

Документ подписан Министерством науки и высшего образования Российской Федерации  
Информация о владельце:  
ФИО: Макаренко Елена Николаевна  
Должность: Ректор  
Дата подписания: 17.06.2026 13:24:14  
Уникальный программный ключ:  
c098bc0c1041cb2a4cf926cf171d6715d99a6ae00adc8e27b55cbe1e2dbd7c78

Федеральное государственное бюджетное образовательное учреждение высшего образования «Ростовский государственный экономический университет (РИНХ)»  
Финансово-экономический колледж



УТВЕРЖДАЮ

Директор

Р. А. Сычев

2026г.

## Рабочая программа МДК Обеспечение качества программного обеспечения

Специальность  
09.02.12 ТЕХНИЧЕСКАЯ ЭКСПЛУАТАЦИЯ И СОПРОВОЖДЕНИЕ  
ИНФОРМАЦИОННЫХ СИСТЕМ

Форма обучения	очная
Часов по учебному плану	144
в том числе:	
аудиторные занятия	100
самостоятельная работа	38

Ростов-на-Дону  
2026 г.

### Распределение часов дисциплины по

Семестр (<Курс>.<Семестр на курсе>)	4 (2.2)		Итого	
	10			
Неделя	10			
Вид занятий	УП	РП	УП	РП
Лекции	40	40	40	40
Практические	40	40	40	40
Курсовое	20	20	20	20
Итого ауд.	100	100	100	100
Контактная работа	100	100	100	100
Сам. работа	38	38	38	38
Часы на контроль	6	6	6	6
Итого	144	144	144	144

### ОСНОВАНИЕ

Федеральный государственный образовательный стандарт среднего профессионального образования по специальности 09.02.12 ТЕХНИЧЕСКАЯ ЭКСПЛУАТАЦИЯ И СОПРОВОЖДЕНИЕ ИНФОРМАЦИОННЫХ СИСТЕМ (Приказ Министерства просвещения Российской Федерации от 10 марта 2025г № 184)

Рабочая программа составлена по образовательной программе 09.02.12 ТЕХНИЧЕСКАЯ ЭКСПЛУАТАЦИЯ И СОПРОВОЖДЕНИЕ ИНФОРМАЦИОННЫХ СИСТЕМ для набора 2026 года  
программа среднего профессионального образования

Учебный план утвержден учёным советом вуза от 03.03.2026 протокол № 9

Программу составил(и): Преп., Финтисова А.И.

Председатель ЦМК: Ламин В.А.

Рассмотрено на заседании ЦМК от 06.03.2026 протокол № 7

## 1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

- |     |   |
|-----|---|
| 1.1 | освоение вида деятельности «Сопровождение процессов тестирования в процессе эксплуатации» |
|-----|---|

## 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Цикл (раздел) ООП: МДК.02

### 2.1 Требования к предварительной подготовке обучающегося:

2.1.1 Архитектура аппаратных средств и основы сетевых технологий

2.1.2 Базы данных

2.1.3 Внедрение ИС

### 2.2 Дисциплины и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее:

2.2.1 Внедрение ИС

## 3. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

### 3.1 Знать

#### ПК 2.1 Осуществлять подготовку тестовых данных в соответствии с заданием на тестирование программного обеспечения

Основную терминологию по тестированию ПО

Язык, на котором написана техническая документация тестируемого ПО на уровне, достаточном для чтения технической документации

Основные термины и сокращения, используемые в технической документации и принятые в организации

Процедуры обеспечения безопасности при выполнении тестирования ПО

Область применения инструментальных средств для выполнения тестирования ПО

Особенности основных операционных систем

Требования по обеспечению безопасности аппаратных и программных средств автоматизированных систем, используемых при выполнении тестовых процедур, включая вопросы антивирусной защиты

#### ПК 2.2 Выполнять тестирование программного обеспечения

Нормативно-технические материалы по вопросам испытания и тестирования ПО

Основные термины и сокращения, используемые в технической документации и принятые в организации

Основы работы в операционной системе, в которой производится тестирование, на уровне, необходимом для тестирования ПО соответствующего типа

Основы теории алгоритмов и дискретной математики в объеме полученного профессионального образования

### 3.2 Уметь

#### ПК 2.1 Осуществлять подготовку тестовых данных в соответствии с заданием на тестирование программного обеспечения

Устанавливать корректную последовательность операций при выполнении тестирования ПО

Выявлять недостающую информацию для выполнения тестирования ПО в заданном объеме

Устанавливать операционные системы

Выполнять базовую настройку операционных систем

Подготавливать необходимые средства и ресурсы для выполнения задания по тестированию ПО

Составлять отчет о результатах подготовки к выполнению тестирования ПО

#### ПК 2.2 Выполнять тестирование программного обеспечения

Выполнять модульные тесты с использованием инструментов тестирования, в том числе автоматизированного тестирования

Использовать системы контроля дефектов ПО

Составлять отчет о выполнении тестирования ПО

### 3.3 Владеть

#### ПК 2.1 Осуществлять подготовку тестовых данных в соответствии с заданием на тестирование программного обеспечения

Изучением необходимых для проведения тестирования ПО действий, перечисленных в задании на тестирование

Подготовки тестовых платформ (установка операционной системы, дополнительного ПО и другого по необходимости)

Оценки объема тестирования ПО с целью определения необходимых ресурсов для его выполнения

