

Документ подписан простой электронной подписью
Информация о владельце: **МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ**
ФИО: Порохня Андрей Алексеевич **РОССИЙСКОЙ ФЕДЕРАЦИИ**
Должность: И.о. директора института перспективной инженерии
Дата подписания: 19.06.2026 14:50:50
Уникальный программный ключ:
990bea3aeec48c23bd8699e48288f8d1960c600

Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

УТВЕРЖДАЮ
И.о. директора института
перспективной инженерии канд.
техн. наук, доцент Порохня А.А.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО ДИСЦИПЛИНЕ

Автоматизация тестирования и качество ПО

Направление подготовки 09.04.04 Программная инженерия

Направленность (профиль)	Разработка, мониторинг и управление жизненным циклом инженерных систем
Год начала обучения	2026
Форма обучения	очная
Реализуется в семестре	2

Введение

Назначение: ФОС предназначен для объективной оценки уровня сформированности компетенций.

ФОС является приложением к программе дисциплины «Автоматизация тестирования и качество ПО».

Разработчик Е.Н. Новикова, к.ф.-м.н., доцент, заведующий межинститутский базовой кафедрой департамента цифровых, робототехнических систем и электроники.

Проведена экспертиза ФОС.

Председатель Новикова Е.Н., межинститутской базовой кафедры департамента цифровых, робототехнических систем и электроники института перспективной инженерии

Члены комиссии:

Азаров И.В., и.о. директора департамента цифровых, робототехнических систем и электроники института перспективной инженерии

Николаев Е. И., доцент департамента цифровых, робототехнических систем и электроники института перспективной инженерии

Представитель организации-работодателя:

Руководитель группы разработки А.А. Шипулин, ООО «Стилсофт».

Экспертное заключение: ФОС по дисциплине позволяет оценить уровень сформированности компетенций. Приступить к апробации.

Срок действия ФОС определяется сроком реализации образовательной программы.

1. Описание критериев оценивания компетенции на различных этапах их формирования, описание шкал оценивания

Уровни сформированности компетенций, индикаторов	Дескрипторы			
	Минимальный уровень не достигнут (Неудовлетворительно) 2 балла	Минимальный уровень (удовлетворительно) 3 балла	Средний уровень (хорошо) 4 балла	Высокий уровень (отлично) 5 баллов
ПК-1 Способен проектировать, реализовывать, оптимизировать и поддерживать сквозные конвейеры непрерывной интеграции, поставки и развертывания (CI/CD) для программного обеспечения и встроенных систем, интегрируя практики автоматизированного тестирования, проверки безопасности, управления артефактами и конфигурациями.				
ИД-1 ПК-1. Проектирует архитектуру CI/CD-конвейеров, определяя этапы сборки, тестирования и развертывания для различных типов приложений и платформ.	Не может спроектировать структуру пайплайна, не различает виды тестирования, не понимает их места в CI/CD. Архитектурная схема отсутствует или не соответствует требованиям.	Способен спроектировать базовый пайплайн из 2-3 этапов (сборка → модульные тесты). Допускает ошибки в последовательности этапов. Слабо обосновывает выбор стратегии тестирования.	Способен спроектировать пайплайн из 4-5 этапов, включая модульные, интеграционные тесты и SAST. Обосновывает выбор порядка этапов. Учитывает тип приложения, но допускает неточности.	Способен спроектировать многостадийный пайплайн (6+ этапов), включая модульные, интеграционные, E2E, нагрузочные тесты, SAST и DAST. Глубоко обосновывает стратегию тестирования на основе пирамиды тестирования и специфики приложения. Учитывает архитектурные особенности.
ИД-2 ПК-1. Реализует автоматизированные процессы непрерывной интеграции, включая компиляцию кода, запуск модульных и интеграционных тестов, статический анализ.	Не умеет писать модульные тесты, не понимает принципов AAA, не может настроить запуск тестов в CI/CD. Допускает грубые синтаксические ошибки.	Разрабатывает модульные тесты (5-8 тестов), но с нарушением принципов изоляции. Покрываемость кода не отслеживает. Интеграционные тесты выполняет только с ручным участием.	Разрабатывает модульные тесты (10+ тестов) с использованием параметризации и фикстур. Настраивает измерение покрытия кода с порогом 70-80%. Пишет интеграционные тесты для API. Настраивает запуск тестов в CI/CD.	Разрабатывает полный набор модульных тестов (15+ тестов) с параметризацией, фикстурами и моками. Достигает покрытия кода $\geq 85\%$. Разрабатывает интеграционные тесты с Testcontainers. Пишет сценарии нагрузочного тестирования на k6. Настраивает параллельный запуск тестов в CI/CD.

