

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ



Частное учреждение высшего образования
«Высшая школа предпринимательства (институт)»
(ЧУВО «ВШП»)

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
Б1.В.09 «Тестирование программного обеспечения»

Направление подготовки: 09.03.02 Информационные системы и технологии

Направленность (профиль) программы бакалавриата
«Информационные системы в экономике»

ОДОБРЕНО

Ученым советом ЧУВО «ВШП»

Протокол заседания

№01-02/24 от 15 мая 2024 г.



15 мая 2024 г.

Аллабян М.Г.

	Документ подписан электронной цифровой подписью VSHP EDS GEN 1, уникальный ключ документа:
	8F30 - 29EE - EB2F - GNI5
	Организация: ЧУВО «ВШП», ИНН: 6903013604
	Дата подписания: 15.05.2024
	Подписал: Аллабян М. Г.

Тверь, 2024

Рабочая программа учебной дисциплины **Б1.В.09 Тестирование программного обеспечения**, компонента основной профессиональной образовательной программы высшего образования — программы бакалавриата по направлению подготовки **09.03.02 Информационные системы и технологии** направленность (профиль) **«Информационные системы в экономике»**, направлена на обеспечение у обучающегося способности осуществлять профессиональную деятельность в соответствующей области и сферах профессиональной деятельности, в том числе на их практическую подготовку с учётом рабочей программы воспитания и календарного плана воспитательной работы Частном учреждении высшего образования **«Высшая школа предпринимательства (институт)»** (далее — **ЧУВО «ВШП»**).

1. ОБЛАСТЬ ПРИМЕНЕНИЯ И НОРМАТИВНЫЕ ССЫЛКИ

Настоящая рабочая программа учебной дисциплины устанавливает требования к результатам обучения студента и определяет содержание и виды учебных занятий и отчетности.

Программа предназначена для преподавателей и студентов направления подготовки 09.03.02 Информационные системы и технологии.

Программа учебной дисциплины разработана в соответствии с ФГОС ВО, утвержденного приказом Минобрнауки России от 19.09.2017 № 926 «Об утверждении федерального государственного образовательного стандарта высшего образования - бакалавриата по направлению подготовки 09.03.02 Информационные системы и технологии», основной профессиональной образовательной программой высшего образования по направлению подготовки 09.03.02 Информационные системы и технологии, направленность (профиль) Информационные системы в экономике.

2. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ

Целью изучения дисциплины «Тестирование программного обеспечения» является обучение студентов основам тестирования программных продуктов; приобретение навыков самостоятельного создания тестов.

Основными задачами изучения дисциплины являются изучение моделей жизненного цикла программного обеспечения, жизненного цикла дефекта; приобретение навыков анализировать тестовые случаи, создавать тестовые задания, составление отчетов по тестированию программного обеспечения.

3. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП ВО

Дисциплина является компонентом части, формируемой участниками образовательных отношений Блока 1 основной профессиональной образовательной программы высшего образования по направлению подготовки 09.03.02 Информационные системы и технологии, направленность (профиль) — Информационные системы в экономике.

4. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ В РАМКАХ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОСВОЕНИЯ ОСНОВНОЙ ПРОФЕССИОНАЛЬНОЙ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Процесс освоения учебной дисциплины направлен на формирование у обучающихся следующих компетенций.

В результате освоения учебной дисциплины обучающийся должен демонстрировать следующие результаты обучения: УК-2, УК-3, ПК-2, ПК-3, ПК-4.

Таблица 1. Результаты обучения

Код компетенции	Наименование компетенции	Индекс и наименование индикатора содержания компетенции	Дескрипторы – основные признаки освоения (показатели достижения результата)
УК-2	Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	УК-2.1 Способен определять круг задач в рамках поставленной цели	Знать: - Методы анализа и декомпозиции задач в программировании. - Основные этапы разработки программного обеспечения. Уметь: - Определять задачи, вытекающие из поставленной цели, и разрабатывать план их выполнения. - Структурировать задачи и определять их приоритетность. Владеть: - Навыками анализа и декомпозиции задач.

			<ul style="list-style-type: none"> - Методами планирования и структурирования задач в программировании.
		<p>УК-2.2 Способен выбирать оптимальные способы решения задач, исходя из правовых норм, ресурсов и ограничений</p>	<p>Знать:</p> <ul style="list-style-type: none"> - Правовые нормы и ограничения в области программирования. - Методы оценки ресурсов и ограничений при разработке программного обеспечения. <p>Уметь:</p> <ul style="list-style-type: none"> - Выбирать оптимальные методы и инструменты для решения задач в программировании. - Обосновывать выбор методов и инструментов с учетом правовых норм, ресурсов и ограничений. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками оценки ресурсов и ограничений при выборе способов решения задач. - Методами обоснования выбора оптимальных решений в программировании.
УК-3	Способен осуществлять социальное взаимодействие и реализовывать свою роль в команде	<p>УК-3.1 Способен осуществлять социальное взаимодействие в команде</p>	<p>Знать:</p> <ul style="list-style-type: none"> - Принципы и методы командной работы в разработке программного обеспечения. - Основы делового общения и взаимодействия в команде. <p>Уметь:</p> <ul style="list-style-type: none"> - Эффективно взаимодействовать с членами команды при разработке программного обеспечения. - Принимать участие в обсуждениях и совместных проектах. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками командного взаимодействия в процессе разработки программного обеспечения. - Методами делового общения и взаимодействия в команде.
		<p>УК-3.2 Способен реализовывать свою роль в команде</p>	<p>Знать:</p> <ul style="list-style-type: none"> - Различные роли и функции в команде разработки программного обеспечения. - Методы распределения обязанностей в команде. <p>Уметь:</p> <ul style="list-style-type: none"> - Определять и выполнять свою роль в команде разработки программного обеспечения. - Принимать ответственность за выполнение задач в команде. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками самоуправления и координации в рамках команды разработки. - Методами эффективного выполнения своей роли в команде.

ПК-2	Проверка работоспособности и рефакторинг кода программного обеспечения	ПК-2.1 Способен проверять работоспособность программного кода	<p>Знать:</p> <ul style="list-style-type: none"> - Принципы и методы тестирования программного кода. - Основы юнит-тестирования и функционального тестирования. <p>Уметь:</p> <ul style="list-style-type: none"> - Разрабатывать и выполнять тестовые сценарии для проверки работоспособности кода. - Использовать тестовые фреймворки и инструменты для автоматизированного тестирования. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками написания и выполнения тестов. - Навыками анализа результатов тестирования и устранения обнаруженных проблем.
		ПК-2.2 Способен рефакторить программный код	<p>Знать:</p> <ul style="list-style-type: none"> - Принципы и методы рефакторинга программного кода. - Паттерны и антипаттерны проектирования кода. <p>Уметь:</p> <ul style="list-style-type: none"> - Проводить анализ и улучшение структуры кода. - Вносить изменения в код, сохраняя его функциональность. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками применения рефакторинга для улучшения читаемости и поддерживаемости кода. - Навыками использования инструментов для автоматизированного рефакторинга.
ПК-3	Интеграция программных модулей и компонентов и проверка работоспособности выпусков программного продукта	ПК-3.1 Способен интегрировать программные модули и компоненты	<p>Знать:</p> <ul style="list-style-type: none"> - Принципы и методы интеграции программных модулей. - Основы межмодульного взаимодействия и совместимости. <p>Уметь:</p> <ul style="list-style-type: none"> - Планировать и проводить интеграцию модулей в общий проект. - Решать проблемы, возникающие при интеграции различных компонентов. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками использования инструментов и платформ для интеграции. - Навыками написания интеграционных тестов.
		ПК-3.2 Способен проверять работоспособность выпусков программного продукта	<p>Знать:</p> <ul style="list-style-type: none"> - Методы и инструменты для проверки выпусков программного продукта. - Основы тестирования и валидации программного обеспечения. <p>Уметь:</p> <ul style="list-style-type: none"> - Проводить проверку и валидацию программного продукта.

			<ul style="list-style-type: none"> - Анализировать результаты тестирования и устранять выявленные дефекты. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками работы с инструментами для проверки и тестирования выпусков. - Навыками документирования результатов тестирования и предложений по улучшению.
ПК-4	Разработка требований и проектирование программного обеспечения	ПК-4.1 Способен разрабатывать требования к программному обеспечению	<p>Знать:</p> <ul style="list-style-type: none"> - Методы и техники сбора и анализа требований. - Принципы написания требований и спецификаций. <p>Уметь:</p> <ul style="list-style-type: none"> - Собирать и документировать требования к программному обеспечению. - Анализировать и уточнять требования с заинтересованными сторонами. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками написания четких и измеримых требований. - Навыками работы с инструментами для управления требованиями.
		ПК-4.2 Способен проектировать программное обеспечение	<p>Знать:</p> <ul style="list-style-type: none"> - Принципы и методы проектирования программного обеспечения. - Основы архитектурных шаблонов и стилей. <p>Уметь:</p> <ul style="list-style-type: none"> - Разрабатывать архитектуру и дизайн программного обеспечения. - Создавать диаграммы и модели для визуализации архитектуры. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками применения архитектурных паттернов. - Навыками использования инструментов для проектирования и моделирования программного обеспечения.

5. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

5.1. Объем дисциплины в зачетных единицах с указанием количества академических часов

Общая трудоемкость учебной дисциплины составляет 5 зачетных единиц, 180 часов, включая все формы контактной и самостоятельной работы обучающихся.

Объем дисциплины по учебному плану составляет –
5 зачётных единицы = 180 академических часов.

Контактная работа обучающегося (студенты) с научно-педагогическим работником организации (всего) - 62 академических часов,

в том числе:

Лекционные занятия (Лек.) - 20 академических часов,

Практические занятия (Пр.) - 40 академических часов,

Консультации (Конс.) - 2 академических часа.

Самостоятельная работа обучающегося (студента):

Самостоятельная работа (СР) - 82 академический час,

Текущий контроль успеваемости

и промежуточно-заочная аттестация обучающегося (студента):

Часы на контроль - 36 академических часов.

Таблица 2. Объём дисциплины

№ п/п	Раздел/тема дисциплины	Семестр/курс	Виды учебной деятельности, включая самостоятельную работу обучающихся (студентов), и трудоёмкость (в ак. часах)				Коды формируемых компетенций
			Виды учебных занятий по дисциплине			Самостоятельная работа СР	
			Лек.	Пр.	Конс.		
1	Тема 1. Введение в тестирование программного обеспечения	9 семестр/ 5 курс	1	2		6	УК-2.1, ПК-2.1
2	Тема 2. Основы планирования тестирования	9 семестр/ 5 курс	2	4		6	УК-2.2, ПК-4.1
3	Тема 3. Методы тестирования программного обеспечения	9 семестр/ 5 курс	2	2		6	ПК-2.1, ПК-2.2
4	Тема 4. Инструменты автоматизированного тестирования	9 семестр/ 5 курс	2	4		8	ПК-2.2, ПК-3.2
5	Тема 5. Управление тестированием и тестовые данные	9 семестр/ 5 курс	2	2		6	ПК-4.1, ПК-4.2
6	Тема 6. Тестирование производительности и нагрузки	9 семестр/ 5 курс	1	4		6	ПК-3.1, ПК-3.2
7	Тема 7. Тестирование безопасности программного обеспечения	9 семестр/ 5 курс	2	4		8	ПК-3.1, ПК-3.2
8	Тема 8 Управление дефектами и отчетность	9 семестр/ 5 курс	2	4		6	УК-2.1, ПК-2.2
9	Тема 9 Взаимодействие в команде при тестировании	9 семестр/ 5 курс	2	2		6	УК-3.1, УК-3.2
10	Тема 10 Тестирование на финальных стадиях разработки и релизное тестирование	9 семестр/ 5 курс	2	4		8	ПК-3.2, ПК-4.2

11	Тема 11 Современные тренды и будущее тестирования ПО	9 семестр/ 5 курс	2	4		6	УК-1.1, УК-1.2
12	Тема 12 Итоговое занятие и подготовка к экзамену	9 семестр/ 5 курс		4	2	8	УК-2.1, УК-2.2, УК-3.1, УК-3.2, ПК-2.1, ПК-2.2, ПК-3.1, ПК-3.2, ПК-4.1, ПК-4.2
ИТОГО аудиторных часов/СР:		9 семестр/ 5 курс	62 ак. часов			82 ак. часа	-
Часы на контроль		9 семестр/ 5 курс	36 ак. час (форма промежуточной аттестации – экзамен – 9 семестр)				
ВСЕГО ак. часов:		9 семестр/ 5 курс	180 академических часа				

5.2. Тематическое содержание дисциплины

* количество академических часов и виды занятий представлены в таблице № 2.

Тема 1: Введение в тестирование программного обеспечения

Содержание: Основные понятия и цели тестирования. Виды тестирования: функциональное, нефункциональное, регрессионное и т.д. Роль тестирования в жизненном цикле разработки ПО.

Тема 2: Основы планирования тестирования

Содержание: Разработка стратегии тестирования. Создание тестового плана. Определение тестируемых требований и задач. Выбор инструментов и ресурсов для тестирования.