Настройки тестовой среды и аппаратных средств для выполнения тестирования ПО в соответствии с заданием на тестирование в пределах своей компетенции

Формирования и представления отчетности о подготовке к выполнению задания на тестирование ПО в соответствии с установленными регламентами

**ПК 2.2 Выполнять тестирование программного обеспечения**

Проверкой компонентов инструментария и тестируемого ПО на корректное начальное состояние для начала тестирования

Выполнения тестовых процедур на тестовых данных

Сравнения фактического и ожидаемого результатов выполнения тестовых процедур

Формирования и представления отчетности о выполнении процесса тестирования ПО в соответствии с установленными регламентами

#### 4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Код занятия	Наименование разделов и тем /вид занятия/	Семестр / Курс	Часов	Компетенции	Литература	Примечание
	<b>Раздел 1. Основы обеспечения качества программных приложений</b>					
1.1	Понятие качества программного обеспечения (ГОСТ Р ИСО/МЭК 25051). Жизненный цикл программного продукта. Тестирование в жизненном цикле ПО. /Лек/	4	2	ПК 2.1.	Л1.1 Л1.2Л2.1 Л2.2 Э3	
1.2	Техническое задание: структура, методы анализа требований. Понятие верификации и валидации. /Лек/	4	2	ПК 2.1.	Л1.2 Л1.3Л2.1 Л2.2 Э2 Э3	
1.3	Базовые метрики качества ПО. Понятие стратегии тестирования. Определение целей тестирования. /Лек/	4	2	ПК 2.1.	Л1.2 Л1.3Л2.1 Л2.2 Э2 Э3	
1.4	Типовая архитектура современных ИС. Уровни тестирования, пирамида тестирования. /Лек/	4	2	ПК 2.1.	Л1.1 Л1.2Л2.1 Э2 Э3	
1.5	Основные понятия конвейерного подхода (pipelines), основные принципы CI/CD /Лек/	4	2	ПК 2.1.	Л1.1 Л1.2Л2.1 Э2 Э3	
1.6	Анализ требований технического задания на непротиворечивость /Пр/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Э1 Э2	
1.7	Анализ требований технического задания на непротиворечивость /Пр/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Э1 Э2	
1.8	Определение целей тестирования для заданных уровней тестирования /Пр/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Э2 Э3	

1.9	Определение целей тестирования для заданных уровней тестирования /Пр/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Э2 Э3	
1.10	Определение целей тестирования для заданных уровней тестирования /Ср/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2 Э3	
1.11	Определение целей тестирования для заданных уровней тестирования /Ср/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2 Э3	
1.12	Определение целей тестирования для заданных уровней тестирования /Ср/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2 Э3	
1.13	Определение целей тестирования для заданных уровней тестирования /Ср/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2 Э3	
1.14	Определение целей тестирования для заданных уровней тестирования /Ср/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2 Э3	
1.15	Определение целей тестирования для заданных уровней тестирования /Ср/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2 Э3	
	<b>Раздел 2. Тест-дизайн</b>					
2.1	Тестовый сценарий. Тестовый план. Чек -лист. Тестовый пакет, задание на тестирование. /Лек/	4	2	ПК 2.1.	Л1.1 Л1.2Л2.1 Л2.2 Э3	
2.2	Классификация видов тестирования по различным основаниям. Принципы проектирования сценариев для функционального и нефункционального тестирования. /Лек/	4	2	ПК 2.1.	Л1.1 Л1.2Л2.1 Л2.2 Э2 Э3	
2.3	Выдача задания на курсовой проект. Основные принципы и методы. Анализ технического задания. /Курс пр/	4	2	ПК 2.1.	Л1.1 Л1.2Л2.1 Л2.2 Э2 Э3	
2.4	«Черный ящик» или типы, основанные на спецификациях: эквивалентное разбиение, анализ граничных значений, использование таблиц решений, диаграммы причинно-следственных связей, тестирование переходов состояний, тестирование на основе сценариев использования /Лек/	4	2	ПК 2.1.	Л1.1 Л1.2Л2.1 Л2.2 Э2 Э3	

2.5	Методы статического тестирования. /Лек/	4	2	ПК 2.1.	Л1.1 Л1.2Л2.1 Э2 Э3	
2.6	Шаблоны тестов. Основные инструменты проектирования тестов /Лек/	4	2	ПК 2.1.	Л1.1 Л1.2Л2.1 Э2 Э3	
2.7	Особенности организации тестирования безопасности, стрессового и нагрузочного тестирования информационных систем. /Лек/	4	2	ПК 2.1.	Л1.1 Л1.2Л2.1 Э2 Э3	
2.8	Особенности тест-дизайна для различных видов приложений – веб, настольных, мобильных. /Лек/	4	2	ПК 2.1.	Л1.1 Л1.2Л2.1 Э2 Э3	
2.9	Подготовка тестового пакета и задания на тестирование модуля и его размещение в системе контроля версий /Пр/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Э2 Э3	
2.10	Подготовка тестового пакета и задания на тестирование модуля и его размещение в системе контроля версий /Пр/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Э2 Э3	
2.11	Проектирование тест-кейсов для интеграционного тестирования с использованием инструментария его размещение в системе контроля версий /Пр/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э2 Э3	
2.12	Проектирование тест-кейсов для интеграционного тестирования с использованием инструментария его размещение в системе контроля версий /Пр/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э2 Э3	
2.13	Подготовка тестового сценария e2e для веб-приложения с использованием инструментария его размещение в системе контроля версий /Пр/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э2 Э3	
2.14	Подготовка тестового сценария e2e для веб-приложения с использованием инструментария его размещение в системе контроля версий /Пр/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э2 Э3	
2.15	Разработка фреймворка автотестов. Реализация e2e тестирования для web-системы /Курс пр/	4	2	ПК 2.1.	Л1.1 Л1.2Л2.1 Э2 Э3	
2.16	Разработка фреймворка автотестов. Реализация e2e тестирования для web-системы /Курс пр/	4	2	ПК 2.1.	Л1.1 Л1.2Л2.1 Э2 Э3	