ИД-3 ПК-1. Интегрирует инструменты безопасности (SAST, DAST, SCA) в конвейеры CI/CD для автоматического сканирования уязвимостей на ранних этапах разработки.	Не знает инструментов SAST/DAST, не понимает различий между ними, не может запустить сканирование и интерпретировать отчёт.	Запускает SAST (Bandit/Semgrep) в базовом режиме, получает отчёт, но не классифицирует уязвимости. Quality Gate по безопасности не настраивает. DAST не применяет.	Настраивает SAST (Semgrep/Bandit) и DAST (OWASP ZAP Baseline) в CI/CD. Классифицирует уязвимости по CWE и CVSS. Настраивает Quality Gate (пайплайн падает при HIGH-уязвимостях). Исправляет типовые уязвимости.	Настраивает полный security-пайплайн (SAST + DAST Full Scan) с аутентификацией в CI/CD. Пишет собственные правила для Semgrep. Проводит сравнительный анализ SAST и DAST, формулирует выводы о комплементарности методов. Исправляет все выявленные уязвимости, включая специфические (SSTI, XXE, prototype pollution).
ИД-7 ПК-1. Оценивает эффективность работы CI/CD-конвейеров и оптимизирует их для сокращения времени цикла разработки и повышения надежности развертываний.	Не понимает метрик качества тестирования, не может настроить Quality Gates. Отчёты о тестировании не публикует.	Настраивает порог покрытия кода (70%) и публикацию отчётов как артефактов. Quality Gates применяет только для одного этапа. Время выполнения пайплайна не оптимизирует.	Настраивает Quality Gates для 3+ этапов (покрытие, SAST, нагрузочное тестирование). Публикует отчёты в JUnit и HTML. Оптимизирует пайплайн через кэширование и параллелизацию. Анализирует flaky-тесты.	Настраивает комплексные Quality Gates (покрытие $\geq 85\%$, P95 < SLA, отсутствие HIGH-уязвимостей) с автоматической блокировкой пайплайна. Внедряет параллельный запуск и шардирование тестов. Оценивает эффективность пайплайна по метрикам DORA. Разрабатывает меры по устранению flaky-тестов и повышению стабильности.

ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОВЕРКИ УРОВНЯ СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ

Номер задания	Правильный ответ	Содержание вопроса	Компетенция
2 семестр			
1	В	Какой из перечисленных видов тестирования относится к нефункциональному? А) Модульное тестирование В) Нагрузочное тестирование С) Интеграционное тестирование Д) Регрессионное тестирование	ПК-1
2	С	Какой паттерн проектирования UI-автотестов предполагает инкапсуляцию локаторов и действий страницы в отдельном классе? А) Singleton В) Factory С) Page Object Model Д) Observer	ПК-1
3	В	Какой инструмент используется для статического анализа безопасности кода на Python? А) OWASP ZAP В) Bandit С) k6 Д) Selenium	ПК-1
4	Д	Что означает аббревиатура SLO в контексте Site Reliability Engineering? А) Service Level Objective	ПК-1

Номер задания	Правильный ответ	Содержание вопроса	Компетенция
2 семестр			
		B) Service Level Indicator C) Service Level Agreement D) Service Level Object	
5	A	Какой HTTP-метод в REST API должен быть идемпотентным? A) GET B) POST C) PATCH D) все перечисленные	ПК-1
6	C	Какая команда pytest используется для измерения покрытия кода с минимальным порогом 80%? A) pytest --cov=src B) pytest --cov-report=html C) pytest --cov=src --cov-fail-under=80 D) pytest --cov-fail-under=80	ПК-1
7	B	Какой тип нагрузочного тестирования предполагает резкий скачок количества виртуальных пользователей за короткое время? A) Stress test B) Spike test C) Soak test D) Smoke test	ПК-1

