

Документ подписан посредством электронной подписи
Информация о владельце:
ФИО: Шамрай-Курбатова Лидия Викторовна
Должность: Ректор
Дата подписания: 11.06.2026 14:05:44
Уникальный программный ключ:
b1e4399771b07e18f31755456972d73b2ccfc531

Автономная некоммерческая организация высшего образования
«Волгоградский институт бизнеса»

Рабочая программа учебной дисциплины

Программная инженерия

(Наименование дисциплины)

54.03.01 Дизайн, направленность (профиль) «Цифровой дизайн»

(Направление подготовки / Профиль)

Бакалавр

(Квалификация)

Кафедра разработчик

Экономики и управления

Год набора

2026

Вид учебной деятельности	Трудоемкость (объем) дисциплины	
	Очная форма	Очно-заочная форма
	д	в
Зачетные единицы	6	6
Общее количество часов	216	216
Аудиторные часы контактной работы обучающегося с преподавателями:	96	74
– Лекционные (Л)	32	24
– Практические (ПЗ)		
– Лабораторные (ЛЗ)	64	50
– Семинарские (СЗ)		
Самостоятельная работа обучающихся (СРО)	84	106
К (Р-Г) Р (П) (+;-)	+	+
Тестирование (+;-)		
ДКР (+;-)		
Зачет (+;-)	+	+
Зачет с оценкой (+;- (Кол-во часов))		
Экзамен (+;- (Кол-во часов))	+(36)	+(36)

Волгоград 2026

Содержание

Раздел 1. Организационно-методический раздел	3
Раздел 2. Тематический план.....	5
Раздел 3. Содержание дисциплины.....	7
Раздел 4. Организация самостоятельной работы обучающихся.....	16
Раздел 5. Фонд оценочных средств для проведения текущего контроля и промежуточной аттестации обучающихся.....	18
Раздел 6. Оценочные средства промежуточной аттестации (с ключами)	19
Раздел 7. Перечень учебной литературы, необходимой для освоения дисциплины	32
Раздел 8. Материально-техническая база и информационные технологии.....	36
Раздел 9. Методические указания для обучающихся по освоению дисциплины	38

Раздел 1. Организационно-методический раздел

1.1. Цели освоения дисциплины

Дисциплина «Программная инженерия» входит в Часть дисциплин, формируемую участниками образовательных отношений для подготовки обучающихся по направлению подготовки 54.03.01 Дизайн, направленность (профиль) «Цифровой дизайн».

Целью дисциплины является формирование **компетенций** (в соответствии с ФГОС ВО и требованиями к результатам освоения основной профессиональной образовательной программы высшего образования (ОПОП ВО)):

ПК-3. Способен выполнять авторский надзор за выполнением работ по изготовлению объектов визуальной информации, идентификации и коммуникации (ПС 11.013 Графический дизайнер код В/03.6).

Дескрипторы профессиональной компетенции:

ПК-3.1. Способен выполнять авторский надзор за соблюдением технологий.

ПК-3.2. Способен контролировать качество дизайн-проектов.

ПК-4. Способен к художественно-технической разработке дизайн-проектов объектов визуальной информации, идентификации и коммуникации

ПК-4.1. Способен использовать специализированное ПО для проектирования.

ПК-4.2. Способен подготовить дизайн-проект с учетом современных технологий реализации.

Перечисленные компетенции формируются в процессе достижения **индикаторов компетенций**:

Обобщенная трудовая функция/ трудовая функция	Код и наименование дескриптора компетенций	Код и наименование индикатора достижения компетенций (из ПС)
ПК-3. Способен выполнять авторский надзор за выполнением работ по изготовлению объектов визуальной информации, идентификации и коммуникации (ПС 11.013 Графический дизайнер код В/03.6)	ПК-3.1. Способен выполнять авторский надзор за соблюдением технологий ПК-3.2. Способен контролировать качество дизайн-проектов	Знание: ИД-1 ПК-3.2 Технологические процессы производства в области полиграфии, упаковки, кино и телевидения В/03.6 ИД-2 ПК-3.2 Нормативные документы в области качества объектов визуальной информации, идентификации и коммуникации В/03.6 Умения: ИД-3 ПК-3.1 Учитывать при проектировании объектов визуальной информации, идентификации и коммуникации свойства используемых материалов и технологии реализации дизайн-проектов В/02.6 ИД-4 ПК-3.2 Применять показатели и средства контроля качества воспроизведения объектов визуальной информации, идентификации и коммуникации для авторского надзора за их изготовлением в производстве В/03.6. Навыки и (или)опыт деятельности: ИД-5 ПК-3.1 Выбор показателей, необходимых для проверки качества изготовления в производстве проектируемого объекта визуальной информации, идентификации и коммуникации В/03.6 ИД-6 ПК-3.2 Проведение проверки качества изготовления проектируемого объекта визуальной информации, идентификации и коммуникации по выбранным