2.17	Проектирование тест-кейсов для интеграционного тестирования с использованием инструментария его размещение в системе контроля версий /Ср/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2 Э3	
2.18	Проектирование тест-кейсов для интеграционного тестирования с использованием инструментария его размещение в системе контроля версий /Ср/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2 Э3	
2.19	Проектирование тест-кейсов для интеграционного тестирования с использованием инструментария его размещение в системе контроля версий /Ср/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2 Э3	
2.20	Проектирование тест-кейсов для интеграционного тестирования с использованием инструментария его размещение в системе контроля версий /Ср/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2 Э3	
2.21	Проектирование тест-кейсов для интеграционного тестирования с использованием инструментария его размещение в системе контроля версий /Ср/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2 Э3	
2.22	Проектирование тест-кейсов для интеграционного тестирования с использованием инструментария его размещение в системе контроля версий /Ср/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2 Э3	
	<b>Раздел 3. Дефекты ПО и тестовые наборы</b>					
3.1	Жизненный цикл дефекта. Уровни критичности дефектов. /Лек/	4	2	ПК 2.1.	Л1.1 Л1.2Л2.1 Л2.2 Э3	
3.2	Особенности проектирование тестовых наборов для различных методов статического и динамического тестирования. /Лек/	4	2	ПК 2.1.	Л1.1 Л1.2Л2.1 Л2.2 Э3	
3.3	Инструменты автоматизации подготовки тестовых данных. /Лек/	4	2	ПК 2.1.	Л1.1 Л1.2Л2.1 Л2.2 Э3	
3.4	Инструменты автоматизации подготовки тестовых данных. /Лек/	4	2	ПК 2.1.	Л1.1 Л1.2Л2.1 Л2.2 Э3	

3.5	Методы оптимизации тестовых наборов. Оценка тестов на покрытие требований /Лек/	4	2	ПК 2.1.	Л1.1 Л1.2Л2.1 Л2.2 Э3	
3.6	Методы оптимизации тестовых наборов. Оценка тестов на покрытие требований /Лек/	4	2	ПК 2.1.	Л1.1 Л1.2Л2.1 Л2.2 Э3	
3.7	Методы отбора тестов для регрессионного тестирования с учетом критичности выявленных дефектов. /Лек/	4	2	ПК 2.1.	Л1.1 Л1.2Л2.1 Э3	
3.8	Методы отбора тестов для регрессионного тестирования с учетом критичности выявленных дефектов. /Лек/	4	2	ПК 2.1.	Л1.1 Л1.2Л2.1 Э3	
3.9	Реализация тестирования микросервисной архитектуры Сценарий нагрузочного тестирования и визуализация результатов Реализация сценариев тестирования на основе действий различных групп пользователей (BDD-проект) /Кур пр/	4	2	ПК 2.1.	Л1.1 Л1.2Л2.1 Э3	
3.10	Разработка тестового набора для тестирования модуля методом «белого ящика» /Пр/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э2 Э3	
3.11	Разработка тестового набора для тестирования модуля методом «белого ящика» /Пр/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э2 Э3	
3.12	Разработка тестового набора для тестирования модуля методом «белого ящика» /Пр/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э2 Э3	
3.13	Реализация тестов по REST API Интеграция автотестов в актуальную версию приложения UI автотесты с генерацией отчетов с использованием инструментария /Кур пр/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э2 Э3	
3.14	Разработка тестового набора для тестирования веб-приложения с имитацией действий пользователя /Пр/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э2 Э3	
3.15	Разработка тестового набора для тестирования веб-приложения с имитацией действий пользователя /Пр/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э2 Э3	
3.16	Реализация мок-сервиса и его тестирование	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1	

	Реализация полного цикла регрессионного тестирования с учетом автоматизации Поддержка тестов и работа с нестабильными компонентами /Кур пр/				Л2.2 Э2 Э3	
3.17	Разработка тестового набора для тестирования веб-приложения с имитацией действий пользователя /Пр/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э2 Э3	
3.18	Генерация тестовых данных для тестирования интеграции с базой данных с помощью заданного инструментария /Пр/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2	
3.19	Генерация тестовых данных для тестирования интеграции с базой данных с помощью заданного инструментария /Пр/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2	
3.20	Расширение проекта за счёт тестов безопасности (OWASP) Построение системы отслеживания и отчётности по тестам Тестирование отказоустойчивости распределённой системы /Кур пр/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2	
3.21	Оптимизация тестовых наборов и оценка тестов /Пр/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2	
3.22	Анализ покрытия автотестами и внедрение мониторинга результатов тестирования Разработка тестов и результаты тестирования на стабильность и производительность базы данных /Кур пр/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2	
3.23	Оптимизация тестовых наборов и оценка тестов /Пр/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2	
3.24	Интеграция тестов в систему баг-трекинга и сбор аналитики Разработка тестовой инфраструктуры под API-интеграции /Кур пр/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2	
3.25	Генерация тестовых данных для тестирования интеграции с базой данных с помощью заданного инструментария /Ср/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2 Э3	
3.26	Генерация тестовых данных для тестирования интеграции с базой данных с помощью	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2	