Номер задания	Правильный ответ	Содержание вопроса	Компетенция
2 семестр			
8	D	Какой инструмент позволяет поднимать реальные зависимости (например, PostgreSQL, RabbitMQ) в Docker непосредственно из кода тестов? A) Docker Compose B) Podman C) Kubernetes D) Testcontainers	ПК-1
9	A	Что такое Dead Letter Queue (DLQ) в брокерах сообщений? A) Очередь для сообщений, которые не удалось обработать после всех попыток B) Очередь с самым высоким приоритетом C) Очередь для сообщений с истекшим TTL D) Техническая очередь для служебных сообщений	ПК-1
10	C	Какой формат отчёта о тестировании нативно поддерживается GitLab CI для визуализации результатов во вкладке Tests? A) HTML B) JSON C) JUnit XML D) CSV	ПК-1
11	B	Что из перечисленного НЕ может обнаружить статический анализатор безопасности (SAST)? A) Hardcoded secrets	ПК-1

Номер задания	Правильный ответ	Содержание вопроса	Компетенция
2 семестр			
		В) XSS уязвимость, возникающую при выводе данных в браузер С) SQL-инъекцию в строке запроса D) Использование eval() с пользовательским вводом	
12	A	Какой инструмент используется для динамического анализа безопасности веб-приложений (DAST)? A) OWASP ZAP B) Semgrep C) Bandit D) SonarQube	ПК-1
13	D	Какая метрика производительности показывает время, за которое обрабатываются 95% самых медленных запросов? A) Среднее арифметическое B) P50 C) P99 D) P95	ПК-1
14	B	Какой из перечисленных уровней тестирования является самым быстрым и должно быть больше всего в пирамиде тестирования? A) Интеграционное тестирование B) Модульное тестирование	ПК-1

Номер задания	Правильный ответ	Содержание вопроса	Компетенция
2 семестр			
		С) E2E тестирование D) Системное тестирование	
15	С	Что означает принцип FIRST в модульном тестировании, где буква "I" означает Isolated? А) Тесты должны быть интегрированы с БД В) Тесты должны запускаться изолированно от CI/CD С) Тесты не должны зависеть друг от друга и от внешних ресурсов D) Каждый тест должен иметь уникальное имя	ПК-1
16	А	Какой тип уязвимости согласно OWASP Top 10 лучше всего обнаруживается DAST-инструментами? А) XSS (Cross-Site Scripting) В) Hardcoded credentials С) Использование устаревших криптоалгоритмов D) SQL-инъекция в коде (без выполнения)	ПК-1
17	D	Как в к6 называется механизм, позволяющий автоматически провалить тест, если метрика не соответствует заданному порогу? А) Check В) Group С) Scenario D) Threshold	ПК-1

Номер задания	Правильный ответ	Содержание вопроса	Компетенция
2 семестр			
18	В	Какой инструмент можно использовать для запуска коллекции Postman в CI/CD-пайплайне без графического интерфейса? A) Postman Desktop B) Newman C) Postman CLI D) Postman Agent	ПК-1
19	С	Что такое flaky test? A) Тест, который всегда падает B) Тест, который всегда проходит C) Тест, который может случайно падать или проходить без изменения кода D) Тест, который требует ручной проверки	ПК-1
20	В	Какой формат отчёта о результатах SAST рекомендуется для загрузки в GitHub Security Dashboard? A) JUnit XML B) SARIF C) HTML D) CSV	ПК-1
21	А	Какая стратегия развертывания предполагает постепенное направление небольшого процента трафика на новую версию приложения?	ПК-1

Номер задания	Правильный ответ	Содержание вопроса	Компетенция
2 семестр			
		A) Canary deployment B) Blue-green deployment C) Rolling update D) Recreate	
22	C	Какая библиотека Python используется для создания мок-объектов в модульных тестах? A) pytest-cov B) coverage C) unittest.mock D) pytest-xdist	ПК-1
23	D	Какая модель нагрузки в k6 предполагает фиксированное количество виртуальных пользователей? A) Open model B) Constant arrival rate C) Ramping arrival rate D) Closed model	ПК-1
24	B	Для чего используется аннотация <code>@pytest.mark.parametrize</code> ? A) Для пропуска теста B) Для параметризации теста с несколькими наборами данных C) Для маркировки теста как ожидаемо падающего	ПК-1

Номер задания	Правильный ответ	Содержание вопроса	Компетенция
2 семестр			
		D) Для настройки фикстуры	
25	A	Какой статус-код HTTP возвращает сервер при успешном создании ресурса (POST)? A) 201 Created B) 200 OK C) 202 Accepted D) 204 No Content	ПК-1
26	C	Что такое Quality Gate в CI/CD? A) Ворота для доступа к production-окружению B) Метрика качества кода C) Автоматическая проверка, блокирующая пайплайн при недостижении пороговых значений метрик D) Инструмент для сбора отчётов о тестировании	ПК-1
27	B	Какой инструмент написан на Go и использует JavaScript для написания сценариев нагрузочного тестирования? A) JMeter B) k6 C) Locust D) Gatling	ПК-1