		показателям В/03.6
ПК-4 Способен к технической разработке дизайн-проектов объектов визуальной информации, идентификации и коммуникации (ПС 11.013 Графический дизайнер код В/02.6)	ПК-4.1 Способен использовать специализированное ПО для проектирования ПК-4.2 Способен подготовить дизайн-проект с учетом современных технологий реализации	Знание: ИД-1 ПК-4.1 Компьютерное программное обеспечение, используемое в дизайне объектов визуальной информации, идентификации и коммуникации В/02.6 ИД-2 ПК-4.2 Технологические процессы производства в области полиграфии, упаковки, кино и телевидения В/02.6 Умения: ИД-3 ПК-4.1 Использовать специальные компьютерные программы для проектирования объектов визуальной информации, идентификации и коммуникации В/02.6 ИД-4 ПК-4.2 Учитывать при проектировании объектов визуальной информации, идентификации и коммуникации свойства используемых материалов и технологии реализации дизайн-проектов В/02.6 Навыки и (или)опыт деятельности: ИД-5 ПК-4.1 Разработка дизайн-макета объекта визуальной информации, идентификации и коммуникации В/02.6 ИД-6 ПК-4.2 Подготовка графических материалов для передачи в производство В/02.6

**1.2. Место дисциплины в структуре ОПОП ВО
направления подготовки 54.03.01 Дизайн, направленность (профиль) «Цифровой дизайн»**

№	Предшествующие дисциплины (дисциплины, изучаемые параллельно)	Последующие дисциплины
1	2	3
1	Компьютерная графика	
2	Графический дизайн	
3	Программирование для дизайна	
4	Тестирование дизайн-систем	
5	Современные архитектуры нейронных сетей для цифрового дизайна	
6	Разработка клиент-серверных приложений	
7	Эргономика	
8	Анимация интерфейсов	
9	Моушн-дизайн	

Последовательность формирования компетенций в указанных дисциплинах может быть изменена в зависимости от формы и срока обучения, а также преподавания с использованием дистанционных технологий обучения.

1.3. Нормативная документация

Рабочая программа учебной дисциплины составлена на основе:

- Федерального государственного образовательного стандарта высшего образования по направлению подготовки **54.03.01 Дизайн**;
- Учебного плана направления подготовки **54.03.01 Дизайн, направленность (профиль) «Цифровой дизайн»** 2026 года набора;
- Образца рабочей программы учебной дисциплины (приказ № 113-О от 01.09.2021 г.).

Раздел 2. Тематический план

Очная форма обучения (полный срок)