	заданного инструментария /Ср/				Э1 Э2 Э3	
3.27	Генерация тестовых данных для тестирования интеграции с базой данных с помощью заданного инструментария /Ср/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2 Э3	
3.28	Генерация тестовых данных для тестирования интеграции с базой данных с помощью заданного инструментария /Ср/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2 Э3	
3.29	Генерация тестовых данных для тестирования интеграции с базой данных с помощью заданного инструментария /Ср/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2 Э3	
3.30	Генерация тестовых данных для тестирования интеграции с базой данных с помощью заданного инструментария /Ср/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2 Э3	
3.31	Защита курсового проекта /Курс пр/	4	2	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2 Э3	
3.32	Экзамен	4	6	ПК 2.1. ПК 2.2.	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2 Э3	

## 5. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

### 5.1. Фонд оценочных средств для проведения промежуточной аттестации

Промежуточная аттестация проходит в форме экзамена. Перечень вопросов к экзамену:

1. Понятие качества программного обеспечения (ГОСТ Р ИСО/МЭК 25051).
2. Жизненный цикл программного продукта
3. Тестирование в жизненном цикле ПО.
4. Техническое задание: структура, методы анализа требований
5. Понятие верификации и валидации.
6. Базовые метрики качества ПО.
7. Понятие стратегии тестирования
8. Определение целей тестирования
9. Типовая архитектура современных ИС
10. Уровни тестирования, пирамида тестирования.
11. Основные понятия конвейерного подхода (pipelines), основные принципы CI/CD
12. Тестовый сценарий
13. Тестовый план
14. Чек-лист
15. Тестовый пакет, задание на тестирование.
16. Классификация видов тестирования по различным основаниям
17. Принципы проектирования сценариев для функционального и нефункционального тестирования
18. «Черный ящик» или типы, основанные на спецификациях: эквивалентное разбиение, анализ граничных значений, использование таблиц решений, диаграммы причинно-следственных связей, тестирование переходов состояний, тестирование на основе сценариев использования
19. Методы статического тестирования
20. Шаблоны тестов
21. Основные инструменты проектирования тестов
22. Особенности организации тестирования безопасности, стрессового и нагрузочного тестирования информационных систем

23. Особенности тест-дизайна для различных видов приложений – веб, настольных, мобильных
24. Жизненный цикл дефекта
25. Уровни критичности дефектов
26. Особенности проектирование тестовых наборов для различных методов статического и динамического тестирования
27. Инструменты автоматизации подготовки тестовых данных
28. Методы оптимизации тестовых наборов
29. Оценка тестов на покрытие требований
30. Методы отбора тестов для регрессионного тестирования с учетом критичности выявленных дефектов

*Критерии оценивания:*

5 баллов выставляется студентам за полный и правильный ответ на все вопросы билета с логическим обоснованием аргументов, в ответе нет ошибок.

4 балла выставляется студентам, если вопросы билета раскрыты полностью, но обоснования доказательства недостаточны, при этом допущены две-три несущественные ошибки, исправленные по требованию преподавателя.

3 балла ставится студентам за правильный ответ на вопросы билета, при этом допущено более одной ошибки по изложению фактов или более двух-трех недочетов в ответе.

2 балла ставится студентам, если допущены существенные ошибки, показавшие, что обучающийся не обладает обязательными умениями по данной теме в полной мере.

**5.2. Фонд оценочных средств для проведения текущего контроля**

Представлен в Приложении 1 к рабочей программе МДК.

**6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ**

**6.1. Рекомендуемая литература**

**6.1.1. Основная литература**

	Авторы,	Заглавие	Издательство, год	Колич-во
Л1.1	Черткова Е. А.	Программная инженерия. Визуальное моделирование программных систем: учебник для спо	Москва: Юрайт, 2023	<a href="https://urait.ru/book/">https://urait.ru/book/</a> неограниченный доступ для зарегистрированных пользователей
Л1.2	Гниденко И. Г., Павлов Ф. Ф., Федоров Д. Ю.	Технология разработки программного обеспечения: учебное пособие для спо	Москва: Юрайт, 2023	<a href="https://urait.ru/book/">https://urait.ru/book/</a> неограниченный доступ для зарегистрированных пользователей
Л1.3	Черткова Е. А.	Программная инженерия. Визуальное моделирование программных систем: учебник для спо	Москва: Юрайт, 2023	<a href="https://urait.ru/book/">https://urait.ru/book/</a> неограниченный доступ для зарегистрированных пользователей

**6.1.2. Дополнительная литература**

	Авторы,	Заглавие	Издательство, год	Колич-во
--	---------	----------	-------------------	----------

Л2.1	Золотов С. Ю.	Проектирование информационных систем: Учебное пособие	Томск: Томский государственный университет систем управления и радиоэлектроники.	<a href="http://biblioclub.ru/">http://biblioclub.ru/</a> - неограниченный доступ для
Л2.2	Зараменских Е. П.	Информационные системы: управление жизненным циклом: учебник и практикум для спо	Москва: Юрайт, 2023	<a href="https://urait.ru/book/">https://urait.ru/book/</a> неограниченный доступ для зарегистрированных пользователей