Номер задания	Правильный ответ	Содержание вопроса	Компетенция
2 семестр			
28	D	Какая метрика DevOps (DORA) измеряет время от коммита до успешного развертывания в production? A) Deployment Frequency B) Mean Time to Recovery C) Change Failure Rate D) Lead Time for Changes	ПК-1
29	C	Что делает метод <code>page.route()</code> в Playwright? A) Переходит на указанный URL B) Возвращает текущий URL страницы C) Позволяет перехватывать и модифицировать сетевые запросы D) Очищает маршрутизацию между страницами	ПК-1
30	A	Какая категория уязвимостей CWE соответствует проблеме хардкода паролей в исходном коде? A) CWE-798 B) CWE-89 C) CWE-79 D) CWE-502	ПК-1
31	B	Что означает "blameless postmortem" в SRE-практиках? A) Анализ инцидента с поиском виновного B) Анализ инцидента без обвинений, с фокусом на системные причины	ПК-1

Номер задания	Правильный ответ	Содержание вопроса	Компетенция
2 семестр			
		С) Документ, который никто не читает D) Отчёт о плановых работах	
32	D	Какой протокол использует RabbitMQ по умолчанию? A) MQTT B) STOMP C) HTTP D) AMQP 0-9-1	ПК-1
33	C	Что делает метод basic_nack в RabbitMQ с параметром requeue=False? A) Подтверждает успешную обработку сообщения B) Отклоняет сообщение и возвращает его в очередь для повторной обработки C) Отклоняет сообщение и отправляет его в Dead Letter Queue (не возвращает в очередь) D) Отклоняет все сообщения в очереди	ПК-1
34	A	Какой плагин Kong Gateway используется для аутентификации с помощью JWT? A) jwt B) key-auth C) oauth2 D) ldap-auth	ПК-1
35	B	Какая команда Git позволяет просмотреть историю коммитов в виде графика?	ПК-1

Номер задания	Правильный ответ	Содержание вопроса	Компетенция
2 семестр			
		A) git log --oneline B) git log --graph --oneline --decorate C) git status D) git diff	
36	C	Для чего используется функция pytest.fixture с параметром scope="session"? A) Фикстура создаётся для каждого теста B) Фикстура создаётся для каждого модуля C) Фикстура создаётся один раз на всю сессию тестирования D) Фикстура не создаётся вообще	ПК-1
37	D	Какая метрика показывает долю успешных запросов к сервису? A) SLI B) SLO C) Error Budget D) Success Rate (Availability)	ПК-1
38	A	Какой тип тестирования проверяет корректность взаимодействия нескольких модулей между собой? A) Интеграционное тестирование B) Модульное тестирование C) Приемочное тестирование	ПК-1

Номер задания	Правильный ответ	Содержание вопроса	Компетенция
2 семестр			
		D) Дымовое тестирование	
39	B	Какой инструмент используется для кэширования Docker-слоёв в GitLab CI? A) actions/cache B) cache: ключ с путями к директориям Docker C) artifacts: D) needs:	ПК-1
40	C	Что такое Error Budget в SRE? A) Бюджет на исправление ошибок B) Количество багов в релизе C) Допустимый уровень недоступности сервиса = 100% - SLO D) Штраф за нарушение SLA	ПК-1
41	A	Какой метод HTTP используется для частичного обновления ресурса в REST API? A) PATCH B) PUT C) POST D) UPDATE	ПК-1
42	D	Что такое "mock" в контексте модульного тестирования? A) Реальный объект, используемый в тесте	ПК-1

Номер задания	Правильный ответ	Содержание вопроса	Компетенция
2 семестр			
		В) Набор тестовых данных С) Временная папка для файлов D) Объект-заменитель, имитирующий поведение реального объекта	
43	В	Какая стратегия обновления приложений предполагает одновременное существование двух идентичных окружений (текущего и нового) с переключением трафика? А) Canary В) Blue-Green С) Rolling update D) Recreate	ПК-1
44	С	Какой из перечисленных инструментов предназначен для анализа зависимостей на наличие известных уязвимостей (SCA)? А) Semgrep В) OWASP ZAP С) OWASP Dependency-Check D) Bandit	ПК-1
45	А	Какой метод в библиотеке requests в Python используется для отправки POST-запроса? А) requests.post(url, json=data) В) requests.get(url, params=data) С) requests.put(url, data=data)	ПК-1

