM, графические редакторы
Лабораторное занятие №90. Составление систем уравнений Колмогорова. Нахождение финальных вероятностей. Нахождение характеристик простейших систем массового обслуживания	Формирование умений моделировать СМО, составлять и решать уравнения Колмогорова	ПК, математическое ПО, WEBSTORM, симуляторы СМО
Лабораторное занятие №91. Решение задач массового обслуживания методами имитационного моделирования	Формирование умений разрабатывать имитационные модели СМО и анализировать их характеристики	ПК, среды имитационного моделирования (AnyLogic/Python), WEBSTORM
Лабораторное занятие №92. Построение прогнозов	Формирование умений применять методы временных рядов и строить прогнозные модели	ПК, аналитические пакеты, WEBSTORM, системы визуализации данных
Лабораторное занятие №93. Решение матричной игры методом итераций	Формирование умений находить оптимальные стратегии и равновесия в матричных играх	ПК, математическое ПО, WEBSTORM, таблицы
Лабораторное занятие №94. Моделирование прогноза	Формирование умений валидировать прогнозные модели и оценивать погрешности предсказаний	ПК, аналитические среды, WEBSTORM, инструменты статистики
МДК.02.05 Численные методы		
Лабораторные занятия		
Лабораторное занятие №96. Решение задач из основных разделов высшей математики	Формирование умений применять численные методы для решения задач линейной алгебры и анализа	ПК, математические пакеты, WEBSTORM, калькуляторы/ПО

Лабораторное занятие №97. Решение простейших задач на вычисление погрешностей	Формирование умений оценивать абсолютные и относительные погрешности вычислений	ПК, математическое ПО, WEBSTORM, таблицы
Лабораторное занятие №98. Решение систем линейных уравнений	Формирование умений применять прямые и итерационные методы решения СЛАУ	ПК, математические среды, WEBSTORM, визуализаторы матриц
Лабораторное занятие №99. Решение алгебраических и трансцендентных уравнений	Формирование умений находить корни уравнений методами деления отрезка, Ньютона и простой итерации	ПК, математическое ПО, WEBSTORM, инструменты построения графиков
Лабораторное занятие №100. Интерполирование функции	Формирование умений строить интерполяционные многочлены Лагранжа, Ньютона и сплайны	ПК, математические пакеты, WEBSTORM, системы визуализации
Лабораторное занятие №101. Численное интегрирование и дифференцирование	Формирование умений применять квадратурные формулы и методы численного дифференцирования	ПК, математическое ПО, WEBSTORM, графические библиотеки
Лабораторное занятие №102. Численное решение дифференциальных уравнений	Формирование умений решать ОДУ методами Эйлера и Рунге-Кутты, анализировать устойчивость	ПК, математические среды, WEBSTORM, инструменты визуализации фазовых портретов
МДК.02.06 Безопасность программного обеспечения		
Лабораторные занятия		
Лабораторное занятие №103. SQL инъекции - эксплуатация и защита уязвимого приложения	Формирование умений выявлять SQL-уязвимости, применять параметризованные запросы и ORM-защиту	ПК, уязвимое веб-приложение, SQLMap, WEBSTORM, СУБД
Лабораторное занятие №104. XSS атаки - создание и предотвращение межсайтового скриптинга	Формирование умений тестировать XSS-уязвимости, применять экранирование и CSP	ПК, браузеры, инструменты сканирования, WEBSTORM, веб-серверы
Лабораторное занятие №105. CSRF защита - реализация токенов и проверки Origin/Referer	Формирование умений внедрять CSRF-токены, настраивать валидацию заголовков и SameSite cookies	ПК, веб-фреймворки, WEBSTORM, браузеры, прокси-инструменты
Лабораторное занятие №106. Составление модели угроз для типового веб-приложения	Формирование умений применять STRWEBSTORM, выявлять векторы атак и документировать модель угроз	ПК, средства моделирования угроз, WEBSTORM, офисное ПО
Лабораторное занятие №107. Настройка безопасной аутентификации с JWT и refresh токенами	Формирование умений реализовывать безопасную выдачу, валидацию и ротацию JWT	ПК, серверная среда, WEBSTORM, инструменты работы с токенами
Лабораторное занятие №108. Шифрование	Формирование умений применять	ПК, криптографические библиотеки,

данных с использованием AES и RSA	симметричное и асимметричное шифрование, управлять ключами	WEBSTORM, терминал
Лабораторное занятие №109. Хэширование паролей с salt и adaptive functions (bcrypt, Argon2)	Формирование умений правильно хешировать пароли, настраивать сложность вычислений и соль	ПК, криптографические библиотеки, WEBSTORM, СУБД
Лабораторное занятие №110. Анализ сетевого трафика с помощью Wireshark	Формирование умений захватывать и анализировать пакеты, выявлять аномалии и уязвимости в трафике	ПК, Wireshark, сетевые интерфейсы, фильтры анализа
Лабораторное занятие №111. Настройка HTTPS и создание самоподписанных сертификатов	Формирование умений генерировать сертификаты, настраивать TLS и проверять цепочки доверия	ПК, OpenSSL, веб-серверы, WEBSTORM, браузеры
Лабораторное занятие №112. Защита от brute-force атак с ограничением попыток входа	Формирование умений внедрять rate limiting, блокировку IP и учет неудачных попыток	ПК, серверная среда, WEBSTORM, системы мониторинга
Лабораторное занятие №113. Безопасная работа с файлами	Формирование умений валидировать загрузку, проверять MIME-типы и защищать файловые пути	ПК, веб-фреймворки, WEBSTORM, сканеры безопасности
Лабораторное занятие №114. Реализация безопасной десериализации данных	Формирование умений безопасно парсить данные, предотвращать RCE и использовать безопасные форматы	ПК, WEBSTORM, библиотеки парсинга, тестовые среды
Лабораторное занятие №115. Настройка CORS политик для веб-приложений	Формирование умений настраивать политики доступа, ограничивать источники и методы запросов	ПК, веб-серверы, WEBSTORM, браузеры, инструменты тестирования API
Лабораторное занятие №116. Защита от DDOS атак с помощью rate limiting	Формирование умений настраивать правила ограничения запросов, использовать балансировщики и WAF	ПК, Nginx/HAProxy, системы мониторинга, WEBSTORM
Лабораторное занятие №117. Безопасная работа с памятью в приложениях	Формирование умений предотвращать переполнения буфера, управлять указателями и использовать безопасные типы	ПК, языки программирования (C/C++/Rust), отладчики, анализаторы памяти
Лабораторное занятие №118 Создание безопасного API с валидацией всех входных данных	Формирование умений разрабатывать безопасные REST API, выполнять серверную валидацию входных данных и тестировать устойчивость API к атакам	ПК, VS Code, Node.js/Python, Postman, SQLite/PostgreSQL, браузеры