Тема 3: Методы тестирования программного обеспечения

Содержание: Динамическое и статическое тестирование. Тестирование "черного ящика" и "белого ящика". Интеграционное и системное тестирование. Методы автоматизированного тестирования.

Тема 4: Инструменты автоматизированного тестирования

Содержание: Обзор популярных инструментов автоматизированного тестирования (Selenium, JUnit, TestNG и др.). Внедрение автоматизации в процесс тестирования. Создание и выполнение автотестов.

Тема 5: Управление тестированием и тестовые данные

Содержание: Управление тестовыми данными и конфигурациями. Подготовка тестовых сценариев и данных. Методы и инструменты для управления тестовыми данными.

Тема 6: Тестирование производительности и нагрузки

Содержание: Основные концепции и цели тестирования производительности. Инструменты и методы для проведения тестов производительности и нагрузочных тестов. Анализ результатов и оптимизация производительности.

Тема 7: Тестирование безопасности программного обеспечения

Содержание: Методы и техники тестирования безопасности. Обнаружение уязвимостей и угроз. Инструменты для тестирования безопасности (OWASP ZAP, Burp Suite и др.).

Тема 8: Управление дефектами и отчетность

Содержание: Жизненный цикл дефекта. Использование систем управления дефектами (Jira, Bugzilla и др.). Создание и ведение отчетности по результатам тестирования.

Тема 9: Взаимодействие в команде при тестировании

Содержание: Роли и обязанности в команде тестирования. Методы и инструменты для

совместной работы (Trello, Confluence и др.). Коммуникация и управление взаимодействием в команде.

Тема 10: Тестирование на финальных стадиях разработки и релизное тестирование

Содержание: Подготовка к финальному тестированию. Проведение релизного тестирования. Подготовка отчетов и документации для релиза.

Тема 11: Современные тренды и будущее тестирования ПО

Содержание: Последние тенденции в тестировании ПО. Влияние DevOps и CI/CD на процесс тестирования. Будущее автоматизации и искусственного интеллекта в тестировании.

Тема 12: Итоговое занятие и подготовка к экзамену

Содержание: Повторение и обобщение пройденного материала, решение типовых задач и вопросов для подготовки к экзамену. Обсуждение практических аспектов тестирования программного обеспечения.

5.2.1 Содержание практических занятий

Таблица 3

№ п/п	Наименование темы (раздела) дисциплины	Содержание практического занятия
1	Практическое занятие 1: Введение в тестирование программного обеспечения	Задание: Ознакомиться с основными понятиями и целями тестирования. Изучить различные виды тестирования (функциональное, нефункциональное, регрессионное и т.д.). Провести анализ роли тестирования в жизненном цикле разработки ПО на примере реального проекта. Цель: Понять основные концепции и значение тестирования в процессе разработки ПО.
2	Практическое занятие 2: Основы планирования тестирования	Задание: Разработать стратегию тестирования для заданного проекта. Создать тестовый план, определить тестируемые требования и задачи. Подготовить список необходимых инструментов и ресурсов для тестирования. Цель: Научиться планировать процесс тестирования и подбирать необходимые ресурсы.
3	Практическое занятие 3: Методы тестирования программного обеспечения	Задание: Провести анализ методов динамического и статического тестирования. Выполнить тестирование "черного ящика" и "белого ящика" на примерах. Реализовать интеграционное и системное тестирование. Цель: Освоить различные методы тестирования и их применение на практике.
4	Практическое занятие 4: Инструменты автоматизированного тестирования	Задание: Ознакомиться с инструментами автоматизированного тестирования (Selenium, JUnit, TestNG и др.). Установить и настроить выбранные инструменты. Создать и выполнить простые автотесты. Цель: Изучить возможности автоматизированного тестирования и освоить базовые навыки работы с инструментами.
5	Практическое занятие 5: Управление тестированием и тестовые данные	Задание: Разработать методы управления тестовыми данными и конфигурациями. Подготовить тестовые сценарии и данные для заданного проекта. Использовать инструменты для управления тестовыми данными. Цель: Научиться управлять тестовыми данными и конфигурациями, а также готовить тестовые сценарии.
6	Практическое занятие 6: Тестирование производительности и нагрузки	Задание: Изучить основные концепции и цели тестирования производительности. Провести тестирование производительности и нагрузочные тесты с использованием инструментов (например, JMeter). Проанализировать результаты тестов и предложить способы оптимизации производительности. Цель: Освоить методы тестирования производительности и анализа результатов.
7	Практическое занятие 7: Тестирование безопасности программного обеспечения	Задание: Ознакомиться с методами и техниками тестирования безопасности. Использовать инструменты (OWASP ZAP, Burp Suite и др.) для обнаружения уязвимостей. Провести анализ безопасности заданного ПО. Цель: Научиться выявлять уязвимости и обеспечивать безопасность программного обеспечения.

8	Практическое занятие 8: Управление дефектами и отчетность	Задание: Изучить жизненный цикл дефекта и системы управления дефектами (Jira, Bugzilla и др.). Создать и вести отчетность по результатам тестирования. Составить отчет по обнаруженным дефектам и предложить решения. Цель: Освоить управление дефектами и создание отчетности.
9	Практическое занятие 9: Взаимодействие в команде при тестировании	Задание: Ознакомиться с ролями и обязанностями в команде тестирования. Использовать инструменты для совместной работы (Trello, Confluence и др.). Организовать коммуникацию и управление взаимодействием в команде тестирования. Цель: Научиться эффективно работать в команде и взаимодействовать с коллегами.
10	Практическое занятие 10: Тестирование на финальных стадиях разработки и релизное тестирование	Задание: Подготовиться к финальному тестированию. Провести релизное тестирование для заданного проекта. Составить отчет и документацию для релиза. Цель: Освоить методы финального тестирования и подготовки к релизу.
11	Практическое занятие 11: Современные тренды и будущее тестирования ПО	Задание: Изучить последние тенденции в тестировании ПО. Проанализировать влияние DevOps и CI/CD на процесс тестирования. Обсудить будущее автоматизации и искусственного интеллекта в тестировании. Цель: Ознакомиться с современными тенденциями и перспективами развития тестирования.
12	Практическое занятие 12: Итоговое занятие и подготовка к экзамену	Задание: Повторить и обобщить пройденный материал. Решить типовые задачи и вопросы для подготовки к экзамену. Обсудить практические аспекты тестирования программного обеспечения. Цель: Повторить и обобщить изученный материал для успешной сдачи экзамена.

5.2.2 Содержание самостоятельной работы

Таблица 4

№ п/п	Наименование темы (раздела) дисциплины	Содержание самостоятельной работы	Форма контроля
1	Самостоятельная работа 1: Введение в тестирование программного обеспечения	Задание: Прочитать рекомендованные материалы по основным понятиям и целям тестирования, подготовить краткий обзор. Цель: Понять роль тестирования в жизненном цикле разработки ПО и основные виды тестирования.	Ответы на контрольные вопросы.
2	Самостоятельная работа 2: Основы планирования тестирования	Задание: Изучить материалы по разработке стратегии тестирования, созданию тестового плана и определению тестируемых требований и задач. Выполнить упражнения по разработке тестового плана для заданного проекта. Цель: Освоить основы планирования процесса тестирования.	Ответы на контрольные вопросы, выполнение практического задания.
3	Самостоятельная работа 3: Методы тестирования программного обеспечения	Задание: Прочитать материалы по динамическому и статическому тестированию, методам "черного ящика" и "белого ящика". Выполнить упражнения по применению этих методов на примерах. Цель: Научиться применять различные методы тестирования на практике.	Выполнение практического задания.
4	Самостоятельная работа 4: Инструменты автоматизированного тестирования	Задание: Изучить материалы по инструментам автоматизированного тестирования (Selenium, JUnit, TestNG и др.), создать и выполнить автотесты. Цель: Освоить использование инструментов автоматизированного тестирования.	Выполнение практического задания.
5	Самостоятельная работа 5: Управление тестированием и тестовые данные	Задание: Прочитать материалы по управлению тестовыми данными и конфигурациями, подготовке тестовых сценариев и данных. Выполнить упражнения по управлению тестовыми данными. Цель: Научиться управлять тестовыми данными и подготовкой тестовых сценариев.	Выполнение практического задания.
6	Самостоятельная работа 6:	Задание: Изучить материалы по тестированию производительности и нагрузочным тестам. Выполнить	Выполнение практического

	Тестирование производительности и нагрузки	упражнения по проведению тестов производительности и анализу результатов. Цель: Освоить методы тестирования производительности и анализа результатов.	задания.
7	Самостоятельная работа 7: Тестирование безопасности программного обеспечения	Задание: Прочитать материалы по методам и техникам тестирования безопасности. Выполнить упражнения по обнаружению уязвимостей с использованием инструментов (OWASP ZAP, Burp Suite и др.). Цель: Научиться выявлять уязвимости и обеспечивать безопасность ПО.	Выполнение практического задания.
8	Самостоятельная работа 8: Управление дефектами и отчетность	Задание: Изучить материалы по жизненному циклу дефекта и системам управления дефектами (Jira, Bugzilla и др.). Создать отчет по результатам тестирования, описывающий обнаруженные дефекты. Цель: Освоить управление дефектами и создание отчетности.	Выполнение практического задания.
9	Самостоятельная работа 9: Взаимодействие в команде при тестировании	Задание: Прочитать материалы по ролям и обязанностям в команде тестирования, методам и инструментам для совместной работы (Trello, Confluence и др.). Выполнить упражнения по организации взаимодействия в команде. Цель: Научиться эффективно работать в команде и взаимодействовать с коллегами.	Выполнение практического задания.
10	Самостоятельная работа 10: Тестирование на финальных стадиях разработки и релизное тестирование	Задание: Изучить материалы по подготовке к финальному тестированию и проведению релизного тестирования. Подготовить отчет и документацию для релиза. Цель: Освоить методы финального тестирования и подготовки к релизу.	Выполнение практического задания.
11	Самостоятельная работа 11: Современные тренды и будущее тестирования ПО	Задание: Прочитать материалы по последним тенденциям в тестировании ПО, влиянию DevOps и CI/CD на процесс тестирования. Подготовить доклад о будущем автоматизации и искусственного интеллекта в тестировании. Цель: Ознакомиться с современными тенденциями и перспективами развития тестирования.	Тест.
12	Самостоятельная работа 12: Итоговое занятие и подготовка к экзамену	Задание: Повторить пройденный материал, решить типовые задачи и вопросы для подготовки к экзамену. Обсудить практические аспекты тестирования программного обеспечения. Цель: Повторение и обобщение изученного материала для успешной сдачи экзамена.	Экзамен.

6. Оценочные материалы по дисциплине

Оценочные материалы находятся в документе «Оценочные материалы по дисциплине «Тестирование программного обеспечения».

7. Методические материалы для обучающихся по освоению дисциплины (модуля)

А) Рекомендации обучающемуся (студенту) по работе с конспектом после лекции

Какими бы замечательными качествами в области методики ни обладал лектор, какое бы большое значение на занятиях ни уделял лекции слушатель, глубокое понимание материала достигается только путем самостоятельной работы над ним. Самостоятельную работу следует начинать с доработки конспекта, желательно в тот же день, пока время не стерло содержание лекции из памяти (через 10 часов после лекции в памяти остается не более 30-40 % материала). С целью доработки необходимо в первую очередь прочитать записи, восстановить текст в памяти, а также исправить описки, расшифровать не принятые ранее сокращения, заполнить пропущенные места, понять текст, вникнуть в его смысл. Далее прочитать материал по рекомендуемой литературе, разрешая в ходе чтения, возникшие ранее затруднения, вопросы, а также дополнения и исправляя свои записи. Записи должны быть наглядными, для чего следует

применять различные способы выделений. В ходе доработки конспекта углубляются, расширяются и закрепляются знания, а также дополняется, исправляется и совершенствуется конспект. Подготовленный конспект и рекомендуемая литература используется при подготовке к практическому занятию. Подготовка сводится к внимательному прочтению учебного материала, к выводу с карандашом в руках всех утверждений и формул, к решению примеров, задач, к ответам на вопросы, предложенные в конце лекции преподавателем или помещенные в рекомендуемой литературе. Примеры, задачи, вопросы по теме являются материалом самоконтроля. Непременным условием глубокого усвоения учебного материала является знание основ, на которых строится изложение материала. Обычно преподаватель напоминает, какой ранее изученный материал и в какой степени требуется подготовить к очередному занятию. Эта рекомендация, как и требование систематической и серьезной работы над всем лекционным курсом, подлежит безусловному выполнению. Потери логической связи как внутри темы, так и между ними приводит к негативным последствиям: материал учебной дисциплины перестает основательно восприниматься, а творческий труд подменяется утомленным переписыванием. Обращение к ранее изученному материалу не только помогает восстановить в памяти известные положения, выводы, но и приводит разрозненные знания в систему, углубляет и расширяет их. Каждый возврат к старому материалу позволяет найти в нем что-то новое, переосмыслить его с иных позиций, определить для него наиболее подходящее место в уже имеющейся системе знаний. Неоднократное обращение к пройденному материалу является наиболее рациональной формой приобретения и закрепления знаний. Очень полезным, но, к сожалению, еще мало используемым в практике самостоятельной работы, является предварительное ознакомление с учебным материалом. Даже краткое, беглое знакомство с материалом очередной лекции дает многое. Обучающиеся (студенты) получают общее представление о её содержании и структуре, о главных и второстепенных вопросах, о терминах и определениях. Все это облегчает работу на лекции и делает ее целеустремленной.