Номер задания	Правильный ответ	Содержание вопроса	Компетенция
2 семестр			
		D) requests.delete(url)	
46	B	Что такое "shift left" в контексте безопасности? A) Перенос тестирования безопасности на этап эксплуатации B) Перенос проверок безопасности на максимально ранние этапы разработки C) Сдвиг влево по временной шкале релиза D) Переход на левую клавиатуру	ПК-1
47	D	Какой из перечисленных форматов сериализации является бинарным и используется в gRPC? A) JSON B) YAML C) XML D) Protocol Buffers (protobuf)	ПК-1
48	C	Какая команда в k6 запускает тест с указанием количества виртуальных пользователей и длительности? A) k6 run --vus 10 --duration 30s script.js B) k6 test --users 10 --time 30s script.js C) k6 run --vus 10 --duration 30s script.js D) k6 execute -v 10 -d 30 script.js	ПК-1
49	B	Что представляет собой SARIF (Static Analysis Results Interchange Format)?	ПК-1

Номер задания	Правильный ответ	Содержание вопроса	Компетенция
2 семестр			
		A) Формат отчёта о покрытии кода B) Стандартный формат обмена результатами статического анализа C) Формат логов выполнения тестов D) Формат конфигурации CI/CD	
50	A	Какой принцип FIRST в модульном тестировании означает, что тест должен однозначно сообщать об успехе или провале без ручной интерпретации? A) Self-validating B) Fast C) Repeatable D) Timely	ПК-1

Описание шкалы оценивания

В рамках рейтинговой системы успеваемость студентов по каждой дисциплине оценивается в ходе текущего контроля и промежуточной аттестации. Рейтинговая система оценки знаний студентов основана на использовании совокупности контрольных мероприятий по проверке пройденного материала, оптимально расположенных на всем временном интервале изучения дисциплины. Принципы рейтинговой системы оценки знаний студентов основываются на требованиях, описанных в Положении об организации образовательного процесса на основе рейтинговой системы оценки знаний студентов в ФГАОУ ВО «СКФУ».

Критерии оценивания компетенций

1. **Полно и глубоко** демонстрирует знание теоретических основ автоматизации тестирования, включая пирамиду тестирования, классификацию тестов, принципы FIRST, метрики покрытия кода и методы оценки качества ПО.
2. **Способен самостоятельно проектировать** архитектуру CI/CD-пайплайна, обоснованно определяя состав и последовательность этапов тестирования (модульные, интеграционные, E2E, нагрузочные, безопасности) в зависимости от типа приложения и требований к качеству.
3. **В совершенстве владеет** инструментарием автоматизации тестирования: pytest (включая фикстуры, параметризацию, моки), Playwright (Page Object Model), k6 (написание сценариев, thresholds), Postman/Newman, Testcontainers.
4. **Успешно интегрирует** инструменты безопасности (SAST: Semgrep/Bandit; DAST: OWASP ZAP) в CI/CD-пайплайн, настраивает Quality Gates по безопасности (блокировка при HIGH/CRITICAL уязвимостях), классифицирует уязвимости по CWE и CVSS.
5. **Самостоятельно разрабатывает и отлаживает** все виды тестов (модульные, интеграционные, UI, нагрузочные) с достижением целевых метрик: покрытие кода $\geq 85\%$, P95 latency $\leq$ SLA, отсутствие критических уязвимостей.
6. **Настраивает комплексные Quality Gates** в CI/CD (покрытие кода, успешность тестов, производительность, безопасность) с автоматической блокировкой пайплайна при нарушении порогов; публикует отчёты в форматах JUnit XML, Cobertura XML, SARIF.
7. **Анализирует и оптимизирует** время выполнения пайплайна за счёт параллелизации, кэширования и условного выполнения этапов; оценивает эффективность тестирования по метрикам DORA.
8. **Проводит сравнительный анализ** SAST и DAST, формулирует обоснованные выводы о комплементарности методов для целостной стратегии DevSecOps.
9. **Выполняет сквозной проект** в полном объёме (14+ этапов пайплайна), оформляет отчёт в соответствии с требованиями, уверенно отвечает на дополнительные вопросы при защите.