Лабораторное занятие №119 Создание secure OAuth 2.0 провайдера с PKCE и защитой от атак	Формирование умений реализовывать OAuth 2.0 авторизацию, настраивать PKCE flow и обеспечивать защиту токенов	ПК, VS Code, Node.js/Python, OAuth 2.0 библиотеки, Postman, браузеры
Лабораторное занятие №120 Имплементация JWE (JSON Web Encryption) для защищённых токенов	Формирование умений реализовывать шифрование токенов JWE и обеспечивать безопасную передачу данных	ПК, VS Code, Node.js/Python, OpenSSL, JWT/JWE библиотеки
Лабораторное занятие №121 Разработка безопасного voting system с homomorphic encryption	Формирование умений разрабатывать безопасные системы электронного голосования и применять криптографические методы защиты данных	ПК, VS Code, Python/Node.js, криптографические библиотеки, SQLite/PostgreSQL
Лабораторное занятие №122 Реализация secure password manager с client-sWebStorm encryption	Формирование умений реализовывать безопасное хранение учетных данных и применять клиентское шифрование	ПК, VS Code, Node.js/Python, криптографические библиотеки, браузеры
Лабораторное занятие №123 Настройка HSM эмулятора для аппаратной защиты ключей	Формирование умений использовать HSM-эмуляторы для хранения и защиты криптографических ключей	ПК, SoftHSM, OpenSC, OpenSSL, VS Code
Лабораторное занятие №124 Разработка secure API gateway с JWT verification и rate limiting	Формирование умений настраивать API gateway, реализовывать JWT verification и ограничение запросов	ПК, Nginx/HAProxy/Kong Community Edition, Postman, VS Code
Лабораторное занятие №125 Создание hardware-backed key storage для мобильного приложения	Формирование умений реализовывать безопасное хранение криптографических ключей в мобильных приложениях	ПК, Android Studio, Android Keystore, Java/Kotlin SDK, эмулятор Android
Лабораторное занятие №126 Настройка certificate transparency logs для мониторинга SSL сертификатов	Формирование умений выполнять мониторинг SSL/TLS сертификатов и анализировать certificate transparency logs	ПК, OpenSSL, crt.sh, браузеры, VS Code
Лабораторное занятие №127 Разработка secure session management с защитой от hijacking	Формирование умений реализовывать безопасное управление пользовательскими сессиями и защиту от hijacking атак	ПК, VS Code, Node.js/Python/PHP, браузеры, Postman
Лабораторное занятие №128 Создание cryptographically secure RNG (random number generator)	Формирование умений реализовывать криптографически стойкие генераторы случайных чисел и анализировать качество генерации	ПК, VS Code, Python/Node.js, OpenSSL, инструменты тестирования randomness

3 УСЛОВИЯ РЕАЛИЗАЦИИ РАБОЧЕЙ ПРОГРАММЫ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

3.1 Материально-техническое обеспечение

Для реализации программы учебной дисциплины предусмотрены следующие специальные помещения:

Лаборатория «Информационной безопасности», оснащенная в соответствии с приложением 3 образовательной программы.

Лаборатория «Информатики и информационных технологий в профессиональной деятельности», оснащенная в соответствии с приложением 3 образовательной программы.

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

Помещение для воспитательной работы, оснащенное в соответствии с приложением 3 образовательной программы.

Компьютерный класс, оснащенный в соответствии с приложением 3 образовательной программы.

3.2 Учебно-методическое и информационное обеспечение реализации программы

Основные источники:

1. Емелина, Е. И. Поддержка и тестирование программных модулей : учебник / Е. И. Емелина. — Москва : КноРус, 2025. — 267 с. — ISBN 978-5-406-14483-1. — URL: <https://book.ru/book/957274> (дата обращения: 30.03.2026). — Текст : электронный.

2. Гагарина, Л. Г. Основы проектирования и разработки информационных систем : учебное пособие / Л.Г. Гагарина, Ю.С. Шевнина. — Москва : ИНФРА-М, 2026. — 211 с. — (Среднее профессиональное образование). - ISBN 978-5-16-020463-5. - Текст : электронный. - URL: <https://znanium.ru/catalog/product/2221110> (дата обращения: 30.03.2026). – Режим доступа: по подписке.

3. Истратова, Е. Е. Алгоритмы и структуры данных : учебное пособие / Е. Е. Истратова, И. Н. Томилов, А. А. Якименко. - Новосибирск : Изд-во НГТУ, 2025. - 68 с. - ISBN 978-5-7782-5502-9. - Текст : электронный. - URL: <https://znanium.ru/catalog/product/2244205> (дата обращения: 30.03.2026). – Режим доступа: по подписке.

4. Гниденко, И. Г. Технология разработки программного обеспечения : учебник для среднего профессионального образования / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 248 с. — (Профессиональное образование). — ISBN 978-5-534-18131-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/563151> (дата обращения: 30.03.2026).

5. Гателюк, О. В. Численные методы : учебник для среднего профессионального образования / О. В. Гателюк, Ш. К. Исмаилов, Н. В. Манюкова. — Москва : Издательство Юрайт, 2026. — 110 с. — (Профессиональное образование). — ISBN 978-5-534-07480-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/585190>.

6. Колдаев, В. Д. Численные методы и программирование : учебное пособие / В.Д. Колдаев ; под ред. Л.Г. Гагариной. — Москва : ФОРУМ : ИНФРА-М, 2026. — 336 с. — (Среднее профессиональное образование). - ISBN 978-5-8199-0779-5. - Текст : электронный. - URL: <https://znanium.ru/catalog/product/2219434>. – Режим доступа: по подписке.

7. Казарин, О. В. Программно-аппаратные средства защиты информации. Защита программного обеспечения : учебник и практикум для вузов / О. В. Казарин, А. С. Забабурин. — Москва : Издательство Юрайт, 2026. — 312 с. — (Высшее образование). — ISBN 978-5-9916-9043-0. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/584673>.

8. Заботина, Н. Н. Методы и средства проектирования информационных систем : учебное пособие / Н.Н. Заботина. — Москва : ИНФРА-М, 2023. — 331 с. + Доп. материалы [Электронный

ресурс]. — (Среднее профессиональное образование). - ISBN 978-5-16-015597-5. - Текст : электронный. - URL: <https://znanium.ru/catalog/product/1902833> . – Режим доступа: по подписке.

Дополнительные источники:

1. Казарин, О. В. Основы информационной безопасности: надежность и безопасность программного обеспечения : учебник для среднего профессионального образования / О. В. Казарин, И. Б. Шубинский. — 2-е изд. — Москва : Издательство Юрайт, 2026. — 352 с. — (Профессиональное образование). — ISBN 978-5-534-19384-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/587457> .

2. Зализняк, В. Е. Математическое моделирование : учебник для среднего профессионального образования / В. Е. Зализняк, О. А. Золотов. — Москва : Издательство Юрайт, 2026. — 125 с. — (Профессиональное образование). — ISBN 978-5-534-20526-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/587938> .

3. Сысолетин, Е. Г. Разработка интернет-приложений : учебник для вузов / Е. Г. Сысолетин, С. Д. Ростунцев ; под научной редакцией Л. Г. Доросинского. — Москва : Издательство Юрайт, 2025. — 80 с. — (Высшее образование). — ISBN 978-5-534-17124-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/562916> (дата обращения: 30.03.2026).

Интернет-ресурсы:

1. Интуит – национальный открытый университет. Основы программирования на AJAX [Электронный ресурс]. — Режим доступа: https://www.intuit.ru/studies/professional_retraining/16256/vWebStormo_courses/1017/info , свободный. – Загл. с экрана. Яз. рус.

