

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**



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

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
Б1.В.05 «Технология разработки программного обеспечения»**

Направление подготовки: 09.03.02 Информационные системы и технологии

**Направленность (профиль) программы бакалавриата
«Информационные технологии»**

ОДОБРЕНО

Ученым советом ЧУВО «ВШП»

Протокол заседания

№01-02/24 от 15 мая 2025 г.



Тверь, 2025

Рабочая программа учебной дисциплины **Б1.В.05 Технология разработки программного обеспечения**, компонента основной профессиональной образовательной программы высшего образования — программы бакалавриата по направлению подготовки **09.03.02 Информационные системы и технологии** направленность (профиль) «**Информационные технологии**», 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 технологии.

2. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ

Целью изучения дисциплины «Технология разработки программного обеспечения» является формирование у обучающихся теоретических и практических навыков по применению принципов создания приложений, поддерживающих требования интерфейса операционной среды ОС MS Windows, типовых приемов организации и конструирования пакетов программ сложной структуры, этапов процесса проектирования программного обеспечения, создание прикладных программ с высокой степенью автоматизации управления.

Для этого в рамках дисциплины решаются следующие задачи:

- изучение методов проектирования и производства программного продукта, принципов построения, структуры и приемы работы с инstrumentальными средствами, поддерживающими разработку программного обеспечения; методов организации работы в коллективах разработчиков программного обеспечения;
- формирование навыков проектирования, реализации, оценки качества и анализа эффективности 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 технологии.

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-1, УК-2, ПК-1, 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-1	Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход	УК-1.1 Способен осуществлять поиск и критический анализ информации	Знать: <ul style="list-style-type: none">- Основные источники информации для разработки программного обеспечения (далее ПО).- Методы критического анализа информации. Уметь:

	для решения поставленных задач		<ul style="list-style-type: none"> - Находить актуальную информацию в специализированных источниках. - Анализировать и оценивать достоверность и релевантность информации. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками эффективного поиска информации. - Навыками критического мышления при анализе информации.
	УК-1.2 Способен применять системный подход для решения поставленных задач		<p>Знать:</p> <ul style="list-style-type: none"> - Принципы системного подхода в разработке ПО. - Основные этапы системного анализа и проектирования. <p>Уметь:</p> <ul style="list-style-type: none"> - Применять системный подход для анализа задач. - Разрабатывать системные решения для поставленных задач. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками системного анализа проблем. - Навыками интеграции различных элементов системы в единое целое.
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	<p>УК-2.1 Способен определять круг задач в рамках поставленной цели</p> <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"> - Методы декомпозиции целей на задачи. - Принципы приоритизации задач. <p>Уметь:</p> <ul style="list-style-type: none"> - Определять и формулировать задачи на основе целей проекта. - Устанавливать приоритеты для выполнения задач. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками декомпозиции целей на задачи. - Навыками управления задачами и ресурсами. <p>Знать:</p> <ul style="list-style-type: none"> - Основные правовые нормы в области информационных технологий (далее ИТ). - Принципы оценки и управления ресурсами. <p>Уметь:</p> <ul style="list-style-type: none"> - Оценивать доступные ресурсы и ограничения. - Выбирать наилучшие способы решения задач с учетом правовых норм. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками оценки ресурсов и ограничений. - Навыками принятия оптимальных решений в рамках правовых норм.
ПК-1	Разработка и отладка программного кода	ПК-1.1 Способен разрабатывать программный код	<p>Знать:</p> <ul style="list-style-type: none"> - Основные языки программирования для разработки ПО. - Принципы написания качественного кода. <p>Уметь:</p> <ul style="list-style-type: none"> - Писать программный код на одном из изучаемых языков. - Разрабатывать алгоритмы и структуры данных для решения задач. <p>Владеть:</p>

			<ul style="list-style-type: none"> - Навыками написания и документирования кода. - Навыками использования инструментов разработки.
		ПК-1.2 Способен отлаживать программный код	<p>Знать:</p> <ul style="list-style-type: none"> - Основные методы и инструменты отладки кода. - Принципы устранения ошибок и багов в коде. <p>Уметь:</p> <ul style="list-style-type: none"> - Использовать отладчики и логи для выявления ошибок. - Исправлять выявленные ошибки и оптимизировать код. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками отладки и тестирования кода. - Навыками применения инструментов для отладки.
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 кода. - Основные виды тестов (модульные, интеграционные, системные). <p>Уметь:</p> <ul style="list-style-type: none"> - Писать и выполнять тесты для проверки кода. - Использовать тестовые фреймворки для автоматизации тестирования. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками написания и выполнения тестов. - Навыками применения инструментов для тестирования.
		ПК-2.2 Способен рефакторить программный код	<p>Знать:</p> <ul style="list-style-type: none"> - Принципы и методы рефакторинга. - Основные техники улучшения структуры кода. <p>Уметь:</p> <ul style="list-style-type: none"> - Анализировать код на предмет улучшений. - Применять техники рефакторинга для повышения качества кода. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками рефакторинга кода. - Навыками оценки и улучшения структуры кода.
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 интеграции модулей и компонентов. - Методы и инструменты для интеграции. <p>Уметь:</p> <ul style="list-style-type: none"> - Интегрировать модули и компоненты в единое приложение. - Проверять корректность интеграции и совместимость компонентов. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками интеграции программных модулей. - Навыками использования инструментов для интеграции.

		ПК-3.2 Способен проверять работоспособность выпусков программного продукта	<p>Знать:</p> <ul style="list-style-type: none"> - Методы и инструменты для проверки выпусков ПО. - Принципы контроля качества выпусков. <p>Уметь:</p> <ul style="list-style-type: none"> - Проводить тестирование и верификацию выпусков ПО. - Оценивать стабильность и производительность программного продукта. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками тестирования и верификации выпусков ПО. - Навыками применения инструментов контроля качества.
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"> - Принципы и методы анализа требований. - Техники документирования требований. <p>Уметь:</p> <ul style="list-style-type: none"> - Собирать и анализировать требования к ПО. - Документировать требования в виде спецификаций. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками анализа и сбора требований. - Навыками создания и поддержки документации по требованиям.
		ПК-4.2 Способен проектировать программное обеспечение	<p>Знать:</p> <ul style="list-style-type: none"> - Принципы и методы проектирования ПО. - Основные подходы к архитектурному дизайну. <p>Уметь:</p> <ul style="list-style-type: none"> - Разрабатывать архитектуру и дизайн ПО. - Использовать UML и другие инструменты для проектирования. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками проектирования программного обеспечения. - Навыками применения инструментов для проектирования.

5. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

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

Общая трудоемкость учебной дисциплины составляет 4 зачетных единицы, 144 часа, включая все формы контактной и самостоятельной работы обучающихся.

Объем дисциплины по учебному плану составляет –

4 зачетных единицы = 144 академических часа.

Контактная работа обучающегося (студенты) с научно-педагогическим работником организации (всего) - 48 академических часов,

в том числе:

Лекционные занятия (Лек.) - 18 академических часов,

Практические занятия (Пр.) - 28 академических часов,

Консультации (Конс.) - 2 академических часа.

Самостоятельная работа обучающегося (студента):

Самостоятельная работа (СР) - 51 академический час,

Текущий контроль успеваемости

и промежуточно-заочная аттестация обучающегося (студента):

Часы на контроль

- 45 академических часов.