№	Тема дисциплины	Трудоемкость				Код индикатора и дескриптора достижения компетенций
		Всего	Аудиторные занятия		СРО	
			Л	ПЗ (ЛЗ, СЗ)		
1	2	3	4	5	6	7
1	Основы программной инженерии и профессиональная практика разработчика	14	2	4	8	ИД-1 ПК-4.1 ИД-2 ПК-4.2
2	Инженерия программных требований	14	2	4	8	ИД-3 ПК-4.1 ИД-4 ПК-4.2
3	Архитектура и проектирование программного обеспечения	14	2	4	8	ИД-5 ПК-4.1 ИД-6 ПК-4.2
4	Конструирование программного обеспечения и стандарты кодирования	14	2	4	8	ИД-3 ПК-4.1 ИД-6 ПК-4.2
5	Тестирование и обеспечение качества программного обеспечения	14	2	4	8	ИД-3 ПК-4.1 ИД-4 ПК-4.2
6	Управление конфигурациями и системами контроля версий	14	2	4	8	ИД-5 ПК-4.1 ИД-6 ПК-4.2
7	Сопровождение программного обеспечения и управление релизами	14	2	4	8	ИД-3 ПК-4.1 ИД-6 ПК-4.2
8	Управление проектами разработки и экономика программного обеспечения	10	2	4	4	ИД-5 ПК-4.1 ИД-6 ПК-4.2
Вид промежуточной аттестации (Зачет)		0				
		108	16	32	60	
9	Особенности проектирования цифровых продуктов: от концепции к реализации	9	2	4	3	ИД-1 ПК-4.1 ИД-2 ПК-4.2
10	Версионирование и управление артефактами цифрового продукта	9	2	4	3	ИД-3 ПК-4.1 ИД-4 ПК-4.2
11	Спецификации передачи дизайна в разработку (handoff)	9	2	4	3	ИД-5 ПК-4.1 ИД-6 ПК-4.2
12	Тестирование цифровых продуктов: валидация дизайна и пользовательского опыта	9	2	4	3	ИД-3 ПК-4.1 ИД-6 ПК-4.2
13	Релиз-менеджмент цифровых продуктов: подготовка, деплой, мониторинг	9	2	4	3	ИД-3 ПК-4.1 ИД-4 ПК-4.2
14	Интеграция ИИ-компонентов в цифровые продукты: инженерные аспекты	9	2	4	3	ИД-5 ПК-4.1 ИД-6 ПК-4.2
15	Обеспечение качества и безопасность цифровых продуктов	9	2	4	3	ИД-3 ПК-4.1 ИД-6 ПК-4.2
16	Интеграция инженерных артефактов	9	2	4	3	ИД-5 ПК-4.1 ИД-6 ПК-4.2
Вид промежуточной аттестации (Экзамен)		36	16	32	24	
Итого		216	32	64	84	

Очно-заочная форма обучения (полный срок)

№	Тема дисциплины	Трудоемкость				Код индикатора и дескриптора достижения компетенций
		Всего	Аудиторные занятия		СРО	
			Л	ПЗ (ЛЗ, СЗ)		
1	2	3	4	5	6	7
1	Основы программной инженерии и профессиональная практика разработчика	14	2	2	10	ИД-1 ПК-4.1 ИД-2 ПК-4.2
2	Инженерия программных требований	14	2	4	8	ИД-3 ПК-4.1 ИД-4 ПК-4.2
3	Архитектура и проектирование программного обеспечения	14	1	4	9	ИД-5 ПК-4.1 ИД-6 ПК-4.2
4	Конструирование программного обеспечения и стандарты кодирования	14	1	2	11	ИД-3 ПК-4.1 ИД-6 ПК-4.2
5	Тестирование и обеспечение качества программного обеспечения	14	1	4	9	ИД-3 ПК-4.1 ИД-4 ПК-4.2
6	Управление конфигурациями и системами контроля версий	14	1	2	11	ИД-5 ПК-4.1 ИД-6 ПК-4.2
7	Сопровождение программного обеспечения и управление релизами	14	1	2	11	ИД-3 ПК-4.1 ИД-6 ПК-4.2
8	Управление проектами разработки и экономика программного обеспечения	10	1	2	7	ИД-5 ПК-4.1 ИД-6 ПК-4.2
Вид промежуточной аттестации (Зачет)		0				
		108	10	22	76	
9	Особенности проектирования цифровых продуктов: от концепции к реализации	9	2	4	3	ИД-1 ПК-4.1 ИД-2 ПК-4.2
10	Версионирование и управление артефактами цифрового продукта	9	2	4	3	ИД-3 ПК-4.1 ИД-4 ПК-4.2
11	Спецификации передачи дизайна в разработку (handoff)	9	2	4	3	ИД-5 ПК-4.1 ИД-6 ПК-4.2
12	Тестирование цифровых продуктов: валидация дизайна и пользовательского опыта	9	2	4	3	ИД-3 ПК-4.1 ИД-6 ПК-4.2
13	Релиз-менеджмент цифровых продуктов: подготовка, деплой, мониторинг	9	2	4	3	ИД-3 ПК-4.1 ИД-4 ПК-4.2
14	Интеграция ИИ-компонентов в цифровые продукты: инженерные аспекты	9	2	4	3	ИД-5 ПК-4.1 ИД-6 ПК-4.2
15	Обеспечение качества и безопасность цифровых продуктов	9	2	4	3	ИД-3 ПК-4.1 ИД-6 ПК-4.2
16	Интеграция инженерных артефактов	9			9	ИД-5 ПК-4.1 ИД-6 ПК-4.2
Вид промежуточной аттестации (Экзамен)		36	14	28	30	
Итого		216	24	50	106	