2. Портал по php, MySQL и другим веб-технологиям [Электронный ресурс] - Режим доступа: <http://www.php.su>

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 результаты (ПК, ОК)	Наименование оценочного средства	Критерии оценки
МДК.02.01 Разработка программных модулей		
ПК 2.1 Проектировать модули программного обеспечения ПК 2.2 Разрабатывать модули программного обеспечения 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	Виды работ по практике Лабораторное занятие	См. ниже
МДК.02.02 Осуществление интеграции программных модулей		
ПК 2.3 Выполнять интеграцию модулей и компонентов программного обеспечения 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 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	Виды работ по практике Лабораторное занятие Курсовой проект	См. ниже
МДК.02.03 Поддержка и тестирование программных модулей		
ПК 2.4 Выполнять тестирование и отладку программного обеспечения 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	Лабораторное занятие	См. ниже
МДК.02.04 Математическое моделирование		
ПК 2.2 Разрабатывать модули программного обеспечения ПК 2.5 Осуществлять документирование программных модулей программного обеспечения ОК 01 Выбирать способы решения задач	Лабораторное занятие	См. ниже

профессиональной деятельности применительно к различным контекстам ОК 02 Использовать современные средства поиска, анализа и интерпретации информации, и информационные технологии для выполнения задач профессиональной деятельности		
МДК.02.05 Численные методы		
ПК 2.2 Разрабатывать модули программного обеспечения ПК 2.5 Осуществлять документирование программных модулей программного обеспечения 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	Лабораторное занятие	См. ниже
МДК.02.06 Безопасность программного обеспечения		
ПК 2.2 Разрабатывать модули программного обеспечения ПК 2.4 Выполнять тестирование и отладку программного обеспечения ПК 2.5 Осуществлять документирование программных модулей программного обеспечения 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	Лабораторное занятие	См. ниже
УП.02 Учебная практика		
ПК 2.1 Проектировать модули программного обеспечения ПК 2.2 Разрабатывать модули программного обеспечения ПК 2.3 Выполнять интеграцию модулей и компонентов программного обеспечения ПК 2.4 Выполнять тестирование и отладку программного обеспечения ПК 2.5 Осуществлять документирование программных модулей программного обеспечения 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	Отчёт по практике	См. ниже
ПП.02 Производственная практика		
ПК 2.1 Проектировать модули программного обеспечения	Отчёт по практике	См. ниже

ПК 2.2 Разрабатывать модули программного обеспечения ПК 2.3 Выполнять интеграцию модулей и компонентов программного обеспечения ПК 2.4 Выполнять тестирование и отладку программного обеспечения ПК 2.5 Осуществлять документирование программных модулей программного обеспечения 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		
--	--	--

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.02.01	Разработка программных модулей	Комплексный зачет с оценкой	6
МДК.02.02	Осуществление интеграции программных модулей	Экзамен	6
		Курсовой проект	6
МДК 02.03	Поддержка и тестирование программных модулей	Зачет с оценкой	8
МДК 02.04	Математическое моделирование	Комплексный зачет с оценкой	6
МДК 02.05	Численные методы	Экзамен	5
МДК 02.06	Безопасность программного обеспечения	Экзамен	7
УП.02	Наименование	Комплексный зачет	6
		Зачет	7
ПП.02	Наименование	Комплексный зачет	6
		Зачет	8

4.2.1 Оценочные средства для зачета, экзамена по МДК, практике

Оценочные средства для зачета с оценкой по МДК.02.01 Разработка программных модулей

Результаты обучения (ПК, ОК)	Оценочные средства для промежуточной аттестации
ПК 2.1, ПК 2.2, ОК 01, ОК 02	Теоретические вопросы по содержанию курса: 1. Что такое модульность в контексте веб-разработки?

	2. Перечислите основные преимущества и ограничения модульного подхода для SPA/SSR-приложений. 3. Объясните принцип единственной ответственности (SRP) и принцип открытости/закрытости (OCP) на примере веб-компонента или сервиса. 4. Что такое высокая связность (cohesion) и слабое зацепление (coupling)? Как эти принципы влияют на тестируемость и поддержку веб-модулей? 5. В чём разница между композицией и наследованием при построении веб-компонентов? Почему в современном фронтенде композиция считается предпочтительной? 6. Как JavaScript-интерфейсы и типы обеспечивают слабую связность между модулями? 7. В чём разница между структурной и номинальной типизацией? 8. Опишите архитектуру Event Loop в браузере и Node.js. В чём разница между микротасками (microtasks) и макротасками (macrotasks)? 9. Какие преимущества даёт async/await при обработке цепочек HTTP-запросов? 10. Как работает AbortController и AbortSignal? В каких сценариях веб-разработки их применение критически важно? 11. Для чего используются Web Workers и Worker Threads? 12. В чём отличие Map/Set от обычных объектов в JavaScript? 13. Какие инструменты и подходы обеспечивают безопасное обновление данных? 14. Назовите три поведенческих паттерна проектирования, наиболее востребованных в веб-разработке, и приведите пример их применения. 15. Какие существуют подходы к управлению состоянием в крупных веб-приложениях (локальный стейт, глобальный стор, серверный кэш)? 16. В чём преимущество реактивных паттернов? 17. Что такое сериализация и десериализация данных в контексте веб-API? 18. В чём ключевые различия между монолитной и микросервисной архитектурой? 19. Какие инженерные проблемы возникают при переходе на микросервисы? 20. Сравните синхронное (REST/gRPC) и асинхронное (брокеры сообщений) взаимодействие между сервисами. Когда какой подход предпочтительнее? 21. Что такое BFF (Backend for Frontend) и API Gateway? 22. Что такое CI/CD в контексте веб-проектов? 23. Для чего нужен линтер (например, ESLint)? 24. Что такое моки (mocks) при тестировании модулей? 25. В чём суть стратегии Git Flow? 26. Что такое Code Splitting и как он влияет на производительность? 27. Что такое JWT-токен? 28. Чем аутентификация отличается от авторизации? 29. Что такое XSS-уязвимость? 30. Почему серверная валидация обязательна даже при наличии клиентской?
ПК 2.1, ПК 2.2, ОК 01, ОК 02	Типовые практические задания Задание 1. Разработайте JavaScript-модуль для управления

	списком записей (например, список книг или задач). Для хранения данных вручную реализуйте структуру данных «Односвязный список» (Singly Linked List) с использованием классов (Node и List), не применяя стандартные массивы JS для основного хранения. Задание 2. Создайте веб-страницу с модулем асинхронного анализа текста. Напишите функцию, имитирующую загрузку данных: используйте объект Promise и setTimeout с задержкой в 2–3 секунды, после чего функция должна возвращать большую строку текста, содержащую даты (в формате ДД.ММ.ГГГГ) и email-адреса. Реализуйте простой паттерн проектирования «Наблюдатель» (Observer) или PubSub: создайте объект-менеджер, который генерирует событие завершения загрузки, а другой метод подписывается на это событие для начала обработки. После получения данных используйте регулярные выражения (RegExp) для извлечения всех email-адресов из текста и вывода их на страницу. Проект должен быть закоммитен в Git с сохранением истории изменений.
--	--

Оценочные средства для экзамена по МДК.02.02 Осуществление интеграции программных модулей

Результаты обучения (ПК, ОК)	Оценочные средства для промежуточной аттестации
ПК 2,3 ОК 01, ОК 02, ОК 05, ОК 09	Теоретические вопросы по содержанию курса: 1. Что такое REST API и его базовый архитектурный принцип? 2. Чем отличается метод PUT от PATCH? 3. Какие HTTP-статусы относятся к группе успешных ответов? 4. Зачем нужен заголовок Content-Type? 5. Что означает идемпотентность HTTP-метода? 6. В чём разница между клиентом и сервером в REST-архитектуре? 7. Как работает CORS и какую проблему интеграции он решает? 8. Что такое контракт API и почему он критичен при интеграции? 9. Чем webhook отличается от синхронного REST-запроса? 10. Для чего используются middleware в цепочке обработки запросов? 11. Что такое API Gateway и какую роль он играет в интеграции? 12. Зачем применять моки (mocks) при тестировании интеграции модулей? 13. Как обеспечить обратную совместимость версий API? 14. Чем аутентификация отличается от авторизации? 15. Как работает JWT и где безопасно хранить его на клиенте? 16. В чём суть протокола OAuth 2.0? 17. Что такое CSRF-атака и как от неё защититься? 18. Почему HTTPS обязателен при передаче токенов и персональных данных? 19. Что такое XSS и как санитизация входящих данных защищает интеграцию? 20. Зачем нужен Rate Limiting в публичном API? 21. Почему серверная валидация обязательна даже при