Таблица 2. Объём дисциплины

№ п/п	Раздел/тема дисциплины	Виды учебной деятельности, включая самостоятельную работу обучающихся (студентов), и трудоёмкость (в ак. часах)			Самосто ятельна я работа	Коды формируемых компетенций		
		Виды учебных занятий по дисциплине						
		Лек.	Пр.	Конс.				
5 семестр / 3 курс								
1	Тема 1. Введение в технологии разработки ПО	2	2	-	6	УК-1.1, УК-1.2		
2	Тема 2. Жизненный цикл разработки ПО	2	2	-	6	УК-2.1, УК-2.2		
3	Тема 3. Управление версиями и семантическое версионирование	2	2	-	6	ПК-3.1, ПК-3.2		
4	Тема 4. Git и методы работы с ним	2	2	-	6	ПК-1.1, ПК-3.1		
5	Тема 5. Управление задачами и релизами	2	2	-	6	ПК-4.1, ПК-4.2		
6	Тема 6. Подготовка к зачету		4	-	9	УК-1.1, УК-1.2, УК-2.1, УК-2.2, ПК-1.1, ПК-3.1, ПК-3.2, ПК-4.1, ПК-4.2		
6 семестр / 3 курс								
7	Тема 7. Основы CI/CD	1	2	-	2	ПК-2.1, ПК-2.2		
8	Тема 8. Командная работа и распределение ролей	2	2	-	2	УК-3.1, УК-3.2		
9	Тема 9. Методологии разработки ПО	2	2	-	2	УК-1.2, ПК-4.2		
10	Тема 10. Планирование и контроль проектов	1	2	-	2	УК-2.1, ПК-4.1		
11	Тема 11. Управление конфигурацией и релизами	2	2	-	2	ПК-3.1, ПК-3.2		
12	Тема 12. Итоговое занятие и подготовка к экзамену		4	2	2	УК-1.1, УК-1.2, УК-2.1, УК-2.2, УК-3.1, УК-3.2, ПК-1.1, ПК-2.1, ПК-2.2, ПК-3.1, ПК-3.2, ПК-4.1, ПК-4.2		
ИТОГО аудиторных часов/СР:		5-6 семестры/ 3 курс	48 ак. часов		51 ак. часа	-		
Часы на контроль		5-6 семестры/ 3 курс	9 ак. час (форма промежуточной аттестации – зачет – 5 семестр) 36 ак. час (форма промежуточной аттестации – экзамен – 6 семестр)					
ВСЕГО ак. часов:		5-6 семестры/ 3 курс	144 академических часов					

5.2. Тематическое содержание дисциплины

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

5 семестр

Тема 1: Введение в технологии разработки ПО

Основные подходы и методологии разработки ПО. Понятие и цели разработки программного обеспечения.

Тема 2: Жизненный цикл разработки ПО

Модели жизненного цикла ПО: водопадная, спиральная, Agile, Scrum. Основные этапы жизненного цикла.

Тема 3: Управление версиями и семантическое версионирование

Основы управления версиями. Семантическое версионирование. Использование тегов и релизов.

Тема 4: Git и методы работы с ним

Основы работы с Git. Различные стратегии ветвления и слияния, такие как Git-flow и GitHub Flow. Использование Git для управления проектами.

Тема 5: Управление задачами и релизами

Инструменты и методы управления задачами и релизами. Использование систем трекинга задач (JIRA, Trello).

Тема 6: Подготовка к зачету

Повторение и обобщение пройденного материала. Решение типовых задач и вопросов для подготовки к зачету.

6 семестр

Тема 7: Основы CI/CD

Понятие непрерывной интеграции и непрерывного развертывания. Основные инструменты и методы CI/CD.

Тема 8: Командная работа и распределение ролей

Организация работы в команде. Роли и ответственности в команде разработчиков.

Тема 9: Методологии разработки ПО

Детальное изучение методологий Agile и Scrum. Практики и артефакты.

Тема 10: Планирование и контроль проектов

Методы планирования и контроля проектов. Использование диаграммы Ганта и других инструментов.

Тема 11: Управление конфигурацией и релизами

Принципы управления конфигурацией. Подготовка и выпуск релизов.

Тема 12: Итоговое занятие и подготовка к экзамену

Повторение и обобщение пройденного материала. Решение типовых задач и вопросов для подготовки к экзамену. Обсуждение практических аспектов разработки и публикации программного обеспечения.

5.2.1 Содержание практических занятий

Таблица 3

№ п/п	Наименование темы (раздела) дисциплины	Содержание практического занятия
5 семестр		
1	Практическое занятие 1: Введение в технологии разработки ПО	Задание: Изучение основных подходов и методологий разработки ПО. Примеры различных методологий. Цель: Понять различные подходы к разработке ПО и их применимость в реальных проектах.
2	Практическое занятие 2:	Задание: Изучение моделей жизненного цикла ПО. Сравнение различных

	Жизненный цикл разработки ПО	моделей на примерах. Цель: Понять основные этапы жизненного цикла ПО и различия между моделями.
3	Практическое занятие 3: Управление версиями и семантическое версионирование	Задание: Практическое применение семантического версионирования на примере проекта. Цель: Освоить принципы управления версиями и семантического версионирования.
4	Практическое занятие 4: Git и методы работы с ним	Задание: Работа с Git: создание репозитория, ветвление, слияние, использование Git-flow. Цель: Освоить основные команды Git и стратегии работы с ветками.
5	Практическое занятие 5: Управление задачами и релизами	Задание: Создание и управление задачами в системе трекинга (JIRA, Trello). Планирование релизов. Цель: Научиться эффективно управлять задачами и релизами в проекте.
6	Практическое занятие 6: Подготовка к зачету	Задание: Повторение пройденного материала, решение типовых задач и вопросов для подготовки к зачету. Цель: Повторить и обобщить пройденный материал для успешной сдачи зачета.

6 семестр

7	Практическое занятие 7: Основы CI/CD	Задание: Настройка и использование CI/CD на примере Jenkins или GitHub Actions. Цель: Освоить основные инструменты и методы CI/CD.
8	Практическое занятие 8: Командная работа и распределение ролей	Задание: Организация работы команды: назначение ролей, распределение задач. Цель: Научиться эффективно работать в команде, распределять роли и задачи.
9	Практическое занятие 9: Методологии разработки ПО	Задание: Применение методологий Agile и Scrum в командной работе. Цель: Понять и применять практики и артефакты Agile и Scrum.
10	Практическое занятие 10: Планирование и контроль проектов	Задание: Создание и управление проектами с использованием диаграммы Ганта и других инструментов. Цель: Освоить методы планирования и контроля проектов.
11	Практическое занятие 11: Управление конфигурацией и релизами	Задание: Настройка и управление конфигурацией и релизами на примере проекта. Цель: Понять принципы управления конфигурацией и релизами.
12	Практическое занятие 12: Итоговое занятие и подготовка к экзамену	Задание: Повторение и обобщение пройденного материала, решение типовых задач и вопросов для подготовки к экзамену. Цель: Подготовиться к экзамену, обсудить практические аспекты разработки и публикации ПО.