Раздел 3. Содержание дисциплины

3.1. Содержание дисциплины

Тема 1. Основы программной инженерии

Определение программной инженерии как инженерной дисциплины, её цели, задачи и отличия от эмпирического программирования. Обзор структуры SWEBOOK: базовые области знаний, инженерные процессы, стандарты ИСО и ГОСТ. Жизненный цикл программного обеспечения: модели (каскадная, инкрементальная, спиральная, гибкие методологии), критерии выбора в зависимости от сложности, стабильности требований и состава команды. Роль инженера-программиста в проектной команде: ответственность за воспроизводимость артефактов, контроль версий, документирование и соблюдение стандартов качества. Нормативно-правовой контекст: лицензирование программного обеспечения, защита интеллектуальной собственности, требования информационной безопасности (152-ФЗ, ГОСТ Р ИСО/МЭК 27001). Профессиональная этика: прозрачность разработки, атрибуция сторонних компонентов, предотвращение уязвимостей и деструктивных алгоритмов.

Тема 2. Инженерия программных требований

Классификация требований: функциональные, нефункциональные (производительность, безопасность, надёжность, совместимость, масштабируемость), системные и пользовательские ограничения. Методы выявления: интервью, анализ предметной области, сценарный анализ, прототипирование в контексте технического задания. Формализация и спецификация требований: стандарты оформления, трассируемость, приоритизация. Управление изменениями требований: комиссия по контролю изменений, версионирование спецификаций, анализ влияния изменений на архитектуру и бюджет. Соответствие ГОСТ 34.602-89: структура технического задания, требования к системе, состав работ, контроль и приёмка. Валидация и верификация требований на ранних стадиях жизненного цикла, предотвращение двусмысленности и конфликтов спецификаций.

Тема 3. Архитектура и проектирование программного обеспечения.

Принципы архитектурного проектирования: модульность, связность, слабая связанность, абстрагирование, повторное использование. Паттерны проектирования: структурные, поведенческие, порождающие, их применение в решении типовых инженерных задач. Моделирование архитектуры: диаграммы UML, модель C4, архитектурные виды (логический, процессный, физический). Документирование архитектуры: архитектурные решения, спецификации интерфейсов (контракты API, стандарт OpenAPI), ограничения и стандарты кодирования. Оценка архитектурных решений: анализ рисков, прототипирование ключевых модулей, метрики технической сложности, баланс между гибкостью и производительностью.

Тема 4. Конструирование программного обеспечения и стандарты кодирования.

Конструирование как фаза жизненного цикла: трансляция архитектурных решений в работоспособный код. Стандарты кодирования и стилистические конвенции: читаемость, именование, форматирование, документирование исходного кода. Инструменты статического анализа и автоматизированные проверки синтаксиса: выявление дефектов структуры кода, метрики качества (цикломатическая сложность, объём дублирования, технический долг). Практики безопасного программирования: защита от инъекций, переполнений буфера, межсайтового скриптинга, управление секретами и конфигурациями. Интеграция анализа в процесс разработки: проверки перед фиксацией изменений, конвейеры непрерывной интеграции, автоматическое форматирование, обеспечение детерминированности сборок.

Тема 5. Тестирование и обеспечение качества программного обеспечения.

Концепция качества программного обеспечения: верификация и валидация, модель ИСО/МЭК 25010. Уровни тестирования: модульное, интеграционное, системное, приёмочное.

Виды тестирования: функциональное, регрессионное, нагрузочное, безопасность, совместимость. Методы проектирования тестов: эквивалентное разбиение, анализ граничных значений, таблицы решений, диаграммы состояний, попарное тестирование. Автоматизация тестирования: программные каркасы, имитация зависимостей, фикстуры. Метрики покрытия кода, отчётность о дефектах, жизненный цикл исправлений, приоритизация устранения. Процессы экспертизы кода и коллегиального ревью как инструмент контроля качества и обмена знаниями в команде.

Тема 6. Управление конфигурациями и системами контроля версий.