Б) Рекомендации обучающемуся (студенту) по подготовке к занятиям семинарского типа

Обучающийся (студент) должен чётко уяснить, что именно с лекции начинается его подготовка к лабораторному/ практическому/ семинарскому/ методическому/ клиническому практическому занятию. Вместе с тем, лекция лишь организует мыслительную деятельность, но не обеспечивает глубину усвоения программного материала. При подготовке к такому виду занятий можно выделить 2 этапа:

- 1-й - организационный,
- 2-й - закрепление и углубление теоретических знаний.

На первом этапе обучающийся (студент) планирует свою самостоятельную работу, которая включает:

- уяснение задания на самостоятельную работу;
- подбор рекомендованной литературы;
- составление плана работы, в котором определяются основные пункты предстоящей подготовки.

Составление плана дисциплинирует и повышает организованность в работе. Второй этап включает непосредственную подготовку обучающегося (студента) к занятию. Начинать надо с изучения рекомендованной литературы. Необходимо помнить, что на лекции обычно рассматривается не весь материал, а только его часть. Остальная его часть восполняется в процессе самостоятельной работы. В связи с этим работа с рекомендованной литературой обязательна. Особое внимание при этом необходимо обратить на содержание основных положений и выводов, объяснение явлений и фактов, уяснение практического приложения рассматриваемых теоретических вопросов. В процессе этой работы обучающийся (студент) должен стремиться понять и запомнить основные положения рассматриваемого материала, примеры, поясняющие его, а также разобраться в иллюстративном материале. Заканчивать подготовку следует составлением плана (перечня основных пунктов) по изучаемому материалу (вопросу). Такой план позволяет составить концентрированное, сжатое представление по

изучаемым вопросам. В процессе подготовки к семинарскому занятию рекомендуется взаимное обсуждение материала, во время которого закрепляются знания, а также приобретается практика в изложении и разъяснении полученных знаний, развивается речь. При необходимости следует обращаться за консультацией к преподавателю. Идя на консультацию, необходимо хорошо продумать вопросы, которые требуют разъяснения. В начале семинарского занятия обучающиеся (студента) под руководством преподавателя более глубоко осмысливают теоретические положения по теме занятия, раскрывают и объясняют основные явления и факты. В процессе творческого обсуждения и дискуссии вырабатываются умения и навыки использовать приобретенные знания для решения практических задач.

В) Рекомендации по самостоятельной работе обучающегося (студента) над изучаемым материалом

Успешное освоение данного курса базируется на рациональном сочетании нескольких видов учебной деятельности - лекций, семинарских занятий, самостоятельной работы. При этом самостоятельную работу следует рассматривать одним из главных звеньев полноценного высшего образования, на которую отводится значительная часть учебного времени.

Самостоятельная работа студентов складывается из следующих составляющих:

- работа с основной и дополнительной литературой, с материалами интернета и конспектами лекций;
- внеаудиторная подготовка к контрольным работам, выполнение докладов, рефератов и курсовых работ;
- выполнение самостоятельных практических работ;
- подготовка к экзаменам (зачетам) непосредственно перед ними.

Для правильной организации работы необходимо учитывать порядок изучения разделов курса, находящихся в строгой логической последовательности. Поэтому хорошее усвоение одной части дисциплины является предпосылкой для успешного перехода к следующей. Задания, проблемные вопросы, предложенные для изучения дисциплины, в том числе и для самостоятельного выполнения, носят междисциплинарный характер и базируются, прежде всего, на причинно-следственных связях между компонентами окружающего нас мира. В течение семестра, необходимо подготовить рефераты (проекты) с использованием рекомендуемой основной и дополнительной литературы и сдать рефераты для проверки преподавателю. Важным составляющим в изучении данного курса является решение ситуационных задач и работа над проблемно-аналитическими заданиями, что предполагает знание соответствующей научной терминологии и т.д.

Для лучшего запоминания материала целесообразно использовать индивидуальные особенности и разные виды памяти: зрительную, слуховую, ассоциативную. Успешному запоминанию также способствует приведение ярких свидетельств и наглядных примеров. Учебный материал должен постоянно повторяться и закрепляться.

При выполнении докладов, творческих, информационных, исследовательских проектов особое внимание следует обращать на подбор источников информации и методику работы с ними.

Для успешной сдачи экзамена (зачета) рекомендуется соблюдать следующие правила:

1. Подготовка к экзамену (зачету) должна проводиться систематически, в течение всего семестра.
2. Интенсивная подготовка должна начаться не позднее, чем за месяц до экзамена.
3. Время непосредственно перед экзаменом (зачетом) лучше использовать таким образом, чтобы оставить последний день свободным для повторения курса в целом, для систематизации материала и доработки отдельных вопросов.

На экзамене высокую оценку получают студенты, использующие данные, полученные в процессе выполнения самостоятельных работ, а также использующие собственные выводы на основе изученного материала.

Учитывая значительный объем теоретического материала, студентам рекомендуется регулярное посещение и подробное конспектирование лекций.

8. ОСОБЕННОСТИ РЕАЛИЗАЦИИ ДИСЦИПЛИНЫ ДЛЯ ИНВАЛИДОВ И ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ

Обучающимся с ограниченными возможностями здоровья предоставляются специальные учебники и учебные пособия, иная учебная литература, специальные технические средства обучения коллективного и индивидуального пользования, предоставление услуг ассистента (помощника), оказывающего обучающимся необходимую техническую помощь, а также услуги сурдопереводчиков и тифлосурдопереводчиков.

а) для слабовидящих:

- на промежуточной аттестации присутствует ассистент, оказывающий студенту необходимую техническую помощь с учетом индивидуальных особенностей (он помогает занять рабочее место, передвигаться, прочитать и оформить задание, в том числе записывая под диктовку);

- задания для выполнения, а также инструкция о порядке проведения промежуточной аттестации оформляются увеличенным шрифтом;

- задания для выполнения на промежуточной аттестации зачитываются ассистентом;

- письменные задания выполняются на бумаге, надиктовываются ассистенту;

- обеспечивается индивидуальное равномерное освещение не менее 300 люкс;

- студенту для выполнения задания при необходимости предоставляется увеличивающее устройство;

в) для глухих и слабослышащих:

- на промежуточной аттестации присутствует ассистент, оказывающий студенту необходимую техническую помощь с учетом индивидуальных особенностей (он помогает занять рабочее место, передвигаться, прочитать и оформить задание, в том числе записывая под диктовку);

- промежуточно-заочная аттестация проводится в письменной форме;

- обеспечивается наличие звукоусиливающей аппаратуры коллективного пользования, при необходимости поступающим предоставляется звукоусиливающая аппаратура индивидуального пользования;

- по желанию студента промежуточно-заочная аттестация может проводиться в письменной форме;

д) для лиц с нарушениями опорно-двигательного аппарата (тяжелыми нарушениями двигательных функций верхних конечностей или отсутствием верхних конечностей):

- письменные задания выполняются на компьютере со специализированным программным обеспечением или надиктовываются ассистенту;

- по желанию студента промежуточно-заочная аттестация проводится в устной форме.

Примечание:

а) Для обучающегося (бакалавра), осваивающего учебную дисциплину, обязательный компонент основной профессиональной образовательной программы высшего образования — программы бакалавриата по направлению подготовки **09.03.02 Информационные системы и технологии (направленность (профиль) «Информационные системы в экономике»), форма обучения — очно-заочная), одобренной на заседании Учёного совета образовательной организации, утверждённой ректором Частного образовательного учреждения высшего образования «Высшая школа предпринимательства», **по индивидуальному учебному плану (при наличии факта зачисления в образовательную организацию такого обучающегося (бакалавра)), Институт:****

- разрабатывает, согласовывает с участниками образовательных отношений и утверждает в установленном порядке согласно соответствующему локальному нормативному акту **индивидуальный учебный план** конкретного обучающегося (бакалавра) (*учебный план, обеспечивающий освоение конкретной основной образовательной программы высшего образования на основе индивидуализации её содержания с учётом особенностей и*

образовательных потребностей конкретного обучающегося (бакалавра));

- устанавливает для конкретного обучающегося (бакалавра) по индивидуальному учебному плану **одинаковые дидактические единицы** — элементы содержания учебного материала, изложенного в виде утверждённой в установленном образовательной организацией порядке согласно соответствующему локальному нормативному акту рабочей программы учебной дисциплины, обязательного компонента разработанной и реализуемой Институтом основной профессиональной образовательной программы высшего образования — программы бакалавриата по направлению подготовки **09.03.02 Информационные системы и технологии** (направленность (профиль) «Информационные системы в экономике»), форма обучения — очно-заочная), как и для обучающего (бакалавра), осваивающего основную образовательную программу высшего образования в учебной группе;

- определяет в индивидуальном учебном плане конкретного обучающегося (бакалавра) **объём учебной дисциплины** с указанием количества академических часов/ ЗЕТ, выделенных на его контактную работу (групповую и (или) индивидуальную работу) с руководящими и (или) научно-педагогическими работниками, реализующими основную образовательную программу высшего образования;

- определяет в индивидуальном учебном плане конкретного обучающегося (бакалавра) количество академических часов/ ЗЕТ по учебной дисциплине, выделенных на его самостоятельную работу (*при необходимости*).

б) Для обучающегося (бакалавра) с ограниченными возможностями здоровья и инвалида, осваивающего учебную дисциплину, обязательный компонент основной профессиональной образовательной программы высшего образования — программы бакалавриата по направлению подготовки **09.03.02 Информационные системы и технологии** (направленность (профиль) «Информационные системы в экономике»), форма обучения — очно-заочная), одобренной на заседании Учёного совета образовательной организации, утверждённой ректором Частного образовательного учреждения высшего образования «Высшая школа предпринимательства», (*при наличии факта зачисления в образовательную организацию такого обучающегося (бакалавра) с учётом конкретной (конкретных) нозологии (нозологий)*), **Институт:**

- разрабатывает, согласовывает с участниками образовательных отношений и утверждает в установленном порядке согласно соответствующему локальному нормативному акту **индивидуальный учебный план** конкретного обучающегося (бакалавра) с ограниченными возможностями здоровья/ инвалида (*при наличии факта зачисления в образовательную организацию такого обучающегося (бакалавра) с учётом конкретной (конкретных) нозологии (нозологий)*) (учебный план, обеспечивающий освоение конкретной основной образовательной программы высшего образования на основе индивидуализации её содержания с учётом особенностей и образовательных потребностей конкретного обучающегося (бакалавра));

- устанавливает для конкретного обучающегося (бакалавра) с ограниченными возможностями здоровья содержание образования (**одинаковые дидактические единицы** — элементы содержания учебного материала, как и для обучающего (бакалавра), осваивающего основную образовательную программу высшего образования в учебной группе) и условия организации обучения, изложенного в виде утверждённой в установленном Институте порядке согласно соответствующему локальному нормативному акту рабочей программы учебной дисциплины, обязательного компонента разработанной и реализуемой им адаптированной основной профессиональной образовательной программы высшего образования - программы бакалавриата по направлению подготовки **09.03.02 Информационные системы и технологии** (направленность (профиль) «Информационные системы в экономике»), форма обучения — очно-заочная), а для инвалидов также в соответствии с индивидуальной программой реабилитации инвалида (для конкретного обучающегося (бакалавра) с ограниченными возможностями здоровья/ инвалида (*при наличии факта зачисления в образовательную организацию такого обучающегося (бакалавра) с учётом конкретной (конкретных) нозологии (нозологий)*));

- определяет в индивидуальном учебном плане конкретного обучающегося бакалавра) с ограниченными возможностями здоровья/ инвалида (*при наличии факта зачисления такого обучающегося (бакалавра) с учётом конкретной (конкретных) нозологии (нозологий)*) **объём учебной дисциплины** с указанием количества академических часов/ ЗЕТ, выделенных на его контактную работу (групповую и (или) индивидуальную работу) с руководящими и (или) научно-педагогическими работниками, реализующими основную образовательную программу высшего образования;

- определяет в индивидуальном учебном плане конкретного обучающегося (бакалавра) с ограниченными возможностями здоровья/ инвалида (*при наличии факта зачисления в образовательную организацию такого обучающегося (бакалавра) с учётом конкретной (конкретных) нозологии (нозологий)*) количество академических часов/ ЗЕТ по учебной дисциплине, выделенных на его самостоятельную работу (*при необходимости*).

9. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

9.1 Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

Основная литература:

1. Игнатъев А.В., Тестирование программного обеспечения: учебное пособие для вузов. / А.В. Игнатъев. - 3-е изд. - Лань, 2023. - 56 с. - ISBN 978-5-507-45425-9.