	наличии клиентской проверки? 22. Что такое сериализация/десериализация в контексте обмена между модулями? 23. Какие заголовки управляют кэшированием HTTP-ответов? 24. Как правильно форматировать ошибки API, не раскрывая внутреннюю архитектуру? 25. Что такое WebStormmpotency-Key и в каких сценариях интеграции он применяется?
ПК 2,3 ОК 01, ОК 02, ОК 05, ОК 09	Типовые практические задания Задание 1. Разработайте веб-модуль на чистом JavaScript, который выполняет авторизацию через REST API и загружает защищённые данные. Реализуйте форму входа, отправляющую POST-запрос с учётными данными. При успешном ответе сохраните JWT-токен в памяти приложения и используйте его в заголовке Authorization для последующих GET-запросов. Обработайте состояния интерфейса: загрузка, успешный вывод данных, ошибка 401 (неавторизован), ошибка 400/500 (некорректные данные или сбой сервера). Добавьте защиту от повторной отправки формы и базовую клиентскую валидацию полей перед отправкой. Задание 2. Создайте минимальный серверный модуль с одним REST-маршрутом, принимающим JSON. Реализуйте middleware для проверки JWT-токена, валидацию тела запроса (проверка обязательных полей, типов данных и допустимой длины строк) и возврат строгих HTTP-статусов (200, 201, 400, 401, 403, 500). Настройте заголовки безопасности: ограничение разрешённых методов, запрет кэширования чувствительных ответов, базовая защита от MIME-сниффинга и XSS.

Оценочные средства для зачета с оценкой по МДК.02.03 Поддержка и тестирование программных модулей

Результаты обучения (ПК, ОК)	Оценочные средства для промежуточной аттестации
ПК 2.4, ОК 01, ОК 02	Теоретические вопросы по содержанию курса: 1. Что такое модульное (unit) тестирование? 2. Чем интеграционное тестирование отличается от модульного? 3. Что проверяют end-to-end (E2E) тесты? 4. В чём разница между тестированием «белого» и «чёрного» ящика? 5. Что такое регрессионное тестирование и когда его проводят? 6. Зачем нужны smoke-тесты и sanity-проверки? 7. Что такое тестирование производительности и какие метрики оно измеряет? 8. Назовите семь базовых принципов тестирования ПО. 9. В чём разница между ошибкой (error), дефектом (defect) и сбоем (failure)? 10. Что такое тестовое покрытие (code coverage) и какие его основные виды существуют? 11. Что такое приёмочное тестирование (UAT) и кто является его заказчиком? 12. В чём разница между функциональным и нефункциональным тестированием модуля? 13. Объясните принцип эквивалентного разбиения на классы. 14. Что такое граничные значения и почему их критически важно проверять?

	15. В чём суть попарного тестирования (pairwise testing)? 16. Что такое тестирование на основе состояний (state transition testing)? 17. Как применяется диаграмма причинно-следственных связей при проектировании тестов? 18. Какие обязательные поля должны содержать отчёт о баге (bug report)? 19. Из каких основных частей состоит тест-кейс? 20. Чем тест-план отличается от тестовой стратегии? 21. Что такое чек-лист тестирования и когда его достаточно использовать? 22. Как правильно описывать шаги воспроизведения (steps to reproduce)? 23. Какие критерии определяют целесообразность автоматизации тестирования модуля? 24. Что такое тестовые двойники (mocks, stubs, spies) и зачем они нужны? 25. Как обеспечить изоляцию тестов при работе с сетевыми запросами или БД? 26. Что такое snapshot-тестирование и в каких UI-сценариях оно применяется? 27. Как интеграционные тесты помогают выявлять нарушения контрактов между модулями?
ПК 2.4, ОК 01, ОК 02	Типовые практические задания Задание 1. Проведите проверку серверного модуля фильтрации списка товаров. Вам предоставлены три параметра запроса: категория (строка), цена (число, от 0 до 100000), сортировка (asc/desc). Модуль возвращает JSON-массив. Составьте чек-лист из 12 проверок, покрывающий корректные комбинации параметров, граничные значения цены, невалидные типы данных, отсутствие результатов и обработку ошибок сервера. Выполните проверки через Postman или curl, отметьте Passed/Failed в чек-листе. Напишите один автоматизированный тест на JavaScript с использованием Jest, который отправляет валидный запрос, проверяет статус 200 и гарантирует, что возвращённые цены не превышают указанный лимит. Задание 2. Протестируйте изолированный JavaScript-модуль, который рассчитывает итоговую стоимость доставки на основе трёх входных параметров: индекс населённого пункта (строка из 6 цифр), вес груза (число от 0.5 до 50 кг) и тип доставки («стандарт» или «экспресс»). Модуль возвращает объект с полями price (число), estimatedDays (число) и status («success» или «error» с текстом причины). Составьте чек-лист из 12 проверок, покрывающий: корректные комбинации параметров, граничные значения веса (0.5, 50, 50.1), невалидные форматы индекса, недоступность экспресс-доставки для грузов свыше 30 кг, обработку пустых полей и выброс исключений при некорректных типах данных. Выполните проверки, вызывая функцию модуля напрямую в консоли браузера или через минимальный HTML-интерфейс, фиксируя результат (Passed/Failed) для каждого пункта. Напишите один автоматизированный тест на Jest, который передаёт в функцию валидные данные и проверяет точное совпадение возвращаемой стоимости и количества дней доставки с ожидаемыми значениями.

Оценочные средства для зачета с оценкой по МДК.02.04 Математическое моделирование

Результаты обучения (ПК, ОК)	Оценочные средства для промежуточной аттестации
ПК 2.2, ПК 2.5, ОК 1, ОК 2	1. Теоретические вопросы по содержанию курса: 2. Понятие решения. Множество решений, оптимальное решение. Показатель эффективности решения 3. Математические модели, принципы их построения, виды моделей. 4. Задачи: классификация, методы решения, граничные условия. 5. Общий вид и основная задача линейного программирования. Симплекс – метод. 6. Транспортная задача. Методы нахождения начального решения транспортной задачи. Метод потенциалов. 7. Общий вид задач нелинейного программирования. Графический метод решения задач нелинейного программирования. Метод множителей Лагранжа. 8. Основные понятия динамического программирования: шаговое управление, управление операциями в целом, оптимальное управление, выигрыш на данном шаге, выигрыш за всю операцию, аддитивный критерий, мультипликативный критерий. 9. Простейшие задачи, решаемые методом динамического программирования. 10. Методы хранения графов в памяти ЭВМ. Задача о нахождении кратчайших путей в графе и методы ее решения. 11. Задача о максимальном потоке и алгоритм Форда Фалкерсона. 12. Системы массового обслуживания: понятия, примеры, модели. 13. Основные понятия теории марковских процессов: случайный процесс, марковский процесс, граф состояний, поток событий, вероятность состояния, уравнения Колмогорова, финальные вероятности состояний. 14. Схема гибели и размножения. 15. Метод имитационного моделирования. Единичный жребий и формы его организации. Примеры задач 16. Понятие прогноза. Количественные методы прогнозирования: скользящие средние, экспоненциальное сглаживание, проектирование тренда. Качественные методы прогноза 17. Предмет и задачи теории игр. Основные понятия теории игр: игра, игроки, партия, выигрыш, проигрыш, ход, личные и случайные ходы, стратегические игры, стратегия, оптимальная стратегия. 18. Антагонистические матричные игры: чистые и смешанные стратегии. 19. Методы решения конечных игр: сведение игры $m \times n$ к задаче линейного программирования, численный метод – метод итераций. 20. Область применимости теории принятия решений. Принятие решений в условиях определенности, в условиях риска, в условиях неопределенности. 21. Критерии принятия решений в условиях неопределенности.