Основы управления конфигурациями программного обеспечения: идентификация, контроль изменений, учёт и аудит. Архитектура распределённых систем контроля версий: рабочее дерево, индекс, репозиторий, ветвление, слияние, откат. Стратегии ветвления: основной поток разработки, потоковое ветвление, централизованная модель разработки, критерии выбора в зависимости от модели релизов и частоты интеграции. Семантическое версионирование: правила инкремента версий, управление зависимостями, обеспечение обратной совместимости. Стандартизированные сообщения коммитов, автоматическая генерация журналов изменений, привязка к задачам в трекере. Работа с зависимостями и пакетами: репозитории, файлы фиксации версий, аудит уязвимостей, воспроизводимость сред разработки.

Тема 7. Сопровождение программного обеспечения и управление релизами.

Фаза сопровождения программного обеспечения: классификация изменений (корректирующие, адаптивные, совершенствующие, превентивные), оценка стоимости поддержки на жизненном цикле, управление техническим долгом. Процессы управления релизами: подготовка сборки, заморозка кода, финальное тестирование, артефакты релиза, развёртывание в изолированные окружения. Стратегии развёртывания: параллельное переключение, постепенное внедрение, последовательное обновление. Управление конфигурациями окружений, переменные среды, контейнеризация. Откат релизов и выпуск срочных исправлений: процедуры, анализ инцидентов, пост-релизный анализ, фиксация инженерных уроков. Автоматизация конвейеров непрерывной интеграции и доставки: триггеры, этапы сборки, тестирования, упаковки, уведомления, управление секретами и артефактами.

Тема 8. Управление проектами разработки и экономика программного обеспечения.

Основы управления проектами в программной инженерии: планирование, оценка трудоёмкости, распределение ресурсов, управление рисками и коммуникациями. Методы оценки: экспертная, по аналогии, метод анализа и оценки программ, функциональные точки, условные единицы трудоёмкости, скорость команды. Гибкое управление проектами: артефакты гибкой методологии, реестр требований, итерации, ретроспективы, метрики эффективности команды. Экономика программного обеспечения: расчёт себестоимости разработки, анализ окупаемости инвестиций, общая стоимость владения, учёт лицензий, инфраструктуры и трудозатрат. Нормативное регулирование: стандарты качества, сертификация, аудит соответствия. Этика и ответственность: прозрачность метрик, баланс скорости и качества, академическое и профессиональное цитирование алгоритмических решений, документирование принятых инженерных решений.

Тема 9. Особенности проектирования цифровых продуктов: от концепции к реализации.

Специфика цифровых продуктов как объекта программной инженерии: интерактивность, адаптивность, кросс-платформенность, зависимость от пользовательского опыта. Этапы проектирования: концептуализация, прототипирование, спецификация, передача в разработку. Роль дизайнера в инженерном цикле: ответственность за handoff-готовность, поддержку спецификаций, авторский надзор. Интеграция требований доступности (ГОСТ Р 52872-2019, рекомендации) на этапе проектирования. Документирование проектных решений: спецификации компонентов, состояния, правила адаптивности.

Тема 10. Версионирование и управление артефактами цифрового продукта.

Применение систем контроля версий к дизайн-артефактам: макеты, спецификации, токены, прототипы. Стратегии ветвления для дизайн-разработки: синхронизация с кодовой базой, управление параллельными изменениями. Семантическое версионирование для компонентов интерфейса: правила инкремента версий при изменении поведения, визуала, доступности. Автоматизация документирования изменений: генерация журналов обновлений, привязка к задачам трекера. Воспроизводимость сборок: фиксация версий зависимостей, конфигураций окружений, инструментов проектирования.

Тема 11. Спецификации передачи дизайна в разработку (handoff).

Структура handoff-документации: описание компонентов, пропсы, состояния, примеры использования, правила адаптивности. Форматы спецификаций: текстовые (Markdown), визуальные (интерактивные прототипы), машиночитаемые (JSON-токены). Инструменты автоматизации: генерация спецификаций из макетов, синхронизация с репозиторием кода. Контроль качества передачи: чек-листы полноты, актуальности, однозначности спецификаций. Процессы обратной связи: фиксация расхождений, итеративное уточнение, протоколирование решений.

Тема 12. Тестирование цифровых продуктов: валидация дизайна и пользовательского опыта.