2. Морозова Ю.В., Тестирование программного обеспечения [Электронный ресурс] : учебное пособие / Ю.В. Морозова. - Томск : Эль-Контент, 2019. - 120 с. - ISBN 978-5-4332-0279-5. - Режим доступа: <https://znanium.com/catalog/product/1845910>

3. Романов А.А., Тестирование программного обеспечения : лабораторный практикум / А.А. Романов. - Ульяновск : УлГТУ, 2022. - 45 с.

4. Савинов, А.В., Тестирование программного обеспечения: учебное пособие / А.В. Савинов. - М.: Бином, 2014. - 384 с.

Дополнительная литература:

1. Аниче М., Эффективное тестирование программного обеспечения / М. Аниче. - ДМК Пресс, 2023. - 370 с. - ISBN 978-5-97060-997-2.

2. Бёрд П., Искусство тестирования программ / П. Бёрд. - СПб.: Питер, 2012. - 336 с.

3. Канер С., Тестирование программного обеспечения / С. Канер, Дж. Фолк, Х. Кейнер. - СПб.: Питер, 2017. - 432 с.

4. Майерс Г., Баджетт Т., Сандлер К. Искусство тестирования программ / Г. Майерс, Т. Баджетт, К. Сандлер. - 3-е изд. - Диалектика-Вильямс, 2020. - 272 с. - ISBN 978-5-907203-66-2.

5. Савин Р., Тестирование дот ком / Р. Савин. - СПб.: БХВ-Петербург, 2016. - 480 с.

6. Паттон Р., Практика тестирования программного обеспечения / Р. Паттон. - 2-е изд. - М.: Вильямс, 2015. - 528 с.

9.2 Используемое программное обеспечение (комплект лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства в соответствии с п.4.3.2. ФГОС ВО 09.03.02):

1. Microsoft Windows 11 Pro или аналогичная ОС, включая дистрибутивы Linux, например Debian, Ubuntu, OpenSuse, в том числе отечественного производства, например ОС Astra Linux Common Edition (Разработчик: АО «НПО РусБИТех»), ОС «РОСА» (Разработчик: «НТИЦ ИТ РОСА»).

2. Microsoft Office 365 или аналогичный офисный пакет, например OpenOffice, LibreOffice, ONLYOFFICE, в том числе отечественного производства, например МойОфис (Разработчик: ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ»).

3. Adobe Reader или аналогичный просмотрщик PDF, например Okular, Foxit Reader, в том числе отечественного производства, например Окуляр ГОСТ (Разработчик: ООО «Лаборатория

50»).

4. Google Chrome или аналогичный веб-браузер, например Microsoft Edge, Mozilla Firefox, в том числе отечественного производства, например Яндекс.Браузер (Разработчик: ООО «ЯНДЕКС»).

5. Microsoft Visual Studio Code или аналогичная IDE, например Sublime Text, Eclipse, в том числе отечественного производства

6. PyCharm / IntelliJ IDEA / CLion / WebStorm либо аналогичная IDE полного стека, в том числе отечественного производства

7. MySQL CE 8.0 / MySQL Workbench или аналогичные СУБД, например MS SQL, PostgreSQL, в том числе отечественного производства

8. Android Studio или аналогичная IDE для разработки мобильных приложений, в том числе отечественного производства

9. Figma или аналогичное ПО для подготовки макетов, например Penpot, Lunacy, в том числе отечественного производства.

9.3 Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины (модуля) (в соответствии с п.4.3.4. ФГОС ВО 09.03.02)

1. Электронно-библиотечная система BOOK.RU [Электронный ресурс]. - Режим доступа: <https://book.ru/>

2. Habrahabr: Тестирование ПО [Электронный ресурс]. - Режим доступа: <https://habr.com/ru/hub/testing/>

3. Software-Testing.Ru: Введение в тестирование [Электронный ресурс]. - Режим доступа: <https://software-testing.ru/>

4. DTesting: Блог о тестировании [Электронный ресурс]. - Режим доступа: <http://dtesting.com/>

5. SQA.RU: Портал для тестировщиков [Электронный ресурс]. - Режим доступа: <http://www.sqaforums.ru/>

6. Электронно-библиотечная система IPRbooks [Электронный ресурс]. - Режим доступа: <https://www.iprbookshop.ru/>

9.4 Базы данных, информационно-справочные и поисковые системы (в соответствии с п.4.3.4. ФГОС ВО 09.03.02)

1. КонсультантПлюс: справочно-поисковая система [Электронный ресурс]. - <http://www.consultant.ru>

2. Мировая цифровая библиотека: <http://wdl.org/ru>

3. Научная электронная библиотека «Scopus»: <https://www.scopus.com>

4. Научная электронная библиотека ScienceDirect: <http://www.sciencedirect.com>

5. Научная электронная библиотека «eLIBRARY»: <https://elibrary.ru>

6. Портал «Гуманитарное образование» <http://www.humanities.edu.ru>

7. Федеральный портал «Российское образование» <http://www.edu.ru>

8. Федеральное хранилище «Единая коллекция цифровых образовательных ресурсов» <http://school-collection.edu.ru>

9. Поисковые системы Yandex, Rambler и др.

10. Электронная библиотека Российской Государственной Библиотеки (РГБ): <http://elibrary.rsl.ru>

11. Электронно-библиотечная система <http://www.sciteclibrary.ru>

10. Материально-техническое обеспечение дисциплины

Помещения для проведения всех видов работы, предусмотренных учебным планом, укомплектованы необходимым оборудованием и техническими средствами обучения.

Наименование помещений для проведения всех	Адрес (местоположение) помещений для
--	--------------------------------------

<p>видов учебной деятельности, предусмотренной учебным планом, в том числе помещения для самостоятельной работы, с указанием перечня основного оборудования, учебно-наглядных пособий и используемого программного обеспечения</p>	<p>проведения всех видов учебной деятельности, предусмотренной учебным планом (в случае реализации образовательной программы в сетевой форме дополнительно указывается наименование организации, с которой заключен договор)</p>
<p>Специализированная многофункциональная учебная аудитория для проведения учебных занятий лекционного типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, в том числе, для организации практической подготовки обучающихся, с перечнем основного оборудования:</p> <ul style="list-style-type: none"> - Столы для обучающихся; - Стулья для обучающихся; - Стол педагогического работника; - Стул педагогического работника; - Компьютеры с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду лицензиата; - Маркерная или меловая доска; - Проектор. 	<p>170001, Россия, город Тверь, улица Спартака, дом 26а</p>
<p>Специализированная многофункциональная учебная аудитория для проведения учебных занятий семинарского типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, в том числе, для организации практической подготовки обучающихся, с перечнем основного оборудования:</p> <ul style="list-style-type: none"> - Столы для обучающихся; - Стулья для обучающихся; - Стол педагогического работника; - Стул педагогического работника; - Компьютеры с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду лицензиата; - Маркерная или меловая доска; - Проектор. 	<p>170001, Россия, город Тверь, улица Спартака, дом 26а</p>
<p>Помещение для самостоятельной работы обучающихся с перечнем основного оборудования:</p> <ul style="list-style-type: none"> - Столы для обучающихся; - Стулья для обучающихся; 	<p>170001, Россия, город Тверь, улица Спартака, дом 26а</p>

<ul style="list-style-type: none"> - Стол педагогического работника; - Стул педагогического работника; - Компьютеры с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду лицензиата; - Маркерная или меловая доска; - Проектор. 	
<p>Помещение для практических занятий на персональных компьютерах:</p> <ul style="list-style-type: none"> - Столы для обучающихся; - Стулья для обучающихся; - Стол педагогического работника; - Стул педагогического работника; - Компьютеры с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду лицензиата; - Ноутбуки с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду лицензиата; - Маркерная или меловая доска; - Проектор. 	<p>170001, Россия, город Тверь, улица Спартак, дом 26а</p>

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**



**Частное учреждение высшего образования
«Высшая школа предпринимательства (институт)»
(ЧУВО «ВШП»)**

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

по дисциплине

Б1.В.09 «Тестирование программного обеспечения»

Направление подготовки: 09.03.02 Информационные системы и технологии

Направленность (профиль) программы бакалавриата

«Информационные системы в экономике»

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ В РАМКАХ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОСВОЕНИЯ ОСНОВНОЙ ПРОФЕССИОНАЛЬНОЙ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Процесс освоения учебной дисциплины направлен на формирование у обучающихся следующих компетенций.

В результате освоения учебной дисциплины обучающийся должен демонстрировать следующие результаты обучения: УК-2, УК-3, ПК-2, ПК-3, ПК-4.

Код компетенции	Наименование компетенции	Индекс и наименование индикатора содержания компетенции	Дескрипторы – основные признаки освоения (показатели достижения результата)
УК-2	Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	УК-2.1 Способен определять круг задач в рамках поставленной цели	Знать: <ul style="list-style-type: none"> - Методы анализа и декомпозиции задач в программировании. - Основные этапы разработки программного обеспечения. Уметь: <ul style="list-style-type: none"> - Определять задачи, вытекающие из поставленной цели, и разрабатывать план их выполнения. - Структурировать задачи и определять их приоритетность. Владеть: <ul style="list-style-type: none"> - Навыками анализа и декомпозиции задач. - Методами планирования и структурирования задач в программировании.
		УК-2.2 Способен выбирать оптимальные способы решения задач, исходя из правовых норм, ресурсов и ограничений	Знать: <ul style="list-style-type: none"> - Правовые нормы и ограничения в области программирования. - Методы оценки ресурсов и ограничений при разработке программного обеспечения. Уметь: <ul style="list-style-type: none"> - Выбирать оптимальные методы и инструменты для решения задач в программировании. - Обосновывать выбор методов и инструментов с учетом правовых норм, ресурсов и ограничений. Владеть: <ul style="list-style-type: none"> - Навыками оценки ресурсов и ограничений при выборе способов решения задач. - Методами обоснования выбора оптимальных решений в программировании.
УК-3	Способен осуществлять социальное взаимодействие и реализовывать свою роль в команде	УК-3.1 Способен осуществлять социальное взаимодействие в команде	Знать: <ul style="list-style-type: none"> - Принципы и методы командной работы в разработке программного обеспечения. - Основы делового общения и взаимодействия в команде. Уметь: <ul style="list-style-type: none"> - Эффективно взаимодействовать с членами команды при разработке программного обеспечения.

			<ul style="list-style-type: none"> - Принимать участие в обсуждениях и совместных проектах. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками командного взаимодействия в процессе разработки программного обеспечения. - Методами делового общения и взаимодействия в команде.
		<p>УК-3.2 Способен реализовывать свою роль в команде</p>	<p>Знать:</p> <ul style="list-style-type: none"> - Различные роли и функции в команде разработки программного обеспечения. - Методы распределения обязанностей в команде. <p>Уметь:</p> <ul style="list-style-type: none"> - Определять и выполнять свою роль в команде разработки программного обеспечения. - Принимать ответственность за выполнение задач в команде. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками самоуправления и координации в рамках команды разработки. - Методами эффективного выполнения своей роли в команде.
ПК-2	Проверка работоспособности и рефакторинг кода программного обеспечения	<p>ПК-2.1 Способен проверять работоспособность программного кода</p>	<p>Знать:</p> <ul style="list-style-type: none"> - Принципы и методы тестирования программного кода. - Основы юнит-тестирования и функционального тестирования. <p>Уметь:</p> <ul style="list-style-type: none"> - Разрабатывать и выполнять тестовые сценарии для проверки работоспособности кода. - Использовать тестовые фреймворки и инструменты для автоматизированного тестирования. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками написания и выполнения тестов. - Навыками анализа результатов тестирования и устранения обнаруженных проблем.
		<p>ПК-2.2 Способен рефакторить программный код</p>	<p>Знать:</p> <ul style="list-style-type: none"> - Принципы и методы рефакторинга программного кода. - Паттерны и антипаттерны проектирования кода. <p>Уметь:</p> <ul style="list-style-type: none"> - Проводить анализ и улучшение структуры кода. - Вносить изменения в код, сохраняя его функциональность. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками применения рефакторинга для улучшения

			<p>читаемости и поддерживаемости кода.</p> <ul style="list-style-type: none"> - Навыками использования инструментов для автоматизированного рефакторинга.
ПК-3	Интеграция программных модулей и компонентов и проверка работоспособности выпусков программного продукта	ПК-3.1 Способен интегрировать программные модули и компоненты	<p>Знать:</p> <ul style="list-style-type: none"> - Принципы и методы интеграции программных модулей. - Основы межмодульного взаимодействия и совместимости. <p>Уметь:</p> <ul style="list-style-type: none"> - Планировать и проводить интеграцию модулей в общий проект. - Решать проблемы, возникающие при интеграции различных компонентов. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками использования инструментов и платформ для интеграции. - Навыками написания интеграционных тестов.
		ПК-3.2 Способен проверять работоспособность выпусков программного продукта	<p>Знать:</p> <ul style="list-style-type: none"> - Методы и инструменты для проверки выпусков программного продукта. - Основы тестирования и валидации программного обеспечения. <p>Уметь:</p> <ul style="list-style-type: none"> - Проводить проверку и валидацию программного продукта. - Анализировать результаты тестирования и устранять выявленные дефекты. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками работы с инструментами для проверки и тестирования выпусков. - Навыками документирования результатов тестирования и предложений по улучшению.
ПК-4	Разработка требований и проектирование программного обеспечения	ПК-4.1 Способен разрабатывать требования к программному обеспечению	<p>Знать:</p> <ul style="list-style-type: none"> - Методы и техники сбора и анализа требований. - Принципы написания требований и спецификаций. <p>Уметь:</p> <ul style="list-style-type: none"> - Собирать и документировать требования к программному обеспечению. - Анализировать и уточнять требования с заинтересованными сторонами. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками написания четких и измеримых требований.