5.2.2 Содержание самостоятельной работы

Таблица 4

№ п/п	Наименование темы (раздела) дисциплины	Содержание самостоятельной работы	Форма контроля
5 семестр			
1	Самостоятельная работа 1: Изучение основных подходов и методологий разработки ПО	Задание: Прочитать рекомендованные главы из учебников по разработке ПО. Написать краткий обзор различных методологий. Цель: Понять различные подходы к разработке ПО и их применимость в реальных проектах.	Ответы на контрольные вопросы.
2	Самостоятельная работа 2: Изучение моделей жизненного цикла ПО	Задание: Подготовить сравнительный анализ различных моделей жизненного цикла ПО. Цель: Освоить модели жизненного цикла ПО и научиться выбирать наиболее подходящую для конкретного проекта.	Выполнение практического задания.
3	Самостоятельная работа 3: Управление версиями и семантическое версионирование	Задание: Изучить материалы по семантическому версионированию. Применить принципы к учебному проекту. Цель: Освоить семантическое версионирование и научиться применять его на практике.	Выполнение практического задания.

4	Самостоятельная работа 4: Практическая работа с Git	Задание: Изучить основные команды Git и выполнить практическое задание по ветвлению и слиянию. Цель: Понять основные принципы работы с Git и освоить базовые команды.	Выполнение практического задания.
5	Самостоятельная работа 5: Управление задачами и релизами	Задание: Ознакомиться с системами трекинга задач (JIRA, Trello). Выполнить задание по созданию и управлению задачами. Цель: Научиться использовать системы трекинга задач для управления проектами.	Выполнение практического задания.
6	Самостоятельная работа 6: Подготовка к зачету	Задание: Повторение пройденного материала, решение типовых задач и вопросов для подготовки к зачету. Цель: Повторить и обобщить пройденный материал для успешной сдачи зачета.	Тест.
6 семестр			
7	Самостоятельная работа 7: Изучение основ CI/CD	Задание: Прочитать материалы по CI/CD. Выполнить настройку простого CI/CD пайплайна. Цель: Освоить принципы CI/CD и научиться настраивать CI/CD пайплайны.	Выполнение практического задания.
8	Самостоятельная работа 8: Организация работы в команде	Задание: Изучить принципы организации командной работы и распределения ролей. Изучение командной работы и распределения ролей Цель: Понять принципы организации командной работы и научиться эффективно распределять роли.	Выполнение практического задания.
9	Самостоятельная работа 9: Применение методологий Agile и Scrum	Задание: Изучить материалы по методологиям Agile и Scrum. Описать примеры применения этих методологий в реальных проектах. Цель: Освоить методологии Agile и Scrum и понять их практическое применение.	Ответы на контрольные вопросы.
10	Самостоятельная работа 10: Планирование и контроль проектов	Задание: Изучить методы планирования и контроля проектов. Выполнить практическое задание по созданию диаграммы Ганта. Цель: Научиться планировать и контролировать проекты с использованием различных инструментов.	Выполнение практического задания.
11	Самостоятельная работа 11: Управление конфигурацией и релизами	Задание: Прочитать материалы по управлению конфигурацией и релизами. Применить полученные знания на примере проекта. Цель: Понять принципы управления конфигурацией и релизами и научиться применять их на практике.	Выполнение практического задания.
12	Самостоятельная работа 12: Итоговое занятие и подготовка к экзамену	Задание: Повторение и обобщение пройденного материала, решение типовых задач и вопросов для подготовки к экзамену. Цель: Повторить и обобщить пройденный материал для успешной сдачи экзамена.	Экзамен.

6.Оценочные материалы по дисциплине

Оценочные материалы находятся в документе «Оценочные материалы по дисциплине «Технология разработки 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 приводят 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;

б) 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 технологии»), 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 технологии»), 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 технологии»), 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 Институтом 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 технологии»), 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. Мартин Р., Чистая архитектура. Искусство разработки программного обеспечения: книга / Р. Мартин. - СПб. : Питер, 2019. - 352 с. - ISBN 978-5-496-03181-7.

2. Орлов С. А., Технологии разработки программного обеспечения: учебник / С. А. Орлов. - СПб. : Питер, 2024. - ISBN 5-94723-820-9.

3. Ратушняк Г.Я., Технологии разработки и проектирования информационных систем. Часть 1 [Электронный ресурс] : учебное пособие / Г.Я. Ратушняк, А.Л. Золкин. - М. : Русайнс, 2022. - 201 с. - ISBN 978-5-4365-9893-2. - Режим доступа: <https://book.ru/book/945635>

4. Ратушняк Г.Я., Технологии разработки и проектирования информационных систем. Часть 2 [Электронный ресурс] : учебное пособие / Г.Я. Ратушняк, А.Л. Золкин. - М. : Русайнс, 2022. - 349 с. - ISBN 978-5-466-01439-6. - Режим доступа: <https://book.ru/book/946232>

Дополнительная литература:

1. Зубкова Т. М., Технология разработки программного обеспечения: учебное пособие / Т. М. Зубкова. - М. : Юрайт, 2023.

2. Письменский Г.И., Основы проектирования электронной информационно-образовательной среды [Электронный ресурс] : монография / Г.И. Письменский, В.В. Киселев, Л.В. Неровный, С.В. Сафонова. - М. : Русайнс, 2022. - 102 с. - ISBN 978-5-466-02119-6. - Режим доступа: <https://book.ru/book/947240>

3. Федоричев Л. А., Букунова О. В., Технология разработки программного обеспечения: учебное пособие / Л. А. Федоричев, О. В. Букунова. - СПб. : Лань, 2023.

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: «НТЦ 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 Pepot, 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/>

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 видов учебной деятельности, предусмотренной учебным планом, в том числе помещения для самостоятельной работы, с указанием перечня основного оборудования, учебно- наглядных пособий и используемого программного	Адрес (местоположение) помещений для проведения всех видов учебной деятельности, предусмотренной учебным планом (в случае реализации образовательной программы в сетевой форме дополнительно указывается
--	---

обеспечения	наименование организации, с которой заключен договор)
<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 обучающийся, с перечнем основного оборудования:</p> <ul style="list-style-type: none"> - Столы для обучающихся; - Стулья для обучающихся; - Стол педагогического работника; - Стол педагогического работника; - Компьютеры с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду лицензиата; - Маркерная или меловая доска; - Проектор. 	170001, Россия, город Тверь, улица Спартака, дом 26а
<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 обучающийся, с перечнем основного оборудования:</p> <ul style="list-style-type: none"> - Столы для обучающихся; - Стулья для обучающихся; - Стол педагогического работника; - Стол педагогического работника; - Компьютеры с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду лицензиата; - Маркерная или меловая доска; - Проектор. 	170001, Россия, город Тверь, улица Спартака, дом 26а
<p>Помещение для самостоятельной работы обучающихся с перечнем основного оборудования:</p> <ul style="list-style-type: none"> - Столы для обучающихся; - Стулья для обучающихся; - Стол педагогического работника; - Стол педагогического работника; - Компьютеры с возможностью подключения к сети «Интернет» и обеспечением доступа в 	170001, Россия, город Тверь, улица Спартака, дом 26а

<p>электронную информационно-образовательную среду лицензиата;</p> <ul style="list-style-type: none"> - 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 Spartaka, 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.05 «Технология разработки программного обеспечения»**

**Направление подготовки: 09.03.02 Информационные системы и технологии
Направленность (профиль) программы бакалавриата
«Информационные технологии»**

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-1, УК-2, ПК-1, ПК-2, ПК-3, ПК-4.