### **6.2. Перечень ресурсов информационно-телекоммуникационной сети "Интернет"**

Э1	Ежемесячный компьютерный журнал КомпьютерПресс –
Э2	Технология разработки программного обеспечения - <a href="https://kvckr.me/mag/trpo/">https://kvckr.me/mag/trpo/</a>
Э3	Компьютерные книги: <a href="https://www.compress.ru/">https://www.compress.ru/</a>

### **6.3. Перечень программного обеспечения**

6.3.1	Офисный пакет LibreOffice
-------	---------------------------

### **6.4 Перечень информационных справочных систем**

6.4.1	ИСС «КонсультантПлюс»
6.4.2	ИИС «Гарант»

## **7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)**

7.1	Помещения для проведения всех видов работ, предусмотренных учебным планом, укомплектованы необходимой специализированной учебной мебелью и техническими средствами обучения.
-----	--

## **8. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ**

Методические указания по освоению дисциплины представлены в Приложении 2 к рабочей программе МДК.

## ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

### МДК.02.01 Обеспечение качества программного обеспечения

#### 1. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

##### 1.1 Показатели и критерии оценивания компетенций:

УУД, составляющие компетенцию	Показатели оценивания	Критерии оценивания	Средства оценивания
<b>ПК 2.1. Осуществлять подготовку тестовых данных в соответствии с заданием на тестирование программного обеспечения</b>			
<p><b>Знать:</b>  Основную терминологию по тестированию ПО  Язык, на котором написана техническая документация тестируемого ПО на уровне, достаточном для чтения технической документации  Основные термины и сокращения, используемые в технической документации и принятые в организации  Процедуры обеспечения безопасности при выполнении тестирования ПО  Область применения инструментальных средств для выполнения тестирования ПО  Особенности основных операционных систем  Требования по обеспечению безопасности аппаратных и программных средств автоматизированных систем, используемых при выполнении тестовых процедур, включая вопросы антивирусной защиты</p>	<p><b>Получение систематических знаний</b> о методах подготовки тестовых данных и анализа информации в программировании; основных инструментальных средств тестирования</p>	<p><b>Уровень знаний</b> методов подготовки тестовых данных и анализа информации в программировании; основных инструментальных средств тестирования</p>	<b>ПЗ (1-10)</b>
<p><b>Уметь:</b>  Устанавливать корректную последовательность операций при выполнении тестирования ПО  Выявлять недостающую информацию для выполнения тестирования ПО в заданном объеме  Устанавливать операционные системы  Выполнять базовую настройку операционных систем  Подготавливать необходимые</p>	<p><b>Сформировать систематическое умение</b> по осуществлению выполнения базовой настройки операционных систем и анализа информации в программировании; интерпретация и применение полученных данных.</p>	<p><b>Уровень умения</b> выполнению базовой настройки операционных систем и анализа информации в программировании анализ и интерпретацию информации в программировании.</p>	<b>ПЗ (1-10)</b>

<p>средства и ресурсы для выполнения задания по тестированию ПО</p> <p>Составлять отчет о результатах подготовки к выполнению тестирования ПО</p>			
<p><b>Владеть:</b></p> <p>Изучением необходимых для проведения тестирования ПО действий, перечисленных в задании на тестирование</p> <p>Подготовки тестовых платформ (установка операционной системы, дополнительного ПО и другого по необходимости)</p> <p>Оценки объема тестирования ПО с целью определения необходимых ресурсов для его выполнения</p> <p>Настройки тестовой среды и аппаратных средств для выполнения тестирования ПО в соответствии с заданием на тестирование в пределах своей компетенции</p> <p>Формирования и представления отчетности о подготовке к выполнению задания на тестирование ПО в соответствии с установленными регламентами</p>	<p><b>Систематическое владение</b> навыками критического мышления при оценке информации;</p> <p>способность принимать обоснованные решения на основе данных.</p>	<p><b>Уровень владения</b> навыками критического мышления и способностью принимать обоснованные решения на основе анализа данных.</p>	<p><b>ПЗ (1-10), КП(1-43)</b></p>
<p><b>ПК 2.2: Выполнять тестирование программного обеспечения.</b></p>			
<p><b>Знать:</b></p> <p>Нормативно-технические материалы по вопросам испытания и тестирования ПО</p> <p>Основные термины и сокращения, используемые в технической документации и принятые в организации</p> <p>Основы работы в операционной системе, в которой производится тестирование, на уровне, необходимом для тестирования ПО соответствующего типа</p> <p>Основы теории алгоритмов и дискретной математики в объеме полученного профессионального образования</p>	<p><b>Получение систематических знаний</b> в области современных технологий и инструментов тестирования программного обеспечения</p>	<p><b>Уровень знаний</b> – понимать модели и принципы организации тестирования программного обеспечения</p>	<p><b>ПЗ (1-10)</b></p>
<p><b>Уметь:</b></p> <p>Выполнять модульные тесты с использованием инструментов тестирования, в том числе автоматизированного</p>	<p><b>Сформировать систематическое умение</b> в организации, заданной тестирования модулей</p>	<p><b>Уровень умения</b> - организовать требуемую тестирования</p>	<p><b>ПЗ (1-10)</b></p>