			<ul style="list-style-type: none"> - Навыками работы с инструментами для управления требованиями.
		ПК-4.2 Способен проектировать программное обеспечение	Знать: <ul style="list-style-type: none"> - Принципы и методы проектирования программного обеспечения. - Основы архитектурных шаблонов и стилей. Уметь: <ul style="list-style-type: none"> - Разрабатывать архитектуру и дизайн программного обеспечения. - Создавать диаграммы и модели для визуализации архитектуры. Владеть: <ul style="list-style-type: none"> - Навыками применения архитектурных паттернов. - Навыками использования инструментов для проектирования и моделирования программного обеспечения.

КРИТЕРИИ ОЦЕНИВАНИЯ КОМПЕТЕНЦИЙ

(признак, на основании которого, проводится оценка по выбранному показателю)

<i>Показатель оценивания компетенций</i>	<i>Результат обучения</i>	<i>Критерии оценивания компетенций</i>
Высокий уровень (отлично)	Знать	Обучающийся продемонстрировал: глубокие исчерпывающие знания и понимание учебного материала; содержательные, полные, правильные и конкретные ответы на все вопросы, включая дополнительные; свободное владение основной и дополнительной литературой, рекомендованной учебной программой дисциплины.
	Уметь	Обучающийся продемонстрировал: понимание учебного материала; умение свободно решать практические задания (ситуационные задачи), которые следует выполнить или описание результата, который нужно получить и др.; логически последовательные, содержательные, полные, правильные и конкретные ответы (решения) на все поставленные задания (вопросы), включая дополнительные; свободное владение основной и дополнительной литературой, рекомендованной учебной программой дисциплины.
	Владеть	Обучающийся продемонстрировал: понимание учебного материала; умение свободно решать комплексные практические задания (решения задач по нестандартным ситуациям); логически последовательные, полные, правильные и конкретные ответы в ходе защиты задания, включая дополнительные уточняющие вопросы (задания); свободное владение основной и дополнительной литературой, рекомендованной учебной программой дисциплины.
Средний уровень (хорошо)	Знать	Обучающийся продемонстрировал: твердые и достаточно полные знания учебного материала; правильное понимание сущности и взаимосвязи рассматриваемых процессов и явлений; последовательные, правильные, конкретные ответы на поставленные вопросы при свободном устранении замечаний по отдельным вопросам; достаточное владение литературой, рекомендованной учебной программой дисциплины
	Уметь	Обучающийся продемонстрировал: понимание учебного материала; логически последовательные, правильные и конкретные ответы (решения) на основные задания (вопросы),

		включая дополнительные; устранение замечаний по отдельным элементам задания (вопроса); владение основной и дополнительной литературой, рекомендованной учебной программой дисциплины
	Владеть	Обучающийся продемонстрировал: понимание учебного материала; продемонстрировал логически последовательные, достаточно полные, правильные ответы, включая дополнительные; самостоятельно устранил замечания по отдельным элементам задания (вопроса); владение основной и дополнительной литературой, рекомендованной учебной программой дисциплины
Достаточный уровень (удовлетворительно)	Знать	Обучающийся продемонстрировал: твердые знания и понимание основного учебного материала; правильные, без грубых ошибок, ответы на поставленные вопросы при устранении неточностей и несущественных ошибок в освещении отдельных положений при наводящих вопросах преподавателя; недостаточно полное владение литературой, рекомендованной учебной программой дисциплины
	Уметь	Обучающийся продемонстрировал: понимание основного учебного материала; правильные, без грубых ошибок, ответы (решения) на основные задания (вопросы), включая дополнительные, устранение, при наводящих вопросах преподавателя, замечаний по отдельным элементам задания (вопроса); недостаточно полное владение литературой, рекомендованной учебной программой дисциплины
	Владеть	Обучающийся понимание основного учебного материала; без грубых ошибок дал ответы на поставленные вопросы при устранении неточностей и ошибок в решениях в ходе защиты задания (проекта, портфолио) при наводящих вопросах преподавателя; недостаточно полное владение литературой, рекомендованной учебной программой дисциплины

ОПИСАНИЕ ШКАЛ ОЦЕНИВАНИЯ

При проведении промежуточной аттестации в ЧУВО «ВШП» используются традиционные формы аттестации:

Форма промежуточной аттестации	Шкала оценивания
ЗАЧЕТ	«зачтено», «незачтено»
ЭКЗАМЕН	«отлично», «хорошо», «удовлетворительно», «неудовлетворительно»

КРИТЕРИИ И ПРОЦЕДУРЫ ОЦЕНИВАНИЯ КОМПЕТЕНЦИЙ НА РАЗЛИЧНЫХ ЭТАПАХ ИХ ФОРМИРОВАНИЯ

Для оценивания результатов обучения в виде **ЗНАНИЙ** используются следующие процедуры и технологии:

- тестирование.

Для оценивания результатов обучения в виде **УМЕНИЙ и ВЛАДЕНИЙ** используются следующие процедуры и технологии:

- устный или письменный ответ на вопрос.
- практические задания, включающие одну или несколько задач (вопросов) в виде краткой формулировки действий (комплекса действий), которые следует выполнить, или описать результат, который нужно получить.

Критерии оценивания результата обучения по дисциплине (модулю)

Результат обучения по дисциплине	ШКАЛА ОЦЕНИВАНИЯ				Процедуры оценивания
	«отлично»	«хорошо»	«удовлетворительно»	«неудовлетворительно»	

(модулю)					
<u>УК-2</u> , <u>УК-3</u> , <u>ПК-2</u> , <u>ПК-3</u> , <u>ПК-4</u> Знать:	Обучаемый продемонстрировал: ал: глубокие исчерпывающие знания и понимание учебного материала; содержательные, полные, правильные и конкретные ответы на все вопросы, включая дополнительные; свободное владение основной и дополнительной литературой, рекомендованной учебной программой дисциплины.	Обучаемый продемонстрировал: ал: твердые и достаточно полные знания учебного материала; правильное понимание сущности и взаимосвязи рассматриваемых процессов и явлений; последовательные, правильные, конкретные ответы на поставленные вопросы при свободном устранении замечаний по отдельным вопросам; достаточное владение литературой.	Обучаемый продемонстрировал: ал: твердые знания и понимание основного учебного материала; правильные, без грубых ошибок, ответы на поставленные вопросы при устранении неточностей и несущественных ошибок в освещении отдельных положений при наводящих вопросах преподавателя; недостаточно полное владение литературой, рекомендованной учебной программой дисциплины.	Обучаемый продемонстрировал: ал: неправильные ответы на основные вопросы; грубые ошибки в ответах; непонимание сущности излагаемых вопросов; неуверенные и неточные ответы на дополнительные вопросы; не владеет основной литературой, рекомендованной учебной программой дисциплины.	Тестовые задания
<u>УК-2</u> , <u>УК-3</u> , <u>ПК-2</u> , <u>ПК-3</u> , <u>ПК-4</u> Уметь:	Обучаемый продемонстрировал: ал: понимание учебного материала, содержательные, полные, правильные и конкретные ответы на все поставленные вопросы, включая дополнительные; свободное владение основной и дополнительной литературой, рекомендованной учебной программой дисциплины	Обучаемый продемонстрировал: ал: понимание учебного материала; логически последовательные, правильные и конкретные ответы на основные задания/вопросы, включая дополнительные; устранение замечаний по отдельным элементам задания; владение основной и дополнительной литературой, рекомендованной учебной программой дисциплины	Обучаемый продемонстрировал: ал: понимание основного учебного материала; правильные, без грубых ошибок, ответы на основные вопросы, включая дополнительные, при устранении, при наводящих вопросах преподавателя, замечаний по отдельным элементам задания; недостаточное полное владение литературой, рекомендованной учебной программой дисциплины	Обучаемый продемонстрировал: ал: непонимание основного учебного материала; не дал правильные ответы на основные вопросы, включая дополнительные; не устранил, при наводящих вопросах преподавателя, замечания и грубые ошибки по вопросу; не владеет основной литературой, рекомендованной учебной программой дисциплины	Вопросы Практические задания
<u>УК-2</u> , <u>УК-3</u> , <u>ПК-2</u> , <u>ПК-3</u> , <u>ПК-4</u> Владеть:	Обучаемый продемонстрировал: ал: понимание учебного материала; правильные и конкретные	Обучаемый продемонстрировал: ал: понимание учебного материала; продемонстрировал логически	Обучаемый продемонстрировал: ал: понимание основного учебного материала; без грубых ошибок дал ответы на	Обучаемый продемонстрировал: ал: непонимание основного учебного материала; дал неправильные ответы на поставленные	Вопросы Практические задания

	ответы, включая дополнительные уточняющие вопросы; свободное владение основной и дополнительной литературой, рекомендованной учебной программой дисциплины	последовательные, достаточно полные, верные ответы; самостоятельно устранил замечания по отдельным элементам; владение основной и дополнительной литературой, рекомендованной учебной программой дисциплины	поставленные вопросы, в том числе при наводящих вопросах преподавателя; недостаточно полное владение литературой, рекомендованной учебной программой дисциплины	вопросы; не владеет основной учебной литературой, рекомендованной учебной программой дисциплины	
--	--	---	---	---	--

1. Оценочные материалы для самостоятельной работы обучающихся (студентов)

1.1 Доклад

Доклад позволит студентам углубиться в методы и практики тестирования программного обеспечения, а также понять их важность и применение в процессе разработки ПО. Доклад покрывает компетенции УК-2.1, УК-2.2, УК-3.1, УК-3.2, ПК-2.1, ПК-2.2, ПК-3.1, ПК-3.2, ПК-4.1, ПК-4.2

Примерная тематика докладов:

1. **Введение в тестирование программного обеспечения:** Основные понятия и цели тестирования, виды тестирования (функциональное, нефункциональное, регрессионное и т.д.), роль тестирования в жизненном цикле разработки ПО.
2. **Основы планирования тестирования:** Разработка стратегии тестирования, создание тестового плана, определение тестируемых требований и задач, выбор инструментов и ресурсов для тестирования.
3. **Методы тестирования программного обеспечения:** Динамическое и статическое тестирование, тестирование "черного ящика" и "белого ящика", интеграционное и системное тестирование, методы автоматизированного тестирования.
4. **Инструменты автоматизированного тестирования:** Обзор популярных инструментов автоматизированного тестирования (Selenium, JUnit, TestNG и др.), внедрение автоматизации в процесс тестирования, создание и выполнение автотестов.
5. **Управление тестированием и тестовые данные:** Управление тестовыми данными и конфигурациями, подготовка тестовых сценариев и данных, методы и инструменты для управления тестовыми данными.
6. **Тестирование производительности и нагрузки:** Основные концепции и цели тестирования производительности, инструменты и методы для проведения тестов производительности и нагрузочных тестов, анализ результатов и оптимизация производительности.
7. **Тестирование безопасности программного обеспечения:** Методы и техники тестирования безопасности, обнаружение уязвимостей и угроз, инструменты для тестирования безопасности (OWASP ZAP, Burp Suite и др.).
8. **Управление дефектами и отчетность:** Жизненный цикл дефекта, использование систем управления дефектами (Jira, Bugzilla и др.), создание и ведение отчетности по результатам тестирования.
9. **Взаимодействие в команде при тестировании:** Роли и обязанности в команде тестирования, методы и инструменты для совместной работы (Trello, Confluence и др.), коммуникация и управление взаимодействием в команде.
10. **Тестирование на финальных стадиях разработки и релизное тестирование:** Подготовка к финальному тестированию, проведение релизного тестирования, подготовка отчетов и документации для релиза.
11. **Современные тренды и будущее тестирования ПО:** Последние тенденции в тестировании ПО, влияние DevOps и CI/CD на процесс тестирования, будущее автоматизации и искусственного интеллекта в тестировании.

Цель написания докладов: Углубить понимание и критическое осмысление роли тестирования программного обеспечения в процессе разработки ПО, развивая аналитические и научные навыки студентов.

Структура доклада:

1. **Введение**
 - Краткое описание темы и целей доклада.

- Актуальность темы.
- 2. **Основная часть**
 - Теоретические основы темы.
 - История и эволюция (если применимо).
 - Применение в современной экономике/бизнесе.
 - Примеры и кейсы.
 - Проблемы и вызовы.
 - Перспективы и тенденции развития.
- 3. **Заключение**
 - Выводы по результатам исследования.
 - Значение тестирования ПО для современных информационных систем.
- 4. **Список использованных источников**
 - Перечень использованной литературы и интернет-ресурсов.