Код компетенции	Наименование компетенции	Индекс и наименование индикатора содержания компетенции	Дескрипторы – основные признаки освоения (показатели достижения результата)
УК-1	Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.1 Способен осуществлять поиск и критический анализ информации	<p>Знать:</p> <ul style="list-style-type: none"> - Основные источники информации для разработки программного обеспечения (далее ПО). - Методы критического анализа информации. <p>Уметь:</p> <ul style="list-style-type: none"> - Находить актуальную информацию в специализированных источниках. - Анализировать и оценивать достоверность и релевантность информации. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками эффективного поиска информации. - Навыками критического мышления при анализе информации.
		УК-1.2 Способен применять системный подход для решения поставленных задач	<p>Знать:</p> <ul style="list-style-type: none"> - Принципы системного подхода в разработке ПО. - Основные этапы системного анализа и проектирования. <p>Уметь:</p> <ul style="list-style-type: none"> - Применять системный подход для анализа задач. - Разрабатывать системные решения для поставленных задач. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками системного анализа проблем. - Навыками интеграции различных элементов системы в единое целое.
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	<p>Знать:</p> <ul style="list-style-type: none"> - Методы декомпозиции целей на задачи. - Принципы приоритизации задач. <p>Уметь:</p> <ul style="list-style-type: none"> - Определять и формулировать задачи на основе целей проекта. - Устанавливать приоритеты для выполнения задач. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками декомпозиции целей на задачи. - Навыками управления задачами и ресурсами.

		УК-2.2 Способен выбирать оптимальные способы решения задач, исходя из правовых норм, ресурсов и ограничений	<p>Знать:</p> <ul style="list-style-type: none"> - Основные правовые нормы в области информационных технологий (далее ИТ). - Принципы оценки и управления ресурсами. <p>Уметь:</p> <ul style="list-style-type: none"> - Оценивать доступные ресурсы и ограничения. - Выбирать наилучшие способы решения задач с учетом правовых норм. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками оценки ресурсов и ограничений. - Навыками принятия оптимальных решений в рамках правовых норм.
ПК-1	Разработка и отладка программного кода	ПК-1.1 Способен разрабатывать программный код	<p>Знать:</p> <ul style="list-style-type: none"> - Основные языки программирования для разработки ПО. - Принципы написания качественного кода. <p>Уметь:</p> <ul style="list-style-type: none"> - Писать программный код на одном из изучаемых языков. - Разрабатывать алгоритмы и структуры данных для решения задач. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками написания и документирования кода. - Навыками использования инструментов разработки.
		ПК-1.2 Способен отлаживать программный код	<p>Знать:</p> <ul style="list-style-type: none"> - Основные методы и инструменты отладки кода. - Принципы устранения ошибок и багов в коде. <p>Уметь:</p> <ul style="list-style-type: none"> - Использовать отладчики и логи для выявления ошибок. - Исправлять выявленные ошибки и оптимизировать код. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками отладки и тестирования кода. - Навыками применения инструментов для отладки.
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 кода. - Основные виды тестов (модульные, интеграционные, системные). <p>Уметь:</p> <ul style="list-style-type: none"> - Писать и выполнять тесты для проверки кода. - Использовать тестовые фреймворки для автоматизации тестирования. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками написания и выполнения тестов. - Навыками применения инструментов для тестирования.

		<p>ПК-2.2 Способен рефакторить программный код</p>	<p>Знать:</p> <ul style="list-style-type: none"> - Принципы и методы рефакторинга. - Основные техники улучшения структуры кода. <p>Уметь:</p> <ul style="list-style-type: none"> - Анализировать код на предмет улучшений. - Применять техники рефакторинга для повышения качества кода. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками рефакторинга кода. - Навыками оценки и улучшения структуры кода.
ПК-3	<p>Интеграция программных модулей и компонентов и проверка работоспособности выпусков программного продукта</p>	<p>ПК-3.1 Способен интегрировать программные модули и компоненты</p>	<p>Знать:</p> <ul style="list-style-type: none"> - Принципы интеграции модулей и компонентов. - Методы и инструменты для интеграции. <p>Уметь:</p> <ul style="list-style-type: none"> - Интегрировать модули и компоненты в единое приложение. - Проверять корректность интеграции и совместимость компонентов. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками интеграции программных модулей. - Навыками использования инструментов для интеграции.
		<p>ПК-3.2 Способен проверять работоспособность выпусков программного продукта</p>	<p>Знать:</p> <ul style="list-style-type: none"> - Методы и инструменты для проверки выпусков ПО. - Принципы контроля качества выпусков. <p>Уметь:</p> <ul style="list-style-type: none"> - Проводить тестирование и верификацию выпусков ПО. - Оценивать стабильность и производительность программного продукта. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками тестирования и верификации выпусков ПО. - Навыками применения инструментов контроля качества.
ПК-4	Разработка требований и проектирование программного обеспечения	<p>ПК-4.1 Способен разрабатывать требования к программному обеспечению</p>	<p>Знать:</p> <ul style="list-style-type: none"> - Принципы и методы анализа требований. - Техники документирования требований. <p>Уметь:</p> <ul style="list-style-type: none"> - Собирать и анализировать требования к ПО. - Документировать требования в виде спецификаций. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками анализа и сбора требований. - Навыками создания и поддержки документации по требованиям.

		<p>ПК-4.2</p> <p>Способен проектировать программное обеспечение</p>	<p>Знать:</p> <ul style="list-style-type: none"> - Принципы и методы проектирования ПО. - Основные подходы к архитектурному дизайну. <p>Уметь:</p> <ul style="list-style-type: none"> - Разрабатывать архитектуру и дизайн ПО. - Использовать UML и другие инструменты для проектирования. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками проектирования программного обеспечения. - Навыками применения инструментов для проектирования.
--	--	---	---

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 (сituационные 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); недостаточное 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 (модулю)	ШКАЛА ОЦЕНИВАНИЯ				Процедуры оценивания
	«отлично»	«хорошо»	«удовлетворительно»	«неудовлетворительно»	
УК-1, УК-2, ПК-1, ПК-2,	Обучаемый продемонстрировал: глубокие искрывающие	Обучаемый продемонстрировал: твердые и достаточно	Обучаемый продемонстрировал: твердые знания и понимание	Обучаемый продемонстрировал неправильные ответы на основные вопросы;	Тестовые задания