Оценка «хорошо» выставляется студенту, если он:

1. **Демонстрирует хорошее знание** теоретического материала, но допускает незначительные неточности (например, при классификации видов тестирования или метрик).
2. **Способен спроектировать** пайплайн из 4-5 этапов тестирования, но обоснование выбора стратегии может быть неполным.
3. **Владеет основными инструментами** автоматизации тестирования, но испытывает небольшие затруднения при написании сложных сценариев (параметризация, моки, кастомные правила Semgrep).
4. **Интегрирует SAST и DAST** в пайплайн, но Quality Gate по безопасности может быть настроен не для всех типов уязвимостей.
5. **Разрабатывает тесты** с достижением покрытия 70-80%, однако не все тесты параметризованы, либо отсутствует один из видов тестирования (например, нагрузочные или UI-тесты).
6. **Настраивает Quality Gates** для 3+ этапов, но допускает неоптимальные пороговые значения или не публикует все форматы отчётов.
7. **Оптимизирует пайплайн** (кэширование, параллелизация), но не проводит анализа метрик DORA.
8. **Сравнивает SAST и DAST** на базовом уровне, но выводы могут быть неполными.
9. **Выполняет сквозной проект** с небольшими недочётами (11-13 этапов из 18), отчёт оформлен хорошо, но есть незначительные замечания.

Оценка «удовлетворительно» выставляется студенту, если он:

1. **Демонстрирует фрагментарные знания** теоретического материала, путает виды тестирования или метрики, но способен воспроизвести основные определения с помощью наводящих вопросов.
2. **Способен спроектировать базовый пайплайн** из 2-3 этапов (сборка → модульные тесты), но допускает ошибки в последовательности этапов.
3. **Имеет базовые навыки работы с инструментами:** может написать простые модульные тесты, но не использует фикстуры, параметризацию или моки.
4. **Запускает SAST-инструмент** в базовом режиме, получает отчёт, но не классифицирует уязвимости и не настраивает Quality Gate по безопасности. DAST не применяет или применяет формально.
5. **Разрабатывает тесты** с покрытием 60-70%, но тесты могут быть нестабильны (flaky) или отсутствуют некоторые виды тестирования.

6. **Настраивает порог покрытия кода**, но Quality Gates для производительности и безопасности не настроены.
7. **Выполняет минимальные действия по оптимизации** (только кэширование), не применяет параллелизацию и шардирование.
8. **Выполняет сквозной проект** в минимальном объёме (8-10 этапов из 18), отчёт оформлен с нарушениями требований.

Оценка «неудовлетворительно» (2)

Оценка «неудовлетворительно» выставляется студенту, если он:

1. **Не демонстрирует знаний** теоретического материала, не может назвать виды тестирования, метрики покрытия или инструменты автоматизации.
2. **Не способен спроектировать** даже базовый пайплайн, не понимает места тестирования в CI/CD.
3. **Не владеет инструментарием** автоматизации тестирования: не может написать простой модульный тест, не запускал pytest, Postman или k6.
4. **Не знаком** с инструментами безопасности (SAST/DAST) и не может их применить.
5. **Не выполняет тесты** или все тесты падают, покрытие кода отсутствует или менее 60%.
6. **Не настраивает Quality Gates** в пайплайне.
7. **Не выполняет сквозной проект** или выполняет менее 8 этапов, отчёт отсутствует или не соответствует требованиям.
8. **Допускает грубые ошибки** при ответах на контрольные вопросы и не исправляет их даже с помощью преподавателя.

Шкала перевода баллов в оценку (сводная)

Оценка	Баллы (зачётная единица)	Процент выполнения заданий	Характеристика уровня сформированности компетенции
5 (отлично)	4,8 – 5,0	90% – 100%	Высокий уровень: самостоятельное, полное и глубокое освоение всех индикаторов компетенции ПК-1

Оценка	Баллы (зачётная единица)	Процент выполнения заданий	Характеристика уровня сформированности компетенции
4 (хорошо)	3,8 – 4,7	75% – 89%	Средний уровень: хорошее освоение с незначительными недочётами
3 (удовлетворительно)	2,8 – 3,7	60% – 74%	Минимальный уровень: базовые знания и умения, требуют наводящих вопросов
2 (неудовлетворительно)	0 – 2,7	0% – 59%	Уровень не достигнут: компетенция не сформирована