	Дерево решений.
ПК 2.2, ПК 2.5, ОК 1, ОК 2	Типовые практические задания Задание 1. Выполнить экономико-математическое моделирование процесса в соответствии со следующими этапами: 1. Определение и постановка экономической проблемы, ее рассмотрение и тщательный анализ 2. Построение математической модели; 3. Математический анализ построенной модели; 4. Подготовка исходной информационной базы; 5. Решение поставленной задачи в числах; 6. Анализ, обработка и рассмотрение возможных вариантов применения полученных числовых результатов. Пример. Фермер на своем участке выращивает огурцы и помидоры. Чтобы не потерять урожай, он использует азотные, калийные, фосфатные удобрения и навоз. Чтобы удобрить один гектар огурцов ему необходимо 20 ед. азотных удобрений, 40 – калийных, 10 – фосфатных, 20 – навоза; помидоров – 10 – калийных, 15 – фосфатных, 10 – навоза. Его запасы удобрений следующие: азотных – 120, калийных – 320, фосфатных – 160, навоза – 180. Прибыль с 1 га площади, засаженной огурцами, – 5000 у. д. е., а помидорами – 3000 у. д. е. Сколько гектаров огурцов и помидоров необходимо обработать для получения максимальной прибыли.

Оценочные средства для экзамена по МДК.02.05 Численные методы

Результаты обучения (ПК, ОК)	Оценочные средства для промежуточной аттестации
ПК 2.2, ПК 2.5 ОК 1, ОК 2	Теоретические вопросы по содержанию курса: 1. Что такое абсолютная и относительная погрешность вычислений? 2. В чём разница между ошибкой округления и ошибкой усечения? 3. Что означает устойчивость численного алгоритма? 4. Как определяется порядок сходимости итерационного метода? 5. Что такое число обусловленности матрицы и почему оно важно? 6. В чём суть метода половинного деления (бисекции)? 7. Как работает метод Ньютона и какова его скорость сходимости? 8. Чем метод секущих отличается от метода Ньютона? 9. Что такое условие остановки итерационного процесса? 10. Почему выбор начального приближения критичен для метода Ньютона? 11. В чём идея метода Гаусса исключения неизвестных? 12. Что такое LU-разложение и когда его выгодно применять? 13. Чем прямые методы решения СЛАУ отличаются от итерационных? 14. Как работает метод простой итерации для решения СЛАУ? 15. Что такое диагональное преобладание и зачем оно нужно?

	16. На какой идее основан метод прямоугольников для вычисления интеграла? 17. Чем метод трапеций отличается от метода Симпсона по точности? 18. Почему численное дифференцирование считается неустойчивой операцией? 19. Что такое формула конечных разностей для приближения производной? 20. Как выбрать шаг интегрирования для баланса между точностью и скоростью?
ПК 2.2, ПК 2.5 ОК 1, ОК 2	Типовые практические задания Задание 1. Дано уравнение: $f(x) = x^2 - 2 = 0$. Известно, что корень находится на отрезке $[1; 2]$. Требуется найти приближённое значение корня методом половинного деления (бисекции) с точностью $\varepsilon = 0.1$ (по длине отрезка). Задание 2. Требуется вычислить приближённое значение интеграла $\int_0^1 x^2 dx$, используя метод трапеций с количеством разбиений $n = 4$.

Оценочные средства для экзамена по МДК.02.06 Безопасность программного обеспечения

Результаты обучения (ПК, ОК)	Оценочные средства для промежуточной аттестации
ПК 2.2, ПК 2.4, ПК 2.5, ОК 1, ОК 2	Теоретические вопросы по содержанию курса: 1. Что такое триада информационной безопасности (CIA)? 2. Чем отличается уязвимость (vulnerability) от угрозы (threat) и атаки (attack)? 3. Что такое принцип наименьших привилегий и зачем он нужен? 4. В чём разница между аутентификацией, авторизацией и аудитом? 5. Что такое модель угроз и когда её необходимо обновлять? 6. то такое инъекции (SQL, NoSQL, Command) и как от них защититься? 7. В чём суть уязвимости XSS и какие её основные типы существуют? 8. Что такое CSRF-атака и как работает токен-защита? 9. Чем опасна небезопасная десериализация данных? 10. Что такое атака «человек посередине» (MitM) и как HTTPS ей противодействует? 11. В чём разница между симметричным и асимметричным шифрованием? 12. Зачем нужны хеш-функции и чем они отличаются от шифрования? 13. Что такое соль (salt) и зачем её добавляют при хешировании паролей? 14. Как работает цифровая подпись и какие задачи она решает? 15. Почему нельзя хранить пароли в открытом виде и какие алгоритмы для этого безопасны? 16. Что такое безопасный жизненный цикл разработки (Secure SDLC)? 17. В чём суть код-ревью с фокусом на безопасность и на что обращать внимание? 18. Что такое статический (SAST) и динамический (DAST)

	анализ безопасности? 19. Зачем нужен список зависимостей (SBOM) и как он помогает при уязвимостях в сторонних библиотеках? 20. Что такое план реагирования на инциденты (IncWebStormnt Response) и какие этапы он включает?
ПК 2.2, ПК 2.4, ПК 2.5, ОК 1, ОК 2	Типовые практические задания Задание 1. Дано описание веб-модуля: форма входа принимает логин и пароль, отправляет их по HTTPS на сервер, сервер проверяет учётные данные в базе данных, при успехе выдаёт JWT-токен, который клиент сохраняет в localStorage и использует в заголовке Authorization для последующих запросов. Требуется провести анализ безопасности модуля и заполнить таблицу угроз. Задание 2. Дано описание эндпоинта: POST /api/users принимает JSON { "email": "string", "password": "string" }, создаёт пользователя, возвращает { "id": number, "email": string } со статусом 201. При ошибке валидации возвращает 400 с текстом «Invalid email», при дубликате — 409 с текстом «Email already exists». Требуется провести аудит безопасности по чек-листу.

Оценочные средства для зачета по Учебной практике

Результаты обучения (ПК, ОК)	Оценочные средства для промежуточной аттестации
ПК 2.1 – ПК 2.5 ОК 01, ОК 02	Виды работ и задание на учебную практику

Оценочные средства для зачета по Производственной практике

Результаты обучения (ПК, ОК)	Оценочные средства для промежуточной аттестации
ПК 2.1 – ПК 2.5 ОК 01, ОК 02	Виды работ и задание на производственную практику

Критерии зачета с оценкой /комплексного 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.