<u>ПК-3</u> <u>ПК-4</u> Знать:	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.	полные знания учебного материала; правильное понимание сущности и взаимосвязи рассматриваемых процессов и явлений; последовательные, правильные, конкретные ответы на поставленные вопросы при свободном устраниении замечаний по отдельным вопросам; достаточное владение литературой.	основного учебного материала; правильные, без грубых ошибок, ответы на поставленные вопросы при устраниении неточностей и несущественных ошибок в освещении отдельных положений при наводящих вопросах преподавателя; недостаточно полное владение литературой, рекомендованной учебной программой дисциплины.	грубые ошибки в ответах; непонимание сущности излагаемых вопросов; неуверенные и неточные ответы на дополнительные вопросы; не владеет основной литературой, рекомендованной учебной программой дисциплины.	
<u>УК-1</u> <u>УК-2</u> <u>ПК-1</u> <u>ПК-2</u> <u>ПК-3</u> <u>ПК-4</u> Уметь:	Обучаемый продемонстрировал: 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: понимание учебного материала; логически последовательные, правильные и конкретные ответы на основные вопросы, включая задания/вопросы, включая дополнительные; устраниние замечаний по отдельным элементам задания; владение основной и дополнительной литературой, рекомендованной учебной программой дисциплины	Обучаемый продемонстрировал: понимание основного учебного материала; правильные, без грубых ошибок, ответы на основные вопросы, включая дополнительные, устранил, при устраниении, принял наводящих вопросах преподавателя, замечаний и грубых ошибок по вопросу; отдельным элементам задания; недостаточное полное владение литературой, рекомендованной учебной программой дисциплины	Обучаемый продемонстрировал: понимание основного учебного материала; не дал правильные ответы на основные вопросы, включая дополнительные; не устранил, при устраниении, принял наводящих вопросах преподавателя, замечаний и грубых ошибок по вопросу; отдельным элементам задания; недостаточное полное владение литературой, рекомендованной учебной программой дисциплины	Вопросы Практические задания
<u>УК-1</u> <u>УК-2</u> <u>ПК-1</u> <u>ПК-2</u> <u>ПК-3</u> <u>ПК-4</u> Владеть:	Обучаемый продемонстрировал: понимание учебного материала; правильные и конкретные ответы, включая дополнительные уточняющие вопросы;	Обучаемый продемонстрировал: понимание учебного материала; продемонстрировал логически последовательные, достаточны полные, верные ответы;	Обучаемый продемонстрировал: понимание основного учебного материала; без грубых ошибок дал ответы на поставленные вопросы, в том числе при наводящих вопросах	Обучаемый продемонстрировал: понимание основного учебного материала; дал неправильные ответы на поставленные вопросы; не владеет основной учебной литературой, рекомендованной	Вопросы Практические задания

	свободное владение основной и дополнительной литературой, рекомендованной учебной программой дисциплины	самостоятельно устранил замечания по отдельным элементам; владение основной и дополнительной литературой, рекомендованной учебной программой дисциплины	преподавателя; недостаточно полное владение литературой, рекомендованной учебной программой дисциплины	учебной программой дисциплины	
--	---	---	--	-------------------------------	--

1. Оценочные материалы для самостоятельной работы обучающихся (студентов)

1.1 Доклад 5 семестр

Доклад позволит студентам углубиться в технологии разработки программного обеспечения, а также понять их важность и применение в современных информационных системах. Доклад покрывает компетенции УК-1.1, УК-1.2, УК-2.1, УК-2.2, ПК-1.1, ПК-1.2, ПК-2.1, ПК-2.2, ПК-3.1, ПК-3.2, ПК-4.1, ПК-4.2.

Примерная тематика докладов:

- Введение в технологии разработки ПО:** Основные подходы и методологии разработки ПО, понятие и цели разработки программного обеспечения.
- Жизненный цикл разработки ПО:** Изучение моделей жизненного цикла ПО: водопадная, спиральная, Agile, Scrum. Основные этапы жизненного цикла.
- Управление версиями и семантическое версионирование:** Основы управления версиями, семантическое версионирование, использование тегов и релизов.
- Git и методы работы с ним:** Основы работы с Git, различные стратегии ветвления и слияния, такие как Git-flow и GitHub Flow, использование Git для управления проектами.
- Управление задачами и релизами:** Инструменты и методы управления задачами и релизами, использование систем трекинга задач (JIRA, Trello).
- Основы CI/CD:** Понятие непрерывной интеграции и непрерывного развертывания, основные инструменты и методы CI/CD.

1.2 Доклад 6 семестр

Доклад позволит студентам углубиться в технологии разработки программного обеспечения, а также понять их важность и применение в современных информационных системах. Доклад покрывает компетенции УК-1.1, УК-1.2, УК-2.1, УК-2.2, ПК-1.1, ПК-1.2, ПК-2.1, ПК-2.2, ПК-3.1, ПК-3.2, ПК-4.1, ПК-4.2.

Примерная тематика докладов:

- Командная работа и распределение ролей:** Организация работы в команде, роли и ответственности в команде разработчиков.
- Методологии разработки ПО:** Детальное изучение методологий Agile и Scrum, практики и артефакты.
- Планирование и контроль проектов:** Методы планирования и контроля проектов, использование диаграммы Ганта и других инструментов.
- Управление конфигурацией и релизами:** Принципы управления конфигурацией, подготовка и выпуск релизов.
- Современные инструменты разработки ПО:** Обзор современных инструментов и технологий, применяемых в разработке ПО.
- Будущее разработки ПО:** Тенденции и перспективы в области разработки ПО, влияние искусственного интеллекта и машинного обучения на процессы разработки.

Цель написания докладов: Углубить понимание и критическое осмысление роли технологий разработки программного обеспечения в современных информационных системах, развивая аналитические и научные навыки студентов.

Структура доклада:

- 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 технологий разработки 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 разработки ПО.
 - Анализ современных тенденций.
 - Примеры применения технологий разработки ПО в реальных проектах.
- 3. Аналитическая часть (20%)**
 - Глубина анализа роли технологий разработки 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 +.

5 семестр

1. Какие основные подходы к разработке ПО существуют? (УК-1.1)
 - Водопадная, спиральная, Agile, Scrum +
 - Водопадная, спиральная, V-модель, RAD
 - Agile, Scrum, Lean, Kanban
 - Водопадная, Agile, DevOps, Lean
2. Какой методологии соответствует итеративный процесс разработки? (ПК-1.1)
 - Водопадная модель

- V-модель
 - Спиральная модель
 - Agile +
- 3. Что означает семантическое версионирование? (УК-1.2)
 - Присвоение версий на основе времени разработки
 - Случайное назначение версий
 - Использование версии 1.0.0 для всех релизов
 - Принципы изменения версий в зависимости от изменений в коде +
- 4. Какие основные команды используются в Git? (ПК-1.1)
 - init, clone, add, commit, push, pull +
 - open, save, close, sync
 - start, create, save, send, receive
 - begin, copy, update, upload, download
- 5. Какую стратегию работы с ветками использует Git-flow? (УК-2.2)
 - Главная ветка, ветка разработки, функциональные ветки, релизные ветки, исправления +
 - Главная ветка, ветка разработки, экспериментальные ветки
 - Ветка мастера, ветка рабочая
 - Единая ветка для всех изменений
- 6. Что такое JIRA и для чего она используется? (ПК-4.2)
 - Инструмент для создания диаграмм
 - Система для автоматического тестирования
 - Система трекинга задач и управления проектами +
 - Система управления базами данных
- 7. Какие ключевые этапы включает процесс CI/CD? (ПК-2.1)
 - Разработка, тестирование, деплоймент, мониторинг +
 - Анализ, дизайн, реализация, поддержка
 - Планирование, разработка, развертывание, обслуживание
 - Инициация, планирование, выполнение, завершение
- 8. Какие роли могут быть в команде разработки ПО? (УК-3.1)
 - Разработчик, тестировщик, менеджер проекта, аналитик +
 - Менеджер по продажам, бухгалтер, HR, маркетолог
 - Оператор, дежурный, секретарь, водитель
 - Главный инженер, ассистент, секретарь, курьер

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 Тестовые задания для текущего контроля успеваемости в виде ЗНАНИЙ

В тестовом задании вопросы, которые имеют закрытый характер.

Правильные ответы выделены знаком +.