тестирования Использовать системы контроля дефектов ПО Составлять отчет о выполнении тестирования ПО			
<b>Владеть:</b> Проверкой компонентов инструментария и тестируемого ПО на корректное начальное состояние для начала тестирования Выполнения тестовых процедур на тестовых данных Сравнения фактического и ожидаемого результатов выполнения тестовых процедур Формирования и представления отчетности о выполнении процесса тестирования ПО в соответствии с установленными регламентами	<b>Сформировать систематическое владение</b> в проверки и сравнении тестируемого программного обеспечения	<b>Уровень владения</b> - проверять и сравнивать тестируемое программное обеспечение в соответствии с требованиями.	<b>ПЗ (1-10), КП(1-43)</b>

*ПЗ - практические задания, КП- курсовой проект.*

## **2. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы**

### **Практические задания:**

Задание 1. Анализ требований технического задания на непротиворечивость

Условие

Дана выдержка из Технического задания (ТЗ) на мобильное приложение «Умный будильник»:

Требования:

- Приложение должно работать на iOS и Android.
- Установка будильника возможна только на время от 4:00 до 10:59 включительно.
- Повтор будильника: ежедневно, по будням, по выходным.
- Максимальное количество будильников — 10.
- Если пользователь не выключил будильник в течение 3 минут после срабатывания, приложение автоматически увеличивает громкость до максимума.
- Приложение должно отправлять push-уведомление за 15 минут до срабатывания будильника.
- На Android уведомления приходят всегда. На iOS — только если приложение активно или в фоне (но не после принудительного закрытия).
- Увеличение громкости работает только на iOS.
- При установке будильника на время ранее 4:00 показывается ошибка «Слишком рано».
- При установке на время позже 10:59 ошибка: «Слишком поздно».

Задание

- Выявите все логические противоречия, неоднозначности и неполноту требований.
- Сформулируйте уточняющие вопросы к заказчику / аналитику.
- Предложите исправленные / дополненные формулировки требований.

Задание 2. Определение целей тестирования для заданных уровней тестирования

Условие

- Разрабатывается веб-сервис для онлайн-записи к врачам (клиентская часть + API + база данных).
- Вам поручено определить цели тестирования для четырёх уровней:
- Модульное тестирование (Unit)
- Интеграционное тестирование
- Системное тестирование
- Приёмочное тестирование

Задание

Для каждого уровня сформулируйте не менее 2-х конкретных целей, связанных именно с этим уровнем, а не с другими.

Задание 3. Подготовка тестового пакета и задания на тестирование модуля + размещение в Git

Задание

Часть А. Создание тестового пакета (например, на pytest)

- Напишите минимум 4 теста для add и divide:
- нормальные случаи
- граничные значения (например, отрицательные числа, ноль)
- обработка исключений (divide на ноль)

Часть Б. Задание на тестирование модуля

- Опишите в формате Markdown (файл TEST\_PLAN\_module.md):
- Цель тестирования модуля
- Перечень тестовых сценариев (с идентификаторами)
- Входные данные и ожидаемые результаты
- Критерии прохождения / непрохождения

Задание 4. Проектирование тест-кейсов для интеграционного тестирования + размещение в Git

Задание

- Спроектируйте не менее 4 тест-кейсов для интеграционного тестирования (формат: ID, предварительные условия, шаги, ожидаемый результат, постусловия).
- Оформите тест-кейсы в файле integration\_test\_cases.md или в виде CSV, пригодного для импорта в Test Management System.
- Создайте интеграционный тест на языке Python (используя requests и pytest), который реально вызывает API (можно мокать User Service через responses или unittest.mock).

Задание 5. Подготовка тестового сценария E2E для веб-приложения + Git

Задание

Напишите один сквозной E2E-сценарий на Cypress или Playwright, который:

- Заходит на страницу
- Добавляет 2 задачи
- Отмечает первую как выполненную
- Переключается на фильтр Active (проверяет, что осталась только вторая)
- Возвращается на фильтр All
- Удаляет первую задачу
- Проверяет, что осталась одна задача
- Сценарий должен быть воспроизводимым, содержать ожидания (waitFor) и проверки (assert).

Задание 6. Разработка тестового набора для тестирования модуля методом «белого ящика»

Задание

- Разработайте тестовый набор (набор тестовых случаев) с использованием техник белого ящика:
- Покрытие операторов (statement coverage) — минимум тестов для 100% покрытия всех строк.

- Покрытие ветвей (branch coverage) — все возможные пути (if/else).
- Покрытие условий (condition coverage) — все подусловия (len < 6, has\_digit, has\_upper и т.д.).
- Метод граничных значений (границы: 5 и 6 символов).
- Метод классов эквивалентности (слабая/средняя/сильная сложность).

Задание 7. Анализ требований технического задания на непротиворечивость

Условие

Проанализировать ТЗ на «Умный будильник».

Задание 8. Оптимизация тестовых наборов и оценка тестов

Условие

Дан тестовый набор из 25 тестов. Оптимизировать и оценить ценность каждого.

Задание 9. Генерация тестовых данных для тестирования интеграции с БД

Условие

Сгенерировать 500 записей для таблицы users и orders с помощью Faker.

Задание 10. Разработка тестового набора для тестирования веб-приложения с имитацией действий пользователя

Условие

Разработать тестовый набор из 5 тест-кейсов для формы авторизации веб-приложения.