Критерии оценки курсового проекта (работы)

Код и наименование компетенций	Перечень оцениваемых умений, навыков (практического опыта)	Максимальный балл
ОК 01: Выбирать способы решения задач	Умение: выявлять и эффективно искать информацию, необходимую для решения	2

профессиональной деятельности применительно к различным контекстам	задачи и/или проблемы;	
ОК 02: Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности	Умение: использовать современное программное обеспечение, различные цифровые средства для решения профессиональных задач; проявлять культуру информационной безопасности при использовании информационно-коммуникационных технологий;	2
ОК 05: Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста	Умение: грамотно излагать свои мысли и оформлять документы по профессиональной тематике на государственном языке;	2
ОК 09: Пользоваться профессиональной документацией на государственном и иностранном языках	Умение: читать, понимать и находить необходимые технические данные и инструкции в руководствах в любом доступном формате;	2
ПК 2.1 Проектировать модули программного обеспечения	Умение: проектировать архитектуру модуля, включая выбор подходящих паттернов проектирования и структуры данных	2
ПК 2.1 Проектировать модули программного обеспечения	Умение: проектировать интерфейсы программного обеспечения для взаимодействия с другими модулями и системами	2
ПК 2.2 Разрабатывать модули программного обеспечения	Умение: разрабатывать модули программного обеспечения с использованием различных языков программирования и технологий	2
ПК 2.2 Разрабатывать модули программного обеспечения	Умение: работать с системой контроля версий	2
ПК 2.3 Выполнять интеграцию модулей и компонентов программного обеспечения	Умение: работать с API и устанавливать соединения между компонентами	2
ПК 2.3 Выполнять интеграцию модулей и компонентов программного обеспечения	Умение: интегрировать модули и компоненты, обеспечивая их взаимодействие	2
ПК 2.4 Выполнять тестирование и отладку программного обеспечения	Умение: выполнять тестирование программного обеспечения вручную и автоматизировать процесс тестирования	2
ПК 2.5 Осуществлять документирование программных модулей программного обеспечения	Умение: создавать диаграммы для иллюстрации работы модулей	2
ПК 2.5 Осуществлять документирование программных модулей программного обеспечения	Умение: программировать с использованием комментариев для документирования	1
ИТОГО		25,00

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			
Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам	выявлять и эффективно искать информацию, необходимую для решения задачи и/или проблемы	выделяет ключевые запросы и целенаправленно проводит поиск релевантной информации	0 - определяет отдельные запросы, но поиск приводит к незначительным результатам; 1 – не всегда находит нужную или эффективную информацию; 2 – определяет необходимые ключевые запросы, последовательно осуществляет эффективный поиск нужной информации, обеспечивает полноту и точность найденных сведений;	2	1	2
Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности	использовать современное программное обеспечение, различные цифровые средства для решения профессиональных задач; проявлять культуру информационной безопасности при использовании информационно-коммуникационных технологий	Обосновывает выбор, использует актуальные версии, настраивает рабочее окружение под проект	0 – выбирает инструменты без обоснования, использует устаревшие или непроверенные версии, настройка рабочего окружения отсутствует или выполнена с ошибками, что затрудняет выполнение задач; 1 – иногда обосновывает выбор программного обеспечения, использует преимущественно актуальные версии, настраивает окружение частично, но требует корректировок или внешних подсказок для стабильной работы; 2 – аргументированно подбирает цифровые средства под конкретные задачи проекта, последовательно применяет стабильные и актуальные версии, самостоятельно и полно настраивает рабочее окружение,	2	1	2

			обеспечивая эффективность, воспроизводимость и безопасность рабочего процесса;			
Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста	грамотно излагать свои мысли и оформлять документы по профессиональной тематике на государственном языке;	оформляет курсовой проект согласно требованиям	0 - не соблюдает правила оформления; 1 - частично соблюдает требования к оформлению; 2 - полностью соблюдает требования к оформлению;	2	1	2
Пользоваться профессиональной документацией на государственном и иностранном языках	читать, понимать и находить необходимые технические данные и инструкции в руководствах в любом доступном формате;	извлекает необходимую техническую информацию из руководств любого формата для выполнения курсового проекта	0 - не способен извлечь нужную информацию из руководства, допускает грубые ошибки при интерпретации информации; 1 - способен частично извлекать требуемую информацию, однако допускает незначительные ошибки или пропускает важные детали; 2 - свободно и точно извлекает всю необходимую техническую информацию из любых форматов руководств, уверенно применяя её в рамках курсового проекта;	2	1	2
Проектировать модули программного обеспечения	проектировать архитектуру модуля, включая	выбирает архитектурный паттерн и структуры	0 – не может выбрать паттерн или структуру данных, выбор произволен и не обоснован;	2	1	2

	выбор подходящих паттернов проектирования и структуры данных	данных, обосновывает выбор применительно к задаче	1 – выбирает паттерн или структуру данных с подсказками, обоснование частично или поверхностно; 2 – самостоятельно и аргументированно выбирает архитектурный паттерн и структуры данных, чётко соотносит их с требованиями задачи и ограничениями проекта;			
Проектировать модули программного обеспечения	проектировать интерфейсы программного обеспечения для взаимодействия с другими модулями и системами	формулирует контракты взаимодействия (методы, параметры, форматы данных), описывает сценарии обмена	0 – не формулирует контракты или формулирует с грубыми ошибками, сценарии обмена не описаны; 1 – формулирует контракты частично, с неточностями, сценарии описаны схематично; 2 – чётко и полно формулирует контракты взаимодействия, детально описывает сценарии обмена данными, учитывает обработку ошибок и пограничные случаи;	2	1	2
Разрабатывать модули программного обеспечения	разрабатывать модули программного обеспечения с использованием различных языков программирования и технологий	пишет код согласно техническому заданию, применяет синтаксис и возможности языка, соблюдает стандарты кодирования	0 – код не соответствует заданию, содержит синтаксические ошибки, не читается; 1 – код частично решает задачу, есть ошибки стиля или логики, требует доработки; 2 – код полностью реализует требуемую функциональность, соответствует стандартам качества, читаем и поддерживаем;	2	1	2
Разрабатывать модули программного обеспечения	работать с системой контроля версий	создаёт репозиторий, фиксирует изменения осмысленными коммитами, работает с ветками и слияниями	0 – не использует СКВ или использует хаотично, коммиты бессмысленны, ветки не применяются; 1 – фиксирует изменения, но сообщения коммитов формальны, работа с ветками требует	2	1	2

			подсказок; 2 – последовательно ведёт историю изменений, использует feature-branch workflow, осмысленно описывает коммиты, корректно выполняет слияния и разрешение конфликтов;			
Выполнять интеграцию модулей и компонентов программного обеспечения	работать с API и устанавливать соединения между компонентами	настраивает подключение к внешнему API, формирует корректные запросы, обрабатывает ответы и ошибки	0 – не может установить соединение, запросы некорректны, ошибки не обрабатываются; 1 – устанавливает соединение с подсказками, запросы работают частично, обработка ошибок базовая; 2 – самостоятельно настраивает подключение, формирует валидные запросы с необходимыми заголовками и параметрами, реализует полноценную обработку ответов и сценариев ошибок;	2	1	2
Выполнять интеграцию модулей и компонентов программного обеспечения	интегрировать модули и компоненты, обеспечивая их взаимодействие	связывает модули через определённые интерфейсы, обеспечивает передачу данных, проверяет совместную работу	0 – модули не взаимодействуют или взаимодействуют с критическими ошибками, данные теряются; 1 – взаимодействие настроено частично, есть сбои при передаче данных, требуется отладка; 2 – модули интегрированы корректно, данные передаются полностью и в нужном формате, совместная работа стабильна, проведены тесты интеграции;	2	1	2
Выполнять тестирование и отладку программного обеспечения	выполнять тестирование программного обеспечения вручную и автоматизировать процесс	составляет тест-кейсы, выполняет ручные проверки, пишет автотесты для ключевых сценариев	0 – не составляет тест-кейсы, тестирование хаотично, автотесты отсутствуют; 1 – составляет простые тест-кейсы, выполняет базовые проверки, автотесты пишутся с подсказками;	2	1	2