6 семестр

1. Какова основная цель методологии Agile? (УК-1.1)
 - Быстрая разработка без тестирования
 - Гибкость и адаптивность к изменениям +
 - Использование водопадной модели
 - Строгое следование плану
2. Что такое Scrum и какие артефакты он включает? (ПК-4.1)
 - Методология с четко фиксированными этапами, включает диаграмму Ганта
 - Фреймворк Agile, включает спринты, бэклог, доску задач, спринт-ревью +
 - Модель разработки с итеративным подходом, включает тестовые случаи
 - Инструмент для управления проектами, включает план, график, отчет
3. Какие инструменты используются для создания диаграммы Ганта? (ПК-4.2)
 - Excel, Word, PowerPoint
 - Photoshop, Illustrator, CorelDRAW
 - Microsoft Project, GanttProject, Asana +
 - JIRA, Trello, Slack
4. Что включает процесс управления конфигурацией ПО? (УК-2.1)
 - Определение, контроль и отчетность по изменениям в ПО +
 - Маркетинговый анализ и продажи
 - Подбор и обучение персонала
 - Разработка и тестирование ПО
5. Какие инструменты используются для CI/CD? (ПК-2.1)
 - Photoshop, Illustrator, CorelDRAW
 - Trello, Asana, Slack
 - Jenkins, GitHub Actions, GitLab CI +
 - Microsoft Office, LibreOffice
6. Какую функцию выполняет WorkManager в Android? (ПК-3.1)
 - Управление базами данных
 - Обработка мультимедийных данных
 - Управление задачами в фоновом режиме +
 - Создание пользовательского интерфейса
7. Какие этапы включает процесс тестирования ПО? (ПК-2.2)
 - Дизайн, кодирование, развертывание, поддержка
 - Анализ требований, проектирование, реализация, поддержка
 - Планирование тестирования, разработка тест-кейсов, выполнение тестов, отчет о результатах +
 - Планирование, разработка, развертывание, обслуживание

8. Что означает понятие "релиз" в разработке ПО? (ПК-3.2)
 - Начало разработки нового проекта
 - Окончательный выпуск версии ПО для использования +
 - Установка ПО на сервер
 - Тестирование кода

Критерии оценки результатов теста

5. **"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.
6. **"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.
7. **"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.
8. **"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.

2.3 Вопросы для текущего контроля успеваемости в виде УМЕНИЙ **5 семестр**

1. Что такое жизненный цикл разработки ПО и какие основные модели существуют? (УК-1.1)
 - Правильный ответ: Жизненный цикл разработки ПО — это процесс, охватывающий все стадии создания программного обеспечения от концепции до выхода из эксплуатации. Основные модели включают водопадную модель, спиральную модель, V-модель и Agile.
2. В чем отличие водопадной модели от Agile? (ПК-1.1)
 - Правильный ответ: Водопадная модель представляет собой последовательный процесс разработки, где каждая стадия должна быть завершена до начала следующей. Agile — это итеративный подход, где проект делится на небольшие циклы (итерации), позволяя гибко реагировать на изменения требований.
3. Что такое семантическое версионирование и как оно применяется? (УК-1.2)
 - Правильный ответ: Семантическое версионирование — это система назначения версий ПО, где версия обозначается тремя числами (например, 1.0.0), обозначающими мажорные, минорные изменения и патчи. Оно помогает пользователям понять, какие изменения были внесены в ПО и как это влияет на совместимость.

4. Какие основные команды используются в Git для управления версиями? (ПК-1.1)
 - Правильный ответ: Основные команды Git включают init (инициализация репозитория), clone (клонирование репозитория), add (добавление изменений), commit (фиксация изменений), push (отправка изменений на сервер), pull (получение изменений с сервера).
5. Что такое Git-flow и как он помогает в управлении версиями? (УК-2.2)
 - Правильный ответ: Git-flow — это модель ветвления, которая помогает управлять процессом разработки ПО. Она включает основные ветки (master, develop) и вспомогательные ветки (feature, release, hotfix), что позволяет структурировать работу над проектом и управлять релизами.
6. Какие функции выполняет JIRA в управлении проектами? (ПК-4.2)
 - Правильный ответ: JIRA — это система трекинга задач и управления проектами, которая помогает командам планировать, отслеживать и управлять проектами, назначать задачи, устанавливать приоритеты, отслеживать прогресс и создавать отчеты.
7. Что включает процесс CI/CD и какие инструменты используются для его реализации? (ПК-2.1)
 - Правильный ответ: CI/CD (Continuous Integration/Continuous Deployment) включает непрерывную интеграцию кода и непрерывное развертывание приложений. Инструменты для CI/CD включают Jenkins, GitHub Actions, GitLab CI и другие.
8. Какие роли могут быть в команде разработки ПО и какие функции они выполняют? (УК-3.1)
 - Правильный ответ: Роли в команде разработки ПО включают разработчика (создание кода), тестировщика (проверка качества кода), менеджера проекта (управление проектом), аналитика (сбор и анализ требований).

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.

2.4 Вопросы для текущего контроля успеваемости в виде УМЕНИЙ 6 семестр

1. В чем заключается основная идея Agile и какие преимущества она предоставляет? (УК-1.1)
 - Правильный ответ: Основная идея Agile заключается в гибкости и адаптивности к изменениям, итеративном подходе к разработке и тесном взаимодействии с клиентом. Преимущества включают быстрое реагирование на изменения, улучшенное взаимодействие в команде и повышение качества продукта.
2. Что такое Scrum и какие артефакты он включает? (ПК-4.1)
 - Правильный ответ: Scrum — это фреймворк Agile, который включает спринты (короткие итерации), бэклог продукта (список задач), спринт-бэклог (задачи для текущего спринта), доску задач (визуализация прогресса), спринт-ревью (оценка результатов спринта).
3. Какие инструменты используются для создания и управления диаграммами Ганта? (ПК-4.2)
 - Правильный ответ: Для создания и управления диаграммами Ганта используются инструменты Microsoft Project, GanttProject, Asana.
4. Что такое управление конфигурацией ПО и какие задачи оно решает? (УК-2.1)
 - Правильный ответ: Управление конфигурацией ПО включает определение, контроль и отчетность по изменениям в программных продуктах. Задачи включают управление версиями, отслеживание изменений и обеспечение целостности конфигурации.
5. Какие этапы включает процесс управления конфигурацией и релизами? (ПК-2.1)
 - Правильный ответ: Процесс управления конфигурацией и релизами включает идентификацию конфигурационных единиц, контроль изменений, отчетность по конфигурациям, планирование и выполнение релизов.
6. Какие преимущества предоставляет использование CI/CD в разработке ПО? (ПК-2.2)
 - Правильный ответ: CI/CD автоматизирует процесс интеграции и развертывания, уменьшает риск ошибок, ускоряет выпуск новых версий, повышает качество и стабильность ПО.
7. Какую роль выполняет WorkManager в Android и как он используется? (ПК-3.1)
 - Правильный ответ: WorkManager в Android управляет задачами в фоновом режиме, обеспечивая выполнение задач даже при перезагрузке устройства. Он используется для задач, требующих гарантированного выполнения.
8. Какие этапы включает процесс тестирования ПО и какие инструменты для этого используются? (ПК-2.2)
 - Правильный ответ: Процесс тестирования ПО включает планирование тестирования, разработку тест-кейсов, выполнение тестов, отчет о результатах. Инструменты включают JUnit, pytest, Selenium.

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.