Виды тестирования, релевантные для цифровых продуктов: визуальное регрессионное, доступность, адаптивность, кросс-браузерное, юзабилити. Методы проектирования тестовых сценариев: на основе пользовательских сценариев, требований доступности, бизнес-правил. Автоматизация визуального тестирования: инструменты сравнения скриншотов, детекция изменений. Интеграция тестов доступности в процесс разработки: автоматические проверки, ручная валидация. Метрики качества пользовательского опыта: время выполнения задач, уровень ошибок, удовлетворённость.

Тема 13. Релиз-менеджмент цифровых продуктов: подготовка, деплой, мониторинг.

Особенности релиза цифровых продуктов: артефакты (код, ассеты, документация), окружения (dev/stage/prod), стратегии развёртывания. Чек-листы готовности релиза: визуальная целостность, соответствие спецификациям, тестирование доступности, документация. Автоматизация сборки и деплоя: конвейеры непрерывной интеграции для дизайн-артефактов, управление версиями ассетов. Пост-релизный мониторинг: сбор метрик производительности, анализ пользовательского поведения, детекция регрессий. Процедуры отката и выпуска исправлений: сценарии, инструменты, коммуникация.

Тема 14. Интеграция ИИ-компонентов в цифровые продукты: инженерные аспекты.

Особенности проектирования систем с ИИ-компонентами: недетерминированность, зависимость от данных, сложность валидации. Архитектурные подходы: выделение сервисов инференса, управление версиями моделей, мониторинг качества. Интерфейсы взаимодействия: спецификации API, форматы запросов/ответов, обработка ошибок. Документирование ИИ-компонентов: описание логики, ограничений, метрик качества, этических аспектов. Атрибуция ИИ-генерируемого контента: требования к прозрачности, академическое цитирование, нормативное регулирование.

Тема 15. Обеспечение качества и безопасность цифровых продуктов.

Стандарты качества для цифровых продуктов: функциональность, надёжность, удобство использования, эффективность, сопровождаемость, переносимость. Методы обеспечения качества: статический анализ спецификаций, ревью дизайн-решений, автоматизированные проверки доступности. Безопасность цифровых продуктов: защита персональных данных (152-ФЗ), безопасная передача артефактов, аудит зависимостей. Управление инцидентами: мониторинг, реагирование, пост-релизный анализ, фиксация уроков. Этические аспекты: предотвращение манипулятивных паттернов, прозрачность алгоритмических решений, инклюзивность.

Тема 16. Интеграция инженерных артефактов

Структура ВКР по направлению «Цифровой дизайн»: требования к разделам, объёму, стилю, научной новизне, практической значимости. Инженерные артефакты как основа графической части: диаграммы, спецификации, чек-листы, отчёты о тестировании. Нормоконтроль: проверка оформления по ГОСТ, библиографии, перекрёстных ссылок, нумерации, метаданных. Атрибуция сторонних компонентов и ИИ-контента: требования к декларированию, методы проверки уникальности, академическая этика. Подготовка защитной презентации: визуальная нарративность, тайминг, работа с возражениями, демонстрация артефактов.

3.2. Содержание практического блока дисциплины

Очная форма обучения (полный срок)

№	Тема практического (семинарского, лабораторного) занятия
1	2
ПЗ 1-2	Основы программной инженерии и профессиональная практика разработчика. Содержание практического занятия: анализ структуры инженерных процессов согласно стандартам ИСО и ГОСТ; распределение ролей в проектной команде; разработка устава учебного проекта; формирование чек-листа инженерной культуры (воспроизводимость артефактов, документирование, контроль версий); оформление отчёта с матрицей ответственности и реестром нормативных требований к лицензированию и информационной безопасности.
ПЗ 3-4	Инженерия программных требований. Содержание практического занятия: сбор и классификация функциональных и нефункциональных требований; приоритизация и верификация; разработка технического задания в соответствии с ГОСТ 34.602-89; построение матрицы трассируемости требований; оформление комплекта документации с разделами: общие сведения, назначение, требования к системе, состав работ, контроль и приёмка, документирование.
ПЗ 5-6	Архитектура и проектирование программного обеспечения. Содержание практического занятия: декомпозиция системы на модули и компоненты; разработка пакета архитектурных диаграмм (UML: компонентов, развёртывания, последовательностей; логический и физический виды); фиксация архитектурных решений; оценка рисков и метрик технической сложности; оформление отчёта со спецификацией интерфейсов взаимодействия и обоснованием выбранного архитектурного стиля.
ПЗ 7-8	Конструирование программного обеспечения и стандарты кодирования. Содержание практического занятия: трансляция архитектурных решений в работоспособный код; настройка статического анализа и автоматизированных проверок синтаксиса; расчёт метрик качества (цикломатическая сложность, объём дублирования, технический долг); реализация практик безопасного программирования; оформление отчёта о соответствии стандартам кодирования, результатах линтинга и плане рефакторинга.
ПЗ 9-10	Тестирование и обеспечение качества программного обеспечения. Содержание практического занятия: проектирование уровней и видов тестирования; применение техник эквивалентного разбиения и анализа граничных значений; разработка Программы и методики испытаний (ПМИ); расчёт метрик покрытия кода; оформление отчёта с комплектом документов: ПМИ, протокол тестового прогона, реестр дефектов, план устранения.
ПЗ 11-12	Управление конфигурациями и системами контроля версий. Содержание практического занятия: настройка репозитория распределённой системы контроля версий; реализация стратегии ветвления; стандартизация