	тестирования		2 – самостоятельно разрабатывает полные тест-кейсы, системно проводит ручное тестирование, пишет работающие автотесты, покрывающие основные и пограничные сценарии;			
Осуществлять документирование программных модулей программного обеспечения	создавать диаграммы для иллюстрации работы модулей	строит диаграммы (UML, блок-схемы, sequence), отражающие архитектуру, потоки данных или последовательность вызовов	0 – не может построить диаграмму или строит с грубыми ошибками, нотация не соблюдена; 1 – строит диаграммы частично, есть неточности в обозначениях или логике, требует корректировок; 2 – самостоятельно и точно строит диаграммы в принятой нотации, диаграммы полно и наглядно отражают проектируемые решения, понятны команде;	2	1	2
Осуществлять документирование программных модулей программного обеспечения	программировать с использованием комментариев для документирования	добавляет комментарии к коду, оформляет JSDoc/DocString для функций и классов	0 – комментарии отсутствуют или формальны («магия»), не помогают понять код; 1 – комментарии есть, но местами избыточны или недостаточны, оформление документации частично; 2 – код снабжён лаконичными и содержательными комментариями, публичные функции и классы оформлены согласно стандарту документации, что обеспечивает лёгкость чтения и поддержки;	2	0,5	1

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: выявлять и эффективно искать информацию, необходимую для решения задачи и/или проблемы;	2
ОК 02: Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности	Умение: использовать различные цифровые средства для решения профессиональных задач;	2
ПК 2.1 Проектировать модули программного обеспечения	Умение: проектировать архитектуру модуля, включая выбор подходящих паттернов проектирования и структуры данных	2
ПК 2.1 Проектировать модули программного обеспечения	Навык: проектирования модулей ПО с учетом требований заказчика	2
ПК 2.2 Разрабатывать модули программного обеспечения	Навык: создания модулей программного обеспечения на различных языках программирования	2
ПК 2.2 Разрабатывать модули программного обеспечения	Умение: работать с системой контроля версий	2
ПК 2.3 Выполнять интеграцию модулей и компонентов программного обеспечения	Умение: работать с API и устанавливать соединения между компонентами	2
ПК 2.3 Выполнять интеграцию модулей и компонентов программного обеспечения	Умение: интегрировать модули и компоненты, обеспечивая их взаимодействие	2
ПК 2.3 Выполнять интеграцию модулей и компонентов программного обеспечения	Навык: работы с API и веб-сервисами для взаимодействия между модулями	2
ПК 2.4 Выполнять тестирование и отладку программного обеспечения	Умение: выполнять тестирование программного обеспечения вручную и автоматизировать процесс тестирования	2
ПК 2.4 Выполнять тестирование и отладку программного обеспечения	Навык: отладки программного обеспечения на уровне программных модулей	2
ПК 2.5 Осуществлять документирование программных модулей программного обеспечения	Навык: документирования кода, API и интерфейсов	2
ПК 2.5 Осуществлять документирование программных модулей программного обеспечения	Умение: программировать с использованием комментариев для документирования	1
ИТОГО		25,00

Задание 1.

Разработка и интеграция изолированного программного модуля «Сервис управления задачами (Task Manager)» с клиент-серверным взаимодействием, подключением к внешнему API и покрытием тестами.

Функциональные требования:

Необходимо реализовать серверный модуль с REST API, включающий следующие эндпоинты:

POST /tasks – создание задачи.

Входные данные (JSON): title (строка, 3–50 символов), status ("todo" | "in_progress" | "done"), userId (число).

Возвращает: созданный объект задачи со статусом 201 Created.

GET /tasks – получение списка задач.

Поддержка фильтрации по статусу через query-параметр ?status=....

Возвращает: массив задач со статусом 200 OK.

GET /tasks/:id – получение задачи по идентификатору.

Дополнительно: выполнить запрос к внешнему публичному API (например, <https://jsonplaceholder.typicode.com/users/{userId}>), получить имя пользователя и добавить его в ответ.

Возвращает: объединённый объект со статусом 200 OK. При отсутствии задачи – 404 Not Found.

Требования к качеству и процессу разработки:

Архитектура: применить шаблон проектирования (Controller-Service-Repository, MVC или аналог). Чёткое разделение слоёв: маршрутизация, бизнес-логика, работа с данными, обработка ошибок.

Валидация и безопасность: строгая проверка входящих данных на сервере. Возврат структурированных JSON-ошибок. Запрет вывода внутренних stack trace в production-ответах. Отсутствие хардкода чувствительных данных.

Система контроля версий: работа в Git с feature-branch workflow. Минимум 5 осмысленных коммитов с понятными сообщениями. Корректное разрешение конфликтов (если применимо).

Инструментарий: использовать актуальные стабильные версии фреймворков/библиотек. Настроить линтер/форматтер кода и зафиксировать конфигурацию в репозитории.

Тестирование: написать минимум 2 unit-теста (проверка валидации или бизнес-логики) и 1 интеграционный тест (проверка работы эндпоинта). Тесты должны запускаться одной командой.

Требования к документации:

Создать файл README.md с инструкцией по развёртыванию, запуску и тестированию модуля.

Описать контракты API: метод, путь, параметры, примеры успешных и ошибочных ответов.

Включить схему взаимодействия модулей (UML-диаграмма компонентов, Sequence-диаграмма или блок-схема в формате Mermaid/PNG).

Документировать публичные функции/классы комментариями в стандарте JSDoc/DocString/аналог.

ПЕРЕЧЕНЬ ПРЕДОСТАВЛЯЕМЫХ РЕЗУЛЬТАТОВ

– Ссылка на публичный Git-репозиторий (или архив проекта с сохранённой папкой .git и рабочей конфигурацией запуска).

– Файл README.md, полностью соответствующий требованиям п.4.

– Краткий отчёт (1 страница, PDF или Markdown) с обоснованием выбора инструментов, указанием версий зависимостей и описанием принятых мер информационной безопасности.

–Демонстрация работы модуля в режиме реального времени (очный формат) или видеозапись экрана с пошаговым выполнением запросов через Postman/cURL/браузер (дистанционный формат).

Критерии оценивания:

Оцениваемая ПК, ОК	Критерий оценивания (умения, навыки/ практический опыт)	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 (операции) или набор действий для оценки подкритерия	Описание результата выполнения конкретного действия (операции) подкритерия в баллах			
Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам	выявлять и эффективно искать информацию, необходимую для решения задачи и/или проблемы;	выявлять и эффективно искать информацию, необходимую для решения задачи и/или проблемы	0 - определяет отдельные запросы, но поиск приводит к незначительным результатам; 1 – не всегда находит нужную или эффективную информацию; 2 – определяет необходимые ключевые запросы, последовательно осуществляет эффективный поиск нужной информации, обеспечивает полноту и точность найденных сведений;	2	1	2
Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности	использовать различные цифровые средства для решения профессиональных задач;	использовать современное программное обеспечение, различные цифровые средства для решения профессиональных задач; проявлять культуру информационной безопасности при использовании информационно-коммуникационных технологий	0 – выбирает инструменты без обоснования, использует устаревшие или непроверенные версии, настройка рабочего окружения отсутствует или выполнена с ошибками, что затрудняет выполнение задач; 1 – иногда обосновывает выбор программного обеспечения, использует преимущественно актуальные версии, настраивает окружение частично, но требует корректировок или внешних подсказок для стабильной работы; 2 – аргументированно подбирает цифровые средства под конкретные задачи проекта, последовательно применяет стабильные и актуальные версии, самостоятельно и полно настраивает рабочее окружение,	2	1	2