2.5 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 Agile с их определениями (УК-1.1, УК-1.2)

Понятие	Определение
A - Scrum	1 - Фреймворк Agile, использующий спринты и артефакты
B - Sprint	2 - Короткий итерационный цикл разработки в Scrum
C - Product Backlog	3 - Список всех задач и требований к продукту
D - Retrospective	4 - Встреча команды для обсуждения прошедшего спринта

Правильный ответ: A-1, B-2, C-3, D-4

Задача 2. Соотнесите этапы жизненного цикла разработки ПО с их описаниями (ПК-1.1, ПК-1.2)

Этап	Описание
A - Анализ требований	1 - Сбор и документирование требований к системе
B - Дизайн	2 - Разработка архитектуры и дизайна системы
C - Реализация	3 - Написание исходного кода программы

D - Тестирование	4 - Проверка правильности и качества кода
------------------	---

Правильный ответ: A-1, B-2, C-3, D-4

Задача 3. Соотнесите команды Git с их функциями (ПК-1.1, ПК-2.2)

Команда	Функция
A - clone	1 - Клонирование удаленного репозитория
B - commit	2 - Фиксация изменений в локальном репозитории
C - push	3 - Отправка изменений в удаленный репозиторий
D - pull	4 - Получение изменений из удаленного репозитория

Правильный ответ: A-1, B-2, C-3, D-4

Задача 4. Соотнесите ключевые этапы CI/CD с их описаниями (ПК-2.1, ПК-3.1)

Этап	Описание
A - Continuous Integration	1 - Непрерывное интегрирование кода в основной репозиторий
B - Continuous Deployment	2 - Автоматизированное развертывание приложений
C - Testing	3 - Автоматическое тестирование кода
D - Monitoring	4 - Непрерывный мониторинг и сбор метрик

Правильный ответ: A-1, B-2, C-3, D-4

Задача 5. Соотнесите роли в команде разработки ПО с их функциями (УК-3.1, ПК-4.1)

Роль	Функция
A - Разработчик	1 - Написание и отладка кода
B - Тестировщик	2 - Проверка качества кода
C - Менеджер проекта	3 - Управление проектом и координация команды
D - Аналитик	4 - Сбор и анализ требований

Правильный ответ: A-1, B-2, C-3, D-4

Задача 6. Соотнесите основные понятия Scrum с их определениями (ПК-4.2, УК-1.1)

Понятие	Определение
A - Product Owner	1 - Ответственен за максимизацию ценности продукта
B - Scrum Master	2 - Обеспечивает соблюдение Scrum и помогает команде
C - Development Team	3 - Команда, работающая над созданием продукта

Правильный ответ: A-1, B-2, C-3, D-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 ЗАЧЕТА 5 семестр

1. Что такое жизненный цикл разработки программного обеспечения? (УК-1.1)
 - Правильный ответ: Жизненный цикл разработки ПО — это процесс создания и развития программного продукта от начальной идеи до его выхода из эксплуатации. Включает этапы: анализ требований, проектирование, реализация, тестирование, развертывание и сопровождение.
2. Какие модели жизненного цикла разработки ПО вы знаете и в чем их особенности? (УК-1.2)
 - Правильный ответ: Основные модели включают водопадную модель, спиральную модель, V-модель и Agile. Водопадная модель линейна и последовательна, спиральная модель итеративна и включает оценку рисков, V-модель сочетает разработку и тестирование, Agile фокусируется на итеративной разработке и гибкости.
3. Опишите основные принципы Agile и его преимущества. (ПК-1.1)
 - Правильный ответ: Agile — это итеративный и инкрементальный подход к разработке ПО, фокусирующийся на гибкости и быстром отклике на изменения. Преимущества включают быструю адаптацию к изменениям, улучшенное взаимодействие в команде, повышение качества продукта и удовлетворенность клиентов.
4. Что такое Scrum и какие основные роли в нем существуют? (ПК-4.1)
 - Правильный ответ: Scrum — это фреймворк Agile, использующий спринты для итеративной разработки. Основные роли включают Product Owner (отвечает за максимизацию ценности продукта), Scrum Master (обеспечивает соблюдение Scrum и помогает команде) и Development Team (разработчики, работающие над созданием продукта).
5. Какие артефакты существуют в Scrum и для чего они предназначены? (УК-1.1)
 - Правильный ответ: Основные артефакты Scrum включают Product Backlog (список всех задач и требований к продукту), Sprint Backlog (список задач для текущего спринта) и Increment (результат работы команды за спринт, готовый к релизу).
6. Что такое семантическое версионирование и как оно используется? (ПК-1.1)
 - Правильный ответ: Семантическое версионирование — это система назначения версий ПО, где версия обозначается тремя числами (например, 1.0.0), обозначающими мажорные, минорные изменения и патчи. Она помогает

пользователям понять, какие изменения были внесены в ПО и как это влияет на совместимость.

7. Какие основные команды используются в Git для управления версиями? (ПК-1.1)
 - Правильный ответ: Основные команды Git включают init (инициализация репозитория), clone (клонирование репозитория), add (добавление изменений), commit (фиксация изменений), push (отправка изменений на сервер), pull (получение изменений с сервера).
8. Что такое Git-flow и как он помогает в управлении версиями? (ПК-2.1)
 - Правильный ответ: Git-flow — это модель ветвления, которая помогает управлять процессом разработки ПО. Она включает основные ветки (master, develop) и вспомогательные ветки (feature, release, hotfix), что позволяет структурировать работу над проектом и управлять релизами.
9. Какова роль менеджера проекта в команде разработки ПО? (ПК-4.2)
 - Правильный ответ: Менеджер проекта отвечает за планирование, исполнение и завершение проекта. Он координирует работу команды, управляет ресурсами, следит за соблюдением сроков и бюджета, а также за достижением целей проекта.
10. Какие этапы включает процесс CI/CD и какие инструменты для этого используются? (ПК-3.1)
 - Правильный ответ: Процесс CI/CD включает непрерывную интеграцию кода (Continuous Integration) и непрерывное развертывание приложений (Continuous Deployment). Инструменты для CI/CD включают Jenkins, GitHub Actions, GitLab CI и другие.
11. Какие преимущества предоставляет использование CI/CD в разработке ПО? (ПК-3.1)
 - Правильный ответ: CI/CD автоматизирует процесс интеграции и развертывания, уменьшает риск ошибок, ускоряет выпуск новых версий, повышает качество и стабильность ПО.
12. Какие задачи выполняет JIRA в управлении проектами? (ПК-4.2)
 - Правильный ответ: JIRA — это система трекинга задач и управления проектами, которая помогает командам планировать, отслеживать и управлять проектами, назначать задачи, устанавливать приоритеты, отслеживать прогресс и создавать отчеты.
13. Какие роли могут быть в команде разработки ПО и какие функции они выполняют? (УК-2.2)
 - Правильный ответ: Роли в команде разработки ПО включают разработчика (создание кода), тестировщика (проверка качества кода), менеджера проекта (управление проектом), аналитика (сбор и анализ требований).
14. Какую стратегию работы с ветками использует Git-flow? (УК-2.2)
 - Правильный ответ: Git-flow использует главную ветку, ветку разработки, функциональные ветки, релизные ветки и ветки для исправления ошибок. Это помогает структурировать работу над проектом и управлять релизами.

Критерии оценивания для зачета (зачет/незачет)

1. **Понимание основных понятий и терминов**
 - **Зачет:** Студент правильно использует основные понятия и термины, объясняет их в контексте.
 - **Незачет:** Студент не знает или неправильно использует основные понятия и термины, не может объяснить их.
2. **Точность и полнота ответов на теоретические вопросы**
 - **Зачет:** Ответы на большинство вопросов полные и точные, основные аспекты тем раскрыты.