#### **Критерии оценивания:**

- 5 баллов выставляется, если правильные ответы даны на 85-100% практических заданий
- 4 балла выставляется студенту, если правильные ответы даны на 65-84% практических заданий
- 3 балла выставляется студенту, если правильные ответы даны на 50-64% практических заданий
- 2 балла выставляется студенту, если правильные ответы даны на менее 50% практических заданий.

#### **Темы курсовых проектов:**

1. Разработка фреймворка автотестов
2. Реализация e2e тестирования для web-системы
3. Реализация тестирования микросервисной архитектуры
4. Сценарий нагрузочного тестирования и визуализация результатов
5. Реализация сценариев тестирования на основе действий различных групп пользователей (BDD-проект)
6. Реализация тестов по REST API
7. Интеграция автотестов в актуальную версию приложения
8. UI автотесты с генерацией отчетов с использованием инструментария
9. Реализация мок-сервиса и его тестирование
10. Реализация полного цикла регрессионного тестирования с учетом автоматизации
11. Поддержка тестов и работа с нестабильными компонентами
12. Расширение проекта за счёт тестов безопасности (OWASP)
13. Построение системы отслеживания и отчётности по тестам
14. Тестирование отказоустойчивости распределённой системы
15. Анализ покрытия автотестами и внедрение мониторинга результатов тестирования
16. Разработка тестов и результаты тестирования на стабильность и производительность базы данных
17. Интеграция тестов в систему баг-трекинга и сбор аналитики
18. Разработка тестовой инфраструктуры под API-интеграции
19. Разработка гибридного фреймворка автотестов с поддержкой разных уровней тестирования (UI + API + Unit)

20. Реализация параллельного выполнения автотестов с динамическим распределением тестов по агентам
21. Интеграция системы визуального регрессионного тестирования (скриншот-сравнение) в CI/CD пайплайн
22. Разработка модуля автоматической генерации тестовых данных с учетом граничных условий и комбинаторного взрыва
23. Автоматизация развёртывания тестового окружения с использованием Docker Compose и Kubernetes для микросервисной системы
24. Реализация тестов "времени жизни" (longevity testing) контейнеризованного приложения с мониторингом утечек ресурсов
25. Внедрение автоматизированного сканирования зависимостей на уязвимости (SCA) в процесс сборки проекта
26. Реализация chaos-тестирования (сбои сети, отказ дисков, высокая задержка) для распределённой системы с оценкой времени восстановления
27. Разработка тестов на инъекции SQL, XSS и CSRF с генерацией отчетов для веб-приложения
28. Сравнительный анализ результатов нагрузочного тестирования при разных стратегиях балансировки трафика
29. Разработка сценария стресс-тестирования с выходом за пределы ёмкости и автоматическим определением точки отказа
30. Построение дашборда реального времени для отслеживания статуса автотестов и метрик стабильности (flaky-тесты, время выполнения)
31. Разработка системы автоматической классификации упавших тестов по корневой причине (функциональная ошибка, проблема инфраструктуры, нестабильность)
32. Разработка набора тестов в формате Gherkin с интеграцией в процесс непрерывной поставки (Jenkins/GitLab CI)
33. Автоматизация генерации живой документации (living documentation) на основе BDD-сценариев и актуальных результатов прогонов
34. Реализация контрактного тестирования (Pact/Spring Cloud Contract) для взаимодействия микросервисов
35. Автоматизация тестирования асинхронных сценариев (очереди, Kafka, WebSocket) с контролем порядка и таймингов сообщений
36. Разработка тестов для проверки миграций схемы БД без потери данных и с откатом на предыдущую версию
37. Автоматизированное тестирование репликации и консистентности данных в распределенной БД (Cassandra, CockroachDB)
38. Разработка плагина для Allure/ReportPortal для кастомизированной визуализации результатов тестирования микросервисов
39. Интеграция автоматического воспроизведения багов (self-healing тестов) с использованием эвристик поиска элементов
40. Автоматизация триггеров запуска тестов на основе изменённых файлов в кодовой базе (смарт-тестирование)
41. Разработка системы «канареечного» развёртывания с автоматическими проверками стабильности перед переключением трафика
42. Адаптация E2E автотестов для веб-системы под мобильные браузеры с эмуляцией сенсорного ввода
43. Разработка тестов для PWA-приложения (офлайн-режим, push-уведомления, обновление service worker)

#### **Критерии оценивания:**

**-5 баллов** выставляется за работу, написанную на актуальную тему и имеющую практическую значимость. При этом студенту необходимо максимально полно раскрыть тему курсовой работы, проанализировать литературные источники, использовать современные научные методики исследования, оформить с учётом требований ГОСТа.

**-4 балла** выставляется за работу, которая в целом соответствующая вышеприведенным требованиям, но в которой допущены следующие недостатки при раскрытии темы упущены некоторые существенные вопросы, или не нашли отражения современные научные данные, содержащиеся в литературе, или обнаружилось недостаточное использование современной

нормативной базы, или допущено пять ошибок в оформлении. Оценка снижается также за неточные ответы на вопросы.

**-3 балла** ставится за работу, написанную на актуальную тему, правильно оформленную, но, при защите которой студент показал поверхностные теоретические и практические знания, отсутствие умений четко ориентироваться в защищаемой теме. Оценка снижается также при наличии совокупности двух и более замечаний, если в ходе защиты студент не смог убедительно ответить на претензии к своей работе.

**-2 балла** оценивается работа, выполненная на низком теоретическом уровне, не имеющая практической значимости, при защите которой студент не смог ответить на поставленные вопросы.