Критерии оценивания:

1. **Структура и логика изложения (20%)**
 - Четкая структура работы (введение, основная часть, заключение).
 - Логичность и последовательность изложения материала.
2. **Содержание (40%)**
 - Полнота раскрытия темы.
 - Описание основных этапов и методов тестирования ПО.
 - Анализ современных тенденций.
 - Примеры применения тестирования в реальных проектах.
3. **Аналитическая часть (20%)**
 - Глубина анализа роли тестирования ПО в процессе разработки.
 - Наличие собственных выводов и оценок.
4. **Оформление (10%)**
 - Соответствие требованиям к оформлению докладов (шрифт, отступы, заголовки и т.д.).
 - Корректное оформление ссылок и списка литературы.
5. **Язык и стиль (10%)**
 - Грамотность и точность изложения.
 - Научный стиль текста.

Требования к объему:

Объем доклада должен составлять 8-12 страниц печатного текста (шрифт Times New Roman, размер 12, интервал 1.5, поля 2 см со всех сторон).

2. Оценочные материалы для оценки текущей аттестации обучающихся (студентов)

2.1 Тестовые задания для текущего контроля успеваемости в виде ЗНАНИЙ

В тестовом задании вопросы, которые имеют закрытый характер.

Правильные ответы выделены знаком +.

1. **Какое из следующих утверждений правильно описывает основную цель тестирования программного обеспечения? (УК-2.1)**
 - Проверка функциональности и исправности программного кода.
 - Оценка пользовательского интерфейса.
 - Устранение всех багов в программном обеспечении.
 - Повышение уверенности в качестве программного обеспечения. +
2. **Что такое регрессионное тестирование? (ПК-2.1)**
 - Тестирование нового функционала.

- Тестирование производительности системы.
 - Повторное тестирование уже протестированных участков кода после изменений.
 - +
 - Тестирование безопасности.
3. **Как называется процесс определения тестируемых требований и задач? (УК-2.1)**
 - Разработка стратегии тестирования.
 - Создание тестового плана. +
 - Интеграционное тестирование.
 - Верификация требований.
 4. **Какой из следующих методов относится к статическому тестированию? (ПК-2.1)**
 - Юнит-тестирование.
 - Инспекция кода. +
 - Интеграционное тестирование.
 - Нагрузочное тестирование.
 5. **Какое тестирование проверяет систему как черный ящик, не обращая внимания на внутренние структуры? (УК-2.2)**
 - Тестирование "белого ящика".
 - Тестирование "черного ящика". +
 - Статическое тестирование.
 - Инспекционное тестирование.
 6. **Какой инструмент используется для автоматизированного функционального тестирования веб-приложений? (ПК-2.2)**
 - JIRA.
 - Selenium. +
 - Git.
 - Docker.
 7. **Что такое тестовый сценарий? (ПК-4.1)**
 - Документ, описывающий последовательность действий для проверки определенной функциональности. +
 - Код, выполняющий тесты автоматически.
 - Описание архитектуры тестируемого приложения.
 - Стратегия тестирования.
 8. **Какое из следующих утверждений правильно описывает нагрузочное тестирование? (ПК-3.2)**
 - Тестирование производительности системы под нормальной нагрузкой.
 - Тестирование системы на граничных условиях.
 - Тестирование системы под повышенной нагрузкой для определения ее пределов.
 - +
 - Тестирование совместимости системы с различными устройствами.
 9. **Какой инструмент используется для управления дефектами? (УК-3.1)**
 - Selenium.
 - JIRA. +
 - GitHub.
 - Jenkins.
 10. **Какое тестирование проводится с целью обнаружения уязвимостей и угроз в системе? (ПК-2.1)**
 - Нагрузочное тестирование.
 - Регрессионное тестирование.
 - Функциональное тестирование.
 - Тестирование безопасности. +
 11. **Что включает в себя тестовая документация? (ПК-4.1)**
 - Исходный код программы.

- Описание архитектуры системы.
 - Тест-планы, тест-кейсы, чек-листы, отчеты о дефектах. +
 - Пользовательская документация.
12. **Какая роль отвечает за координацию работы команды тестировщиков? (УК-3.2)**
- Разработчик.
 - Системный аналитик.
 - Тест-менеджер. +
 - Администратор базы данных.
13. **Что такое релизное тестирование? (ПК-3.2)**
- Тестирование производительности системы.
 - Финальное тестирование перед выпуском продукта. +
 - Тестирование нового функционала.
 - Тестирование безопасности.
14. **Какой из следующих подходов относится к тестированию на основе риска? (УК-2.2)**
- Тестирование всех функций равномерно.
 - Приоритетное тестирование наиболее критичных функций. +
 - Тестирование только новых функций.
 - Тестирование только интерфейса пользователя.
15. **Какой метод тестирования используется для проверки взаимодействия между модулями? (ПК-3.1)**
- Юнит-тестирование.
 - Интеграционное тестирование. +
 - Нагрузочное тестирование.
 - Регрессионное тестирование.
16. **Какое утверждение правильно описывает роль тестировщика в команде? (УК-3.1)**
- Разработчик нового функционала.
 - Ответственный за обнаружение и документирование дефектов. +
 - Администратор базы данных.
 - Руководитель проекта.
17. **Какой из следующих инструментов используется для тестирования безопасности? (ПК-3.2)**
- JIRA.
 - OWASP ZAP. +
 - Git.
 - Jenkins.
18. **Что такое CI/CD в контексте тестирования ПО? (ПК-4.2)**
- Контроль версий и управление конфигурациями.
 - Непрерывная интеграция и непрерывная доставка. +
 - Инспекция кода.
 - Тестирование производительности.
19. **Какой из методов тестирования относится к динамическому тестированию? (ПК-2.2)**
- Анализ статического кода.
 - Функциональное тестирование. +
 - Инспекционное тестирование.
 - Ревью кода.
20. **Какой подход позволяет обнаружить ошибки на ранних стадиях разработки ПО? (УК-2.1)**
- Релизное тестирование.
 - Статическое тестирование. +
 - Нагрузочное тестирование.
 - Тестирование безопасности.

21. **Как называется процесс выявления, анализа и устранения дефектов в ПО? (ПК-2.1)**
- Тестирование безопасности.
 - Управление конфигурациями.
 - Инспекция кода.
 - Дефект-менеджмент. +
22. **Какое из следующих утверждений верно для управления тестовыми данными? (ПК-4.2)**
- Тестовые данные создаются только в начале тестирования.
 - Тестовые данные должны быть максимально приближены к реальным. +
 - Тестовые данные используются только для функционального тестирования.
 - Тестовые данные не влияют на результаты тестирования.
23. **Какая методология разработки ПО требует тесного взаимодействия между разработчиками и тестировщиками? (УК-2.2)**
- Waterfall.
 - Agile. +
 - V-Model.
 - RUP.
24. **Какой из подходов тестирования предполагает тестирование всей системы в целом после завершения разработки всех модулей? (ПК-2.2)**
- Интеграционное тестирование.
 - Системное тестирование. +
 - Юнит-тестирование.
 - Релизное тестирование.
25. **Какое тестирование фокусируется на проверке производительности и поведения системы под высокой нагрузкой? (ПК-3.2)**
- Функциональное тестирование.
 - Регрессионное тестирование.
 - Нагрузочное тестирование. +
 - Тестирование безопасности.
26. **Какие из следующих задач входят в обязанности тест-менеджера? (УК-3.2)**
- Разработка функционала приложения.
 - Координация работы команды тестировщиков, составление тест-планов, управление ресурсами. +
 - Администрирование баз данных.
 - Создание пользовательской документации.
27. **Что такое тестовая стратегия и как она используется в тестировании ПО? (ПК-4.1)**
- Документ, описывающий конкретные тест-кейсы.
 - Документ, определяющий общий подход к тестированию, включая методы, ресурсы и график. +
 - Код, выполняющий автоматические тесты.
 - Описание архитектуры тестируемого приложения.

Критерии оценки результатов теста

1. "Неудовлетворительно" (0-39%)

- Студент ответил правильно на менее 40% вопросов.
- Значительные пробелы в знаниях по большинству тем.
- Неправильное понимание ключевых понятий и принципов.
- Неспособность применить теоретические знания на практике.

2. "Удовлетворительно" (40-59%)

- Студент ответил правильно на 40-59% вопросов.
- Основные понятия и принципы поняты частично, есть ошибки в ответах.

- Знания по большинству тем на базовом уровне, недостаточная глубина понимания.
 - Частичная способность применять теоретические знания на практике, нужны дополнительные разъяснения.
3. **"Хорошо" (60-79%)**
- Студент ответил правильно на 60-79% вопросов.
 - Хорошее понимание ключевых понятий и принципов, незначительные ошибки.
 - Знания по всем темам на достаточном уровне, однако есть некоторые пробелы.
 - Способность применять теоретические знания на практике, но требуется улучшение точности и уверенности.
4. **"Отлично" (80-100%)**
- Студент ответил правильно на 80-100% вопросов.
 - Полное и правильное понимание всех ключевых понятий и принципов.
 - Глубокие знания по всем темам, минимальные или отсутствующие ошибки.
 - Высокий уровень способности применять теоретические знания на практике, демонстрация уверенности и точности в ответах.

2.2 Вопросы для текущего контроля успеваемости в виде УМЕНИЙ

1. **Что такое тестирование программного обеспечения и для чего оно нужно? (УК-2.1)**
 - **Правильный ответ:** Тестирование программного обеспечения - это процесс проверки и оценки ПО с целью обнаружения дефектов и ошибок. Оно необходимо для обеспечения качества продукта, повышения уверенности в его работоспособности и удовлетворения требований пользователей.
2. **Какие виды тестирования вы знаете? (ПК-2.1)**
 - **Правильный ответ:** Основные виды тестирования включают функциональное, нефункциональное, регрессионное, интеграционное, системное, приемочное тестирование, тестирование безопасности и производительности.
3. **Что включает в себя тестовый план? (УК-2.1)**
 - **Правильный ответ:** Тестовый план включает стратегию тестирования, тестируемые требования, задачи, необходимые ресурсы, график проведения тестирования, критерии завершения и отчетность.
4. **Чем отличается тестирование "черного ящика" от тестирования "белого ящика"? (ПК-2.1)**
 - **Правильный ответ:** Тестирование "черного ящика" проверяет функциональность системы без знания внутренней структуры кода, в то время как тестирование "белого ящика" проверяет внутренние структуры и логику кода.
5. **Каковы основные цели автоматизированного тестирования? (ПК-2.2)**
 - **Правильный ответ:** Основные цели автоматизированного тестирования включают повышение эффективности тестирования, уменьшение времени на выполнение тестов, повышение точности и повторяемости тестирования, а также снижение затрат на тестирование.
6. **Какие инструменты используются для автоматизированного тестирования? (ПК-2.2)**
 - **Правильный ответ:** Популярные инструменты для автоматизированного тестирования включают Selenium, JUnit, TestNG, Jenkins, Appium и другие.
7. **Что такое тестовый сценарий и зачем он нужен? (ПК-4.1)**
 - **Правильный ответ:** Тестовый сценарий - это документ, описывающий последовательность действий для проверки определенной функциональности. Он нужен для систематизации тестирования и обеспечения покрытия всех тестируемых функций.
8. **В чем заключается нагрузочное тестирование? (ПК-3.2)**

- **Правильный ответ:** Нагрузочное тестирование заключается в проверке системы под повышенной нагрузкой для определения ее пределов и производительности. Оно помогает выявить слабые места и узкие места системы.
9. **Что такое дефект-менеджмент и как он реализуется? (УК-3.1)**
- **Правильный ответ:** Дефект-менеджмент - это процесс выявления, анализа, документирования и устранения дефектов в ПО. Он реализуется с помощью систем управления дефектами, таких как JIRA, Bugzilla и других.
10. **Какие методы тестирования используются для обеспечения безопасности ПО? (ПК-2.1)**
- **Правильный ответ:** Методы тестирования безопасности включают анализ кода на наличие уязвимостей, тестирование на проникновение, анализ безопасности данных и конфиденциальности, использование инструментов, таких как OWASP ZAP и Burp Suite.
11. **Что включает в себя тестовая документация? (ПК-4.1)**
- **Правильный ответ:** Тестовая документация включает тест-планы, тест-кейсы, чек-листы, отчеты о дефектах, а также итоговые отчеты по результатам тестирования.
12. **Какие задачи выполняет тест-менеджер? (УК-3.2)**
- **Правильный ответ:** Тест-менеджер координирует работу команды тестировщиков, разрабатывает тест-планы, управляет ресурсами, контролирует процесс тестирования и отвечает за качество тестирования.
13. **Что такое релизное тестирование? (ПК-3.2)**
- **Правильный ответ:** Релизное тестирование - это финальное тестирование, проводимое перед выпуском продукта. Оно включает проверку всех функций и стабильности системы для обеспечения готовности продукта к эксплуатации.
14. **Какой подход используется в тестировании на основе риска? (УК-2.2)**
- **Правильный ответ:** В тестировании на основе риска приоритетное внимание уделяется тестированию наиболее критичных функций, которые могут вызвать наибольшие проблемы при сбоях.
15. **Какую роль играет интеграционное тестирование в процессе разработки ПО? (ПК-3.1)**
- **Правильный ответ:** Интеграционное тестирование проверяет взаимодействие между модулями системы, обеспечивая корректную работу всех компонентов в составе системы.
16. **Какую роль выполняет тестировщик в команде? (УК-3.1)**
- **Правильный ответ:** Тестировщик отвечает за обнаружение и документирование дефектов, проверку функциональности системы, проведение тестов и отчетность о результатах тестирования.
17. **Какие инструменты используются для тестирования безопасности? (ПК-3.2)**
- **Правильный ответ:** Инструменты для тестирования безопасности включают OWASP ZAP, Burp Suite, Nessus и другие.
18. **Что такое CI/CD и как оно связано с тестированием? (ПК-4.2)**
- **Правильный ответ:** CI/CD (Continuous Integration/Continuous Deployment) - это методология, включающая непрерывную интеграцию и непрерывное развертывание. Она обеспечивает автоматизацию тестирования и развертывания, позволяя быстро выявлять и исправлять ошибки.
19. **Что включает в себя динамическое тестирование? (ПК-2.2)**
- **Правильный ответ:** Динамическое тестирование включает в себя проверку программного обеспечения в процессе его выполнения, включая функциональное тестирование, тестирование производительности и другие виды тестирования, проводимые в рабочем режиме системы.
20. **Какие преимущества предоставляет статическое тестирование? (УК-2.1)**