- **Незачет:** Ответы неполные, содержат значительные ошибки или отсутствуют важные аспекты тем.

3. Способность к применению знаний на практике

- **Зачет:** Студент правильно решает практические задачи, выполняет SQL-запросы без ошибок.
- **Незачет:** Студент допускает ошибки при решении практических задач, не может правильно выполнить SQL-запросы.

4. Качество выполнения практических заданий

- **Зачет:** Практические задания выполнены корректно, все необходимые элементы присутствуют.
- **Незачет:** Практические задания выполнены некорректно, содержат ошибки или неполные.

Окончательное решение по зачету:

- **Зачет:** Студент демонстрирует понимание основных понятий, корректно отвечает на теоретические вопросы и выполняет практические задания.
- **Незачет:** Студент не демонстрирует понимания основных понятий, отвечает с ошибками на теоретические вопросы и/или не справляется с практическими заданиями.

3.2 Вопросы для проведения промежуточной аттестации в форме ЭКЗАМЕНА 6 семестр

1. Что такое жизненный цикл разработки программного обеспечения и какие основные этапы он включает? (УК-1.1)
 - Правильный ответ: Жизненный цикл разработки ПО — это процесс создания и развития программного продукта от начальной идеи до его выхода из эксплуатации. Основные этапы включают: анализ требований, проектирование, реализация, тестирование, развертывание и сопровождение.
2. В чем различие между водопадной моделью и Agile? (УК-1.2)
 - Правильный ответ: Водопадная модель представляет собой линейный последовательный процесс, где каждая стадия должна быть завершена перед началом следующей. Agile — итеративный подход, где проект делится на небольшие циклы (итерации), что позволяет гибко реагировать на изменения требований и быстро выпускать обновления.
3. Какие ключевые принципы Agile вы знаете? (ПК-1.1)
 - Правильный ответ: Ключевые принципы Agile включают: гибкость, итеративность, взаимодействие с клиентом, приоритет рабочих продуктов, кросс-функциональные команды, адаптация к изменениям, и постоянное улучшение.
4. Опишите роли и артефакты в Scrum. (ПК-4.1)
 - Правильный ответ: Роли в Scrum включают Product Owner (ответственный за максимизацию ценности продукта), Scrum Master (обеспечивает соблюдение Scrum и помогает команде), и Development Team (работает над созданием продукта). Основные артефакты включают Product Backlog (список всех задач и требований), Sprint Backlog (задачи для текущего спринта), и Increment (результат работы команды за спринт).
5. Что такое семантическое версионирование и как оно используется? (ПК-1.2)
 - Правильный ответ: Семантическое версионирование — это система назначения версий ПО, где версия обозначается тремя числами (например, 1.0.0). Первое число указывает на мажорные изменения, второе на минорные, третье на патчи. Это помогает пользователям понять, какие изменения были внесены и как это влияет на совместимость.
6. Какие основные команды используются в Git и какие функции они выполняют? (ПК-1.1)

- Правильный ответ: Основные команды Git включают init (инициализация репозитория), clone (клонирование репозитория), add (добавление изменений), commit (фиксация изменений), push (отправка изменений на сервер), pull (получение изменений с сервера). Эти команды позволяют управлять изменениями кода и синхронизировать работу в команде.
- 7. Что такое Git-flow и как он структурирует работу над проектом? (ПК-2.1)
 - Правильный ответ: Git-flow — это модель ветвления, которая помогает управлять процессом разработки ПО. Она включает основные ветки (master, develop) и вспомогательные ветки (feature, release, hotfix), что позволяет структурировать работу над проектом и управлять релизами.
- 8. Какие этапы включает процесс CI/CD и какие инструменты для этого используются? (ПК-3.1)
 - Правильный ответ: Процесс CI/CD включает непрерывную интеграцию кода (Continuous Integration) и непрерывное развертывание приложений (Continuous Deployment). Инструменты для CI/CD включают Jenkins, GitHub Actions, GitLab CI и другие.
- 9. Какие преимущества предоставляет использование CI/CD в разработке ПО? (ПК-2.2)
 - Правильный ответ: CI/CD автоматизирует процесс интеграции и развертывания, уменьшает риск ошибок, ускоряет выпуск новых версий, повышает качество и стабильность ПО.
- 10. Каковы основные задачи управления конфигурацией ПО? (УК-2.1)
 - Правильный ответ: Управление конфигурацией ПО включает определение, контроль и отчетность по изменениям в программных продуктах. Задачи включают управление версиями, отслеживание изменений и обеспечение целостности конфигурации.
- 11. Какие задачи выполняет менеджер проекта в команде разработки ПО? (ПК-4.2)
 - Правильный ответ: Менеджер проекта отвечает за планирование, исполнение и завершение проекта. Он координирует работу команды, управляет ресурсами, следит за соблюдением сроков и бюджета, а также за достижением целей проекта.
- 12. Что такое управление рисками в проекте разработки ПО? (УК-2.2)
 - Правильный ответ: Управление рисками включает идентификацию, анализ и планирование мер по уменьшению влияния рисков на проект. Это помогает предвидеть возможные проблемы и своевременно принимать меры для их предотвращения или минимизации.
- 13. Какую роль выполняет JIRA в управлении проектами? (ПК-4.2)
 - Правильный ответ: JIRA — это система трекинга задач и управления проектами, которая помогает командам планировать, отслеживать и управлять проектами, назначать задачи, устанавливать приоритеты, отслеживать прогресс и создавать отчеты.
- 14. Какие методы тестирования ПО вы знаете и как они применяются? (ПК-2.1)
 - Правильный ответ: Основные методы тестирования включают юнит-тестирование (проверка отдельных компонентов), интеграционное тестирование (проверка взаимодействия компонентов), системное тестирование (проверка всей системы) и приемочное тестирование (проверка соответствия требованиям). Эти методы помогают обеспечить качество и надежность ПО.
- 15. Что такое Scrum и какие основные роли в нем существуют? (ПК-4.1)
 - Правильный ответ: Scrum — это фреймворк Agile, использующий спринты для итеративной разработки. Основные роли включают Product Owner (отвечает за максимизацию ценности продукта), Scrum Master (обеспечивает соблюдение Scrum и помогает команде) и Development Team (разработчики, работающие над созданием продукта).

16. Какие ключевые артефакты существуют в Scrum и для чего они предназначены? (ПК-4.2)
 - Правильный ответ: Основные артефакты Scrum включают Product Backlog (список всех задач и требований к продукту), Sprint Backlog (список задач для текущего спринта) и Increment (результат работы команды за спринт, готовый к релизу).
17. В чем заключается основная идея Agile и какие преимущества она предоставляет? (УК-1.1)
 - Правильный ответ: Основная идея Agile заключается в гибкости и адаптивности к изменениям, итеративном подходе к разработке и тесном взаимодействии с клиентом. Преимущества включают быстрое реагирование на изменения, улучшенное взаимодействие в команде и повышение качества продукта.
18. Какие функции выполняет Product Owner в Scrum? (ПК-4.1)
 - Правильный ответ: Product Owner отвечает за максимизацию ценности продукта, управление Product Backlog, определение приоритетов задач, взаимодействие с клиентами и заинтересованными сторонами, а также за обеспечение четкого понимания требований командой разработки.

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.