			обеспечивая эффективность, воспроизводимость и безопасность рабочего процесса;			
Проектировать модули программного обеспечения	проектировать архитектуру модуля, включая выбор подходящих паттернов проектирования и структуры данных	выбирает архитектурный паттерн и структуры данных, обосновывает выбор применительно к задаче	0 – не может выбрать паттерн или структуру данных, выбор произволен и не обоснован; 1 – выбирает паттерн или структуру данных с подсказками, обоснование частично или поверхностно; 2 – самостоятельно и аргументированно выбирает архитектурный паттерн и структуры данных, чётко соотносит их с требованиями задачи и ограничениями проекта;	2	1	2
Проектировать модули программного обеспечения	проектирования модулей ПО с учетом требований заказчика	анализирует требования заказчика, декомпозирует задачу на логические модули, формирует техническое задание и схему архитектуры решения	0 – не анализирует требования или допускает критические ошибки в декомпозиции, архитектура не соответствует задаче; 1 – частично анализирует требования, декомпозиция требует корректировок, архитектура описана схематично; 2 – полно и точно анализирует требования, грамотно декомпозирует задачу на модули, формирует обоснованную архитектуру, соответствующую ограничениям и целям проекта;	2	1	2
Разрабатывать модули программного обеспечения	создания модулей программного обеспечения на различных языках программирования	пишет код модуля на выбранном языке программирования, применяет синтаксические конструкции и парадигмы языка, соблюдает стандарты оформления	0 – код не компилируется/не выполняется, содержит грубые синтаксические и логические ошибки, не соответствует заданию; 1 – код частично реализует функционал, присутствуют ошибки стиля или логики, требует доработки и пояснений; 2 – код полностью	2	1	2

			реализует поставленную задачу, соответствует стандартам кодирования, оптимизирован, читаем и готов к интеграции;			
Разрабатывать модули программного обеспечения	работать с системой контроля версий	создаёт репозиторий, фиксирует изменения осмысленными коммитами, работает с ветками и слияниями	0 – не использует СКВ или использует хаотично, коммиты бессмысленны, ветки не применяются; 1 – фиксирует изменения, но сообщения коммитов формальны, работа с ветками требует подсказок; 2 – последовательно ведёт историю изменений, использует feature-branch workflow, осмысленно описывает коммиты, корректно выполняет слияния и разрешение конфликтов;	2	1	2
Выполнять интеграцию модулей и компонентов программного обеспечения	работать с API и устанавливать соединения между компонентами	настраивает подключение к внешнему API, формирует корректные запросы, обрабатывает ответы и ошибки	0 – не может установить соединение, запросы некорректны, ошибки не обрабатываются; 1 – устанавливает соединение с подсказками, запросы работают частично, обработка ошибок базовая; 2 – самостоятельно настраивает подключение, формирует валидные запросы с необходимыми заголовками и параметрами, реализует полноценную обработку ответов и сценариев ошибок;	2	1	2
Выполнять интеграцию модулей и компонентов программного обеспечения	интегрировать модули и компоненты, обеспечивая их взаимодействие	связывает модули через определённые интерфейсы, обеспечивает передачу данных, проверяет совместную работу	0 – модули не взаимодействуют или взаимодействуют с критическими ошибками, данные теряются; 1 – взаимодействие настроено частично, есть сбои при передаче данных, требуется отладка; 2 – модули интегрированы	2	1	2

			корректно, данные передаются полностью и в нужном формате, совместная работа стабильна, проведены тесты интеграции;			
Выполнять интеграцию модулей и компонентов программного обеспечения	работы с API и веб-сервисами для взаимодействия между модулями	формирует HTTP/HTTPS-запросы, парсит ответы сервера (JSON/XML), обрабатывает сетевые ошибки и таймауты	0 – не может сформировать запрос или обработать ответ, данные не парсятся, ошибки сети игнорируются; 1 – запросы формируются с подсказками, обработка ответов базовая, не все ошибки обрабатываются корректно; 2 – самостоятельно формирует валидные запросы с заголовками и параметрами, корректно парсит ответы, реализует устойчивую обработку сетевых сбоев и таймаутов;	2	1	2
Выполнять тестирование и отладку программного обеспечения	выполнять тестирование программного обеспечения вручную и автоматизировать процесс тестирования	составляет тест-кейсы, выполняет ручные проверки, пишет автотесты для ключевых сценариев	0 – не составляет тест-кейсы, тестирование хаотично, автотесты отсутствуют; 1 – составляет простые тест-кейсы, выполняет базовые проверки, автотесты пишутся с подсказками; 2 – самостоятельно разрабатывает полные тест-кейсы, системно проводит ручное тестирование, пишет работающие автотесты, покрывающие основные и пограничные сценарии;	2	1	2
Выполнять тестирование и отладку программного обеспечения	отладки программного обеспечения на уровне программных модулей	использует встроенный отладчик и анализаторы логов, локализует источники сбоев, устраняет ошибки в коде модуля	0 – не использует инструменты отладки, локализация ошибок хаотична или невозможна, ошибки не устраняются; 1 – использует отладчик с подсказками, локализует очевидные ошибки, устранение требует дополнительной помощи; 2 – уверенно использует отладчик и анализ логов, быстро и	2	1	2

			точно локализует ошибки, эффективно устраняет их, не нарушая работу смежных модулей;			
Осуществлять документирование программных модулей программного обеспечения	документирован ия кода, API и интерфейсов	создаёт техническую документацию, описывает контракты API, оформляет README, поддерживает актуальные диаграммы взаимодействия	0 – документация отсутствует или не соответствует действительности, контракты API не описаны; 1 – документация создана частично, содержит неточности или устаревшие сведения, описаны не все интерфейсы; 2 – самостоятельно создаёт полную и актуальную документацию, чётко описывает контракты API, предоставляет инструкции по развёртыванию и примеры использования;	2	1	2
Осуществлять документирование программных модулей программного обеспечения	программировать с использованием комментариев для документирования	добавляет комментарии к коду, оформляет JSDoc/DocString для функций и классов	0 – комментарии отсутствуют или формальны («магия»), не помогают понять код; 1 – комментарии есть, но местами избыточны или недостаточны, оформление документации частично; 2 – код снабжён лаконичными и содержательными комментариями, публичные функции и классы оформлены согласно стандарту документации, что обеспечивает лёгкость чтения и поддержки;	2	0.5	1

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	Традиционная технология обучения (Я.А.Коменский, И.Ф.Герbart)	Усвоение обучающимися новых знаний	Формирование знаний и воспроизведение усвоенного знания	Использование объяснительно иллюстративного метода во время проведения теоретических уроков
2	Технология электронного обучения (Беляев М.И.)	Формирование умений у обучающихся. Интерактивная форма первичного закрепления. Интерактивная форма текущего контроля	Сокращение времени проведения первичного закрепления материала и текущего контроля. Повышение интереса обучающихся к дисциплине.	1. Домашние задания с использованием электронных учебников или Интернет-ресурсов. 2. Использование средств вычислительной техники для формирования умений на практических занятиях 3. Использование средств вычислительной техники для контроля знаний
3	Информационно - коммуникационная технология	Ускорение интеллектуальной деятельности за счет использования компьютерных и телекоммуникационных технологий	Повышение уровня усваиваемых знаний и формируемых умений обучающихся. Повышение интереса обучающихся к дисциплине	1. Демонстрация презентации, использование графических и видеоматериалов на этапах изучения нового материала. 2. Выполнение элементов практических работ совместно с 95 обучающимися и

				трансляция процесса выполнения на экран с использованием проектора.
4	Технология знаково контекстного обучения (А.А. Вербицкий)	Моделирование в учебной деятельности целостного содержания, форм и условий профессиональной деятельности	Моделирование предметного содержания будущей профессиональной деятельности обучающихся. Усвоение новой информации обучающимися в контексте профессиональных практических действий и поступков	В процессе урока обучающиеся проходят несколько этапов индивидуального и совместного анализа и решения профессиональных ситуаций

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