- **Правильный ответ:** Статическое тестирование позволяет выявлять ошибки на ранних стадиях разработки, не требуя выполнения кода. Оно помогает обнаружить синтаксические ошибки, нарушения стиля кода и другие дефекты до этапа тестирования.
- 21. **Что такое управление конфигурациями в контексте тестирования ПО? (ПК-2.1)**
 - **Правильный ответ:** Управление конфигурациями включает в себя контроль версий программного обеспечения, управление изменениями, контроль состояния тестовой среды и обеспечение консистентности тестовых данных.
- 22. **Какое значение имеет тестирование данных для управления качеством ПО? (ПК-4.2)**
 - **Правильный ответ:** Тестирование данных помогает выявить ошибки в данных, проверить корректность их обработки и обеспечить целостность данных. Оно играет ключевую роль в управлении качеством ПО, особенно при работе с большими объемами данных.
- 23. **Какое значение имеет методология Agile для тестирования ПО? (УК-2.2)**
 - **Правильный ответ:** Методология Agile требует тесного взаимодействия между разработчиками и тестировщиками, обеспечивая быстрое и гибкое реагирование на изменения требований. Она поддерживает непрерывное тестирование и интеграцию, что способствует повышению качества ПО.
- 24. **Что включает в себя системное тестирование? (ПК-2.2)**
 - **Правильный ответ:** Системное тестирование включает проверку всей системы в целом после завершения разработки всех модулей. Оно направлено на проверку взаимодействия всех компонентов системы и их соответствия требованиям.
- 25. **Какие задачи входят в обязанности тест-менеджера в рамках проекта? (УК-3.2)**
 - **Правильный ответ:** Тест-менеджер разрабатывает стратегию тестирования, координирует работу команды тестировщиков, управляет ресурсами и тестовыми данными, контролирует процесс тестирования и отвечает за отчетность по результатам тестирования.
- 26. **Какие инструменты используются для управления тестовыми данными? (ПК-4.1)**
 - **Правильный ответ:** Для управления тестовыми данными используются инструменты, такие как TestRail, HP ALM (Application Lifecycle Management), IBM Rational Quality Manager и другие.
- 27. **Какие задачи решаются при интеграционном тестировании и какие инструменты при этом используются? (ПК-3.1)**
 - **Правильный ответ:** Интеграционное тестирование проверяет взаимодействие между модулями системы, выявляя дефекты, связанные с интеграцией. Инструменты, используемые для интеграционного тестирования, включают JUnit, TestNG и другие фреймворки для модульного и интеграционного тестирования.

Критерии оценки ответов на вопросы

- **"Отлично" (5 баллов)**
 - **Критерии:**
 - Полное и точное объяснение вопроса.
 - Ответ включает все ключевые аспекты и детали.
 - Примеры, если требуются, приведены и правильно объяснены.
 - Ответ демонстрирует глубокое понимание темы.
- **"Хорошо" (4 балла)**
 - **Критерии:**
 - Корректное объяснение вопроса.
 - Ответ охватывает основные аспекты, но может отсутствовать незначительная деталь или пример.
 - Демонстрируется хорошее, но не полное понимание темы.

- **"Удовлетворительно" (3 балла)**
 - **Критерии:**
 - Общее представление о вопросе.
 - Ответ включает основные аспекты, но содержит неточности или пропуски.
 - Примеры, если требуются, могут отсутствовать или быть неверно объяснены.
 - Демонстрируется базовое понимание темы.
- **"Неудовлетворительно" (2 балла)**
 - **Критерии:**
 - Некорректное или неполное объяснение вопроса.
 - Отсутствие ключевых аспектов и деталей.
 - Примеры, если требуются, отсутствуют или приведены неверные.
 - Ответ демонстрирует недостаточное понимание темы.

2.3 Задачи на соответствие понятий для текущего контроля успеваемости в виде ВЛАДЕНИЙ

Правильные ответы расположены в таблицах друг напротив друга, во время тестирования предполагается что порядок данных в рамках каждого столбца будет случайным.

Задача 1. Соотнесите основные понятия с их определениями (УК-2.1, ПК-4.1)

Понятие	Определение
А - Тест-кейс	1 - Документ, описывающий входные данные, условия выполнения и ожидаемый результат.
В - Дефект	2 - Отклонение между ожидаемым и фактическим поведением программы.
С - Чек-лист	3 - Список проверок, используемых для обеспечения полноты и систематичности тестирования.
Д - Тест-план	4 - Документ, описывающий стратегию, объем, ресурсы и расписание тестирования.

Правильный ответ: А-1, В-2, С-3, Д-4

Задача 2. Соотнесите типы тестирования с их описанием (ПК-2.1, ПК-2.2)

Тип тестирования	Описание
А - Функциональное	1 - Проверка соответствия системы функциональным требованиям.
В - Регрессионное	2 - Проверка исправления дефектов и отсутствие новых ошибок после изменений.
С - Нагрузочное	3 - Проверка поведения системы под высокой нагрузкой.
Д - Интеграционное	4 - Проверка взаимодействия между модулями системы.

Правильный ответ: А-1, В-2, С-3, Д-4

Задача 3. Соотнесите инструменты тестирования с их назначением (ПК-2.2, ПК-3.2)

Инструмент	Назначение
A - Selenium	1 - Инструмент для автоматизированного тестирования веб-приложений.
B - JIRA	2 - Система управления дефектами и проектами.
C - JUnit	3 - Фреймворк для модульного тестирования на языке Java.
D - OWASP ZAP	4 - Инструмент для тестирования безопасности веб-приложений.

Правильный ответ: A-1, B-2, C-3, D-4

Задача 4. Соотнесите методологии тестирования с их характеристиками (УК-2.2, ПК-4.1)

Методология	Характеристика
A - Waterfall	1 - Линейная последовательность этапов разработки и тестирования.
B - Agile	2 - Итеративный и инкрементальный подход к разработке и тестированию, тесное взаимодействие команды.
C - V-Model	3 - Расширение Waterfall-модели, включающее валидацию и верификацию на каждом этапе.
D - DevOps	4 - Культура и набор практик, направленных на автоматизацию и интеграцию процессов разработки и тестирования.

Правильный ответ: A-1, B-2, C-3, D-4

Задача 5. Соотнесите термины, связанные с качеством ПО, с их определениями (УК-3.1, ПК-4.2)

Термин	Определение
A - Надежность	1 - Способность ПО выполнять требуемые функции при определенных условиях в течение заданного времени.
B - Поддерживаемость	2 - Легкость, с которой ПО может быть модифицировано для исправления дефектов или добавления новых функций.
C - Переносимость	3 - Способность ПО быть использованным в различных аппаратных или программных средах.
D - Эффективность	4 - Степень, с которой ПО использует минимальные ресурсы для выполнения своих функций.

Правильный ответ: A-1, B-2, C-3, D-4

Задача 6. Соотнесите типы тестовой документации с их описанием (ПК-4.1, ПК-4.2)

Тип документации	Описание
А - Тестовая стратегия	1 - Документ, определяющий общий подход к тестированию, включая цели, методы и критерии завершения.
В - Тестовый план	2 - Документ, описывающий конкретные действия, необходимые для выполнения тестирования.
С - Тест-кейс	3 - Документ, описывающий набор условий и шагов для проверки определенной функциональности.
Д - Отчет о дефекте	4 - Документ, описывающий найденный дефект, условия его проявления и шаги для его воспроизведения.

Правильный ответ: А-1, В-2, С-3, Д-4

Критерии оценки выполнения задач на соответствие понятий

- **Правильность соответствий:**
 - **Отлично (5):** Все соответствия выполнены правильно.
 - **Хорошо (4):** 1 ошибка в соответствиях.
 - **Удовлетворительно (3):** 2 ошибки в соответствиях.
 - **Неудовлетворительно (2):** 3 и более ошибок в соответствиях.

3. Оценочные материалы для проведения промежуточной аттестации обучающихся (студентов)

3.1 Вопросы для проведения промежуточной аттестации в форме ЭКЗАМЕНА

1. Опишите процесс тестирования программного обеспечения и его основные этапы. (УК-2.1)

- **Правильный ответ:** Тестирование программного обеспечения включает несколько этапов: планирование, анализ требований, разработка тест-кейсов, подготовка тестовой среды, выполнение тестов, управление дефектами и отчетность. На этапе планирования определяется стратегия тестирования, разрабатывается тестовый план, включая ресурсы и график. Анализ требований позволяет понять, какие функции и характеристики будут проверяться. Разработка тест-кейсов включает создание конкретных сценариев для тестирования. Подготовка тестовой среды включает настройку аппаратных и программных средств для выполнения тестов. На этапе выполнения тестов запускаются тестовые сценарии, результаты которых фиксируются и анализируются. Управление дефектами включает выявление, документирование и устранение ошибок. В конце составляется отчет по результатам тестирования.

2. Какие виды тестирования вы знаете и каковы их особенности? (ПК-2.1)

- **Правильный ответ:** Виды тестирования включают функциональное, нефункциональное, регрессионное, интеграционное, системное, приемочное, нагрузочное и тестирование безопасности. Функциональное тестирование проверяет соответствие системы функциональным требованиям. Нефункциональное тестирование оценивает производительность, надежность, удобство использования и другие характеристики. Регрессионное тестирование проверяет, не появились ли новые ошибки после внесения изменений. Интеграционное тестирование проверяет взаимодействие

между модулями. Системное тестирование проверяет систему в целом. Приемочное тестирование проводится для подтверждения готовности системы к использованию. Нагрузочное тестирование проверяет систему под высокой нагрузкой. Тестирование безопасности выявляет уязвимости и угрозы в системе.

3. Каковы основные цели и задачи тестового плана? (УК-2.1)

- **Правильный ответ:** Тестовый план служит для определения стратегии и подходов к тестированию, а также для координации и контроля процесса тестирования. Он включает цели тестирования, тестируемые объекты, задачи, необходимые ресурсы, график, критерии завершения тестирования и отчетность. Основная цель тестового плана - обеспечить систематическое и полное тестирование всех аспектов программного обеспечения. Он помогает определить приоритеты тестирования, распределить ресурсы, установить временные рамки и следить за выполнением тестовых задач.

4. Объясните разницу между тестированием "черного ящика" и "белого ящика". (ПК-2.1)

- **Правильный ответ:** Тестирование "черного ящика" проверяет функциональность системы без знания ее внутренней структуры. Тестировщик разрабатывает тест-кейсы на основе требований и спецификаций, не имея доступа к исходному коду. Этот метод позволяет проверить, как система выполняет свои функции с точки зрения пользователя. Тестирование "белого ящика" предполагает знание внутренней структуры и логики программы. Тестировщик разрабатывает тест-кейсы, основываясь на анализе исходного кода, и проверяет внутренние логические пути, структуры данных и граничные условия. Этот метод позволяет глубже проанализировать работу системы и выявить ошибки, которые могут быть незаметны при тестировании "черного ящика".

5. Каковы преимущества и недостатки автоматизированного тестирования? (ПК-2.2)

- **Правильный ответ:** Преимущества автоматизированного тестирования включают повышение эффективности и точности тестирования, сокращение времени на выполнение повторяющихся тестов, возможность выполнения тестов в любое время и на различных платформах, а также повышение точности и повторяемости тестов. Недостатки включают высокие первоначальные затраты на разработку автоматизированных тестов, необходимость в поддержке и обновлении тестов при изменении программного обеспечения, а также ограниченность в тестировании сложных пользовательских взаимодействий и графических интерфейсов.

6. Какие инструменты автоматизированного тестирования вы знаете и каковы их основные функции? (ПК-2.2)

- **Правильный ответ:** Популярные инструменты автоматизированного тестирования включают Selenium, JUnit, TestNG, Jenkins и Appium. Selenium используется для автоматизированного функционального тестирования веб-приложений. JUnit и TestNG применяются для модульного тестирования на языке Java. Jenkins обеспечивает непрерывную интеграцию и автоматизацию выполнения тестов. Appium используется для автоматизированного тестирования мобильных приложений. Эти инструменты помогают автоматизировать процесс тестирования, сократить время и затраты на тестирование, а также повысить точность и повторяемость тестов.