### **3. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций**

Процедура оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций состоит из текущего контроля.

**Текущий контроль** успеваемости проводится с использованием оценочных средств, представленных в п. 2 данного приложения. Результаты текущего контроля доводятся до сведения студентов до промежуточной аттестации и учитываются при оценивании знаний, умений, навыков и (или) опыта деятельности.

## МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)

### МДК.02.01 Обеспечение качества программного обеспечения

Методические указания для студентов по освоению дисциплины задачи МДК.02.01. Обеспечение качества программного обеспечения являются частью рабочей программы дисциплины (РПД) (приложением к рабочей программе).

РПД – рабочая программа, утвержденная директором колледжа для изучения дисциплины МДК.02.01. Обеспечение качества программного обеспечения. Она определяет цели и задачи дисциплины, формируемые в ходе ее изучения компетенции и их компоненты, содержание изучаемого материала, виды занятий и объем выделяемого учебного времени, а также порядок изучения и преподавания дисциплины МДК.02.01. Обеспечение качества программного обеспечения.

Для самостоятельной учебной работы студента важное значение имеют разделы «Структура и содержание дисциплины (модуля)» и «Учебно-методическое и информационное обеспечение дисциплины (модуля)». В первом указываются разделы и темы изучаемой дисциплины МДК.02.01. Обеспечение качества программного обеспечения, а также виды занятий и планируемый объем (в академических часах), во втором – рекомендуемая литература и перечень ресурсов информационно-телекоммуникационной сети «Интернет».

Для подготовки к текущему контролю студенты могут воспользоваться оценочными средствами, представленными в Приложении 1 к рабочей программе дисциплины.

#### 1. Описание последовательности действий студента

Приступая к изучению дисциплины МДК.02.01. Обеспечение качества программного обеспечения необходимо в первую очередь ознакомиться содержанием РПД, где в разделе «Структура и содержание дисциплины (модуля)» приведено общее распределение часов аудиторных занятий и самостоятельной работы по темам дисциплины МДК.02.01. Обеспечение качества программного обеспечения.

Залогом успешного освоения дисциплины МДК.02.01. Обеспечение качества программного обеспечения является регулярное посещение занятий и выполнение предусмотренных программой заданий. Пропуск одного, а тем более нескольких занятий может осложнить освоение разделов курса.

Лекции имеют целью дать систематизированные основы научных знаний по содержанию дисциплины МДК.02.01. Обеспечение качества программного обеспечения. При изучении и проработке теоретического материала необходимо:

- повторить законспектированный на лекционном занятии материал и дополнить его с учетом рекомендованной по данной теме литературы;
- при самостоятельном изучении теоретической темы подготовить конспект, используя рекомендованные в РПД литературные источники и электронные образовательные ресурсы.

Практические занятия проводятся с целью углубления и закрепления знаний, полученных на лекциях и в процессе самостоятельной работы с учебной литературой.

В процессе практического занятия, как вида учебных занятий, обучающиеся выполняют одно или несколько практических заданий под руководством преподавателя в соответствии с изучаемым содержанием учебного материала.

Выполнение обучающимся практических работ проводится с целью:

- систематизации и закрепления полученных теоретических знаний и практических умений;
- углубления теоретических знаний в соответствии с заданной темой;
- формирования умений применять теоретические знания при решении поставленных задач;
- развития профессиональных компетенций у обучающихся;
- развития творческой инициативы, самостоятельности, ответственности и организованности.

Выполнение обучающимися практических заданий направлено на:

- обобщение, систематизацию, углубление, закрепление полученных теоретических знаний по конкретным темам дисциплины;
- формирование умений применять полученные знания на практике, реализацию единства интеллектуальной и практической деятельности;
- выработку при решении поставленных задач таких профессионально значимых качеств, как самостоятельность, ответственность, точность, творческая инициатива.

При подготовке к практическому занятию необходимо изучить или повторить лекционный материал по соответствующей теме.

## **2. Самостоятельная работа студента**

Самостоятельная работа студента – самостоятельная учебная деятельность студента, организуемая колледжем и осуществляемая без непосредственного руководства педагога, но по его заданиям и под его контролем.

Цели самостоятельной работы:

- систематизация и закрепление полученных теоретических знаний и практических умений студентов;
- углубление и расширение теоретических знаний;
- формирование умений использовать нормативную, правовую, справочную документацию и специальную литературу;
- развитие познавательных способностей и активности студентов: творческой инициативы, самостоятельности, ответственности и организованности;
- формирование самостоятельности мышления, способностей к саморазвитию, самосовершенствованию и самореализации;
- воспитание самостоятельности, как личностного качества будущего специалиста.

Самостоятельная работа студента по дисциплине выполняется:

- самостоятельно вне расписания учебных занятий;
- с использованием современных образовательных технологий;
- работа со специальной литературой для подготовки к тестовым, практическим заданиям.

## **3. Рекомендации по работе с литературой и источниками**

Работу с литературой следует начинать с анализа РПД, содержащей список основной и дополнительной литературы, а также знакомства с учебно-методическими разработками.

В случае возникновения затруднений в понимании учебного материала следует обратиться к другим источникам, где изложение может оказаться более доступным.

Работа с литературой не только полезна как средство более глубокого изучения дисциплины МДК.02.01. Обеспечение качества программного обеспечения, но и является неотъемлемой частью профессиональной деятельности будущего выпускника.