7. Что такое тестовый сценарий и как его правильно разработать? (ПК-4.1)

- **Правильный ответ:** Тестовый сценарий - это документ, описывающий последовательность действий для проверки определенной функциональности или характеристики системы. При разработке тестового сценария необходимо учитывать требования и спецификации, определить входные данные, предусмотреть шаги выполнения и зафиксировать ожидаемые результаты. Тестовый сценарий должен быть четким, понятным и воспроизводимым. Он помогает систематизировать тестирование, обеспечить полное покрытие тестируемых функций и облегчить обнаружение и устранение дефектов.

8. Объясните процесс нагрузочного тестирования и его важность для обеспечения качества ПО. (ПК-3.2)

- **Правильный ответ:** Нагрузочное тестирование - это процесс проверки поведения системы под высокой нагрузкой для определения ее пределов и производительности. Оно включает моделирование большого количества пользователей или запросов к системе и измерение времени отклика, пропускной способности и устойчивости. Нагрузочное тестирование помогает выявить узкие места и слабые места системы, которые могут привести к сбоям или снижению производительности при высокой нагрузке. Оно также позволяет оптимизировать ресурсы и улучшить масштабируемость системы.

9. Что такое дефект-менеджмент и какие этапы он включает? (УК-3.1)

- **Правильный ответ:** Дефект-менеджмент - это процесс выявления, анализа, документирования и устранения дефектов в программном обеспечении. Этапы дефект-менеджмента включают обнаружение дефекта, его регистрацию в системе управления дефектами, анализ и приоритизацию, назначение ответственного за устранение, исправление дефекта, повторное тестирование для проверки исправления и закрытие дефекта. Этот процесс помогает систематизировать работу с дефектами, улучшить качество ПО и предотвратить повторное возникновение ошибок.

10. Каковы методы тестирования безопасности ПО и какие инструменты для этого используются? (ПК-2.1)

- **Правильный ответ:** Методы тестирования безопасности включают анализ кода на наличие уязвимостей, тестирование на проникновение, анализ безопасности данных и конфиденциальности, использование инструментов для автоматизированного тестирования безопасности. Инструменты для тестирования безопасности включают OWASP ZAP, Burp Suite, Nessus и другие. Эти методы и инструменты помогают выявлять уязвимости, предотвращать атаки и обеспечивать защиту данных и конфиденциальность.

11. Что включает в себя тестовая документация и как она используется в процессе тестирования? (ПК-4.1)

- **Правильный ответ:** Тестовая документация включает тест-планы, тест-кейсы, чек-листы, отчеты о дефектах, итоговые отчеты по результатам тестирования. Тест-планы описывают стратегию и подходы к тестированию, тест-кейсы содержат конкретные сценарии тестирования, чек-листы помогают систематизировать процесс тестирования, отчеты о дефектах фиксируют выявленные ошибки, а итоговые отчеты подводят итоги тестирования. Тестовая документация используется для планирования, выполнения, контроля и анализа тестирования, а также для передачи знаний и обеспечения прозрачности процесса.

12. Какие задачи выполняет тест-менеджер в команде тестировщиков? (УК-3.2)

- **Правильный ответ:** Тест-менеджер координирует работу команды тестировщиков, разрабатывает тест-планы, управляет ресурсами и тестовыми данными, контролирует процесс тестирования и отвечает за отчетность по результатам тестирования. Он также проводит анализ рисков, определяет приоритеты тестирования, взаимодействует с другими командами и заинтересованными сторонами, решает возникающие проблемы и обеспечивает качество и своевременность выполнения тестов.

13. Объясните процесс релизного тестирования и его значение для успешного выпуска продукта. (ПК-3.2)

- **Правильный ответ:** Релизное тестирование - это финальное тестирование, проводимое перед выпуском продукта. Оно включает проверку всех функций и стабильности системы, выполнение регрессионного тестирования для выявления новых дефектов, анализ производительности и безопасности, подготовку документации и отчетов. Релизное тестирование помогает убедиться в готовности продукта к эксплуатации, выявить и устранить последние дефекты, обеспечить соответствие требованиям и ожиданиям пользователей. Это важный этап, который влияет на успех продукта на рынке и удовлетворенность пользователей.

14. Как применяется тестирование на основе риска и какие преимущества оно предоставляет? (УК-2.2)

- **Правильный ответ:** Тестирование на основе риска предполагает приоритетное внимание к тестированию наиболее критичных функций и компонентов системы, которые могут вызвать наибольшие проблемы при сбоях. Оно включает идентификацию и анализ рисков, определение приоритетов тестирования, разработку тест-кейсов с учетом рисков. Преимущества тестирования на основе риска включают более эффективное использование ресурсов, сокращение времени на тестирование, повышение вероятности выявления критичных дефектов и улучшение качества и надежности системы.

15. Объясните роль интеграционного тестирования в процессе разработки ПО и его основные задачи. (ПК-3.1)

- **Правильный ответ:** Интеграционное тестирование проверяет взаимодействие между модулями системы, обеспечивая корректную работу всех компонентов в составе системы. Основные задачи интеграционного тестирования включают выявление дефектов, связанных с интеграцией модулей, проверку совместимости интерфейсов, тестирование передачи данных между модулями и оценку производительности системы при интеграции. Интеграционное тестирование помогает выявить проблемы на ранних стадиях разработки, улучшить взаимодействие компонентов и обеспечить стабильную работу системы в целом.

16. Какую роль выполняет тестировщик в команде и какие навыки для этого необходимы? (УК-3.1)

- **Правильный ответ:** Тестировщик отвечает за обнаружение и документирование дефектов, проверку функциональности системы, проведение тестов и отчетность о результатах тестирования. Необходимые навыки включают внимание к деталям, аналитическое мышление, знание методов и инструментов тестирования, умение работать в команде, навыки коммуникации и документирования. Тестировщик играет ключевую роль в обеспечении качества ПО и выявлении ошибок на ранних стадиях разработки.

17. Какие инструменты используются для тестирования безопасности и как они применяются? (ПК-3.2)

- **Правильный ответ:** Инструменты для тестирования безопасности включают OWASP ZAP, Burp Suite, Nessus и другие. OWASP ZAP используется для автоматизированного поиска уязвимостей в веб-приложениях. Burp Suite помогает анализировать и модифицировать HTTP-запросы и ответы для выявления уязвимостей. Nessus сканирует сети и системы на наличие уязвимостей и слабых мест. Эти инструменты применяются для автоматизации тестирования безопасности, выявления и устранения уязвимостей, повышения защищенности систем и данных.

18. Объясните концепцию CI/CD и ее влияние на процесс тестирования ПО. (ПК-4.2)

- **Правильный ответ:** CI/CD (Continuous Integration/Continuous Deployment) - это методология, включающая непрерывную интеграцию и непрерывное развертывание. CI/CD обеспечивает автоматизацию выполнения тестов и развертывания кода, позволяет быстро выявлять и исправлять ошибки, сокращает время на выпуск новых версий продукта. CI/CD включает автоматизированные сборки, тестирование, деплоймент и мониторинг. Это позволяет улучшить качество ПО, сократить время разработки и развертывания, повысить гибкость и скорость реагирования на изменения требований.

19. Что включает в себя динамическое тестирование и какие виды тестов к нему относятся? (ПК-2.2)

- **Правильный ответ:** Динамическое тестирование включает в себя проверку программного обеспечения в процессе его выполнения. К видам динамического тестирования относятся функциональное тестирование, регрессионное тестирование, интеграционное тестирование, системное тестирование, приемочное тестирование,

нагрузочное тестирование и тестирование безопасности. Динамическое тестирование позволяет проверить поведение системы в реальных условиях, выявить дефекты, связанные с выполнением кода, и обеспечить соответствие системы функциональным и нефункциональным требованиям.

20. Какие преимущества предоставляет статическое тестирование и как оно проводится? (УК-2.1)

- **Правильный ответ:** Статическое тестирование позволяет выявлять ошибки на ранних стадиях разработки, не требуя выполнения кода. Оно включает анализ исходного кода, инспекции, ревью и анализ требований и документации. Преимущества статического тестирования включают возможность раннего обнаружения дефектов, снижение затрат на исправление ошибок, улучшение качества кода и документации. Оно проводится с помощью инструментов для статического анализа кода и вручную, путем ревью и инспекций.

21. Что такое управление конфигурациями в контексте тестирования ПО и какие задачи оно решает? (ПК-2.1)

- **Правильный ответ:** Управление конфигурациями включает в себя контроль версий программного обеспечения, управление изменениями, контроль состояния тестовой среды и обеспечение консистентности тестовых данных. Основные задачи управления конфигурациями - это отслеживание изменений в коде и документации, управление релизами, поддержка тестовой среды в актуальном состоянии, предотвращение конфликтов и ошибок, связанных с изменениями, и обеспечение воспроизводимости тестов. Это помогает систематизировать процесс разработки и тестирования, улучшить качество и стабильность системы.

22. Какое значение имеет тестирование данных для управления качеством ПО и какие методы для этого используются? (ПК-4.2)

- **Правильный ответ:** Тестирование данных помогает выявить ошибки в данных, проверить корректность их обработки и обеспечить целостность данных. Методы тестирования данных включают проверку валидности данных, тестирование граничных значений, тестирование на основе статистических методов и анализ данных. Тестирование данных играет ключевую роль в управлении качеством ПО, особенно при работе с большими объемами данных и критическими системами, где корректность данных имеет важное значение для работы системы.

23. Как методология Agile влияет на процесс тестирования ПО и какие преимущества она предоставляет? (УК-2.2)

- **Правильный ответ:** Методология Agile требует тесного взаимодействия между разработчиками и тестировщиками, обеспечивая быстрое и гибкое реагирование на изменения требований. Agile поддерживает непрерывное тестирование и интеграцию, что способствует повышению качества ПО. Преимущества Agile включают возможность быстрой адаптации к изменениям, улучшение коммуникации и сотрудничества в команде, повышение прозрачности процесса разработки, сокращение времени на выпуск новых версий продукта и улучшение качества и удовлетворенности пользователей.

24. Что включает в себя системное тестирование и какие цели оно преследует? (ПК-2.2)

- **Правильный ответ:** Системное тестирование включает проверку всей системы в целом после завершения разработки всех модулей. Цели системного тестирования - это проверка соответствия системы требованиям и спецификациям, проверка взаимодействия всех компонентов системы, оценка производительности, надежности и безопасности системы. Системное тестирование позволяет убедиться в готовности системы к эксплуатации, выявить и устранить дефекты, улучшить качество и стабильность системы.

25. Какие задачи входят в обязанности тест-менеджера в рамках проекта и какие навыки для этого необходимы? (УК-3.2)

- **Правильный ответ:** Тест-менеджер разрабатывает стратегию тестирования, координирует работу команды тестировщиков, управляет ресурсами и тестовыми данными, контролирует процесс тестирования и отвечает за отчетность по результатам тестирования. Необходимые навыки включают лидерство, умение планировать и организовывать работу, знание методов и инструментов тестирования, аналитическое мышление, навыки коммуникации и управления командой. Тест-менеджер играет ключевую роль в обеспечении качества ПО и успешной реализации проекта.

26. Какие инструменты используются для управления тестовыми данными и как они применяются? (ПК-4.1)

- **Правильный ответ:** Для управления тестовыми данными используются инструменты, такие как TestRail, HP ALM (Application Lifecycle Management), IBM Rational Quality Manager и другие. Эти инструменты помогают организовать и систематизировать работу с тестовыми данными, обеспечить консистентность и актуальность данных, автоматизировать процесс создания и управления тестовыми данными, улучшить контроль и отчетность. Они применяются для создания, хранения, управления и анализа тестовых данных, а также для обеспечения воспроизводимости и точности тестов.

Критерии оценки ответов на экзамене

- **"Отлично" (5 баллов)**
 - **Критерии:**
 - Полное и точное объяснение вопроса.
 - Ответ включает все ключевые аспекты и детали.
 - Примеры, если требуются, приведены и правильно объяснены.
 - Ответ демонстрирует глубокое понимание темы.
- **"Хорошо" (4 балла)**
 - **Критерии:**
 - Корректное объяснение вопроса.
 - Ответ охватывает основные аспекты, но может отсутствовать незначительная деталь или пример.
 - Демонстрируется хорошее, но не полное понимание темы.
- **"Удовлетворительно" (3 балла)**
 - **Критерии:**
 - Общее представление о вопросе.
 - Ответ включает основные аспекты, но содержит неточности или пропуски.
 - Примеры, если требуются, могут отсутствовать или быть неверно объяснены.
 - Демонстрируется базовое понимание темы.
- **"Неудовлетворительно" (2 балла)**
 - **Критерии:**
 - Некорректное или неполное объяснение вопроса.
 - Отсутствие ключевых аспектов и деталей.
 - Примеры, если требуются, отсутствуют или приведены неверные.
 - Ответ демонстрирует недостаточное понимание темы.