	сообщений коммитов; настройка автоматической генерации журналов изменений; версионирование зависимостей и конфигураций; оформление отчёта со схемой ветвления, шаблоном коммита, реестром версий артефактов и инструкцией по работе с репозиторием.
ПЗ 13-14	Сопровождение программного обеспечения и управление релизами. Содержание практического занятия: классификация изменений и оценка стоимости сопровождения; планирование выпуска стабильной версии; разработка релизного чек-листа; настройка процедуры отката и выпуска срочных исправлений; анализ инцидентов; оформление отчёта с релизным чек-листом, инструкцией по развёртыванию, шаблоном журнала изменений и планом пост-релизного анализа.
ПЗ 15-16	Управление проектами разработки и экономика программного обеспечения. Содержание практического занятия: построение иерархической структуры работ (ИСР); оценка трудоёмкости и распределение ресурсов; разработка календарного плана (диаграмма Ганта); идентификация и анализ рисков; расчёт экономической эффективности (общая стоимость владения, окупаемость инвестиций); оформление отчёта с реестром рисков, планом реагирования и финансовым обоснованием проекта.
ПЗ 17-18	Особенности проектирования цифровых продуктов: от концепции к реализации. Содержание практического занятия: анализ пользовательских сценариев и бизнес-ограничений; разработка диаграмм пользовательских потоков и состояний интерфейса; спецификация правил адаптивности и информационной архитектуры; интеграция требований доступности; оформление отчёта с пакетом проектных диаграмм, спецификацией состояний и матрицей соответствия стандартам доступности.
ПЗ 19-20	Версионирование и управление артефактами цифрового продукта. Содержание практического занятия: организация структуры репозитория для хранения проектных артефактов; настройка семантического версионирования для компонентов интерфейса; привязка изменений к задачам в трекере; автоматизация документирования обновлений; оформление отчёта со схемой версионирования, шаблоном журнала изменений и инструкцией по синхронизации проектных и кодовых баз.
ПЗ 21-22	Спецификации передачи дизайна в разработку. Содержание практического занятия: документирование компонентов, состояний и правил адаптивности; подготовка машиночитаемых и визуальных спецификаций; разработка чек-листа полноты и однозначности передачи; организация процесса обратной связи; оформление отчёта с комплектом спецификаций передачи, протоколом согласования и реестром выявленных расхождений.
ПЗ 23-24	Тестирование цифровых продуктов: валидация пользовательского опыта. Содержание практического занятия: разработка сценариев юзабилити-тестирования и проверки доступности; составление Программы и методики испытаний цифрового интерфейса; проведение автоматизированных и ручных проверок; расчёт метрик выполнения задач и уровня ошибок; оформление отчёта с ПМИ, протоколами тестирования, аудитом доступности и рекомендациями по устранению дефектов.
ПЗ 25-26	Релиз-менеджмент цифровых продуктов: подготовка, деплой, мониторинг. Содержание практического занятия: подготовка чек-листа готовности цифрового релиза; настройка конвейера сборки и развёртывания в изолированные окружения; планирование пост-релизного мониторинга и сбора метрик; разработка процедуры отката; оформление отчёта с релизным чек-листом, конфигурацией окружений, журналом выпуска и планом мониторинга.
ПЗ 27-28	Интеграция ИИ-компонентов в цифровые продукты: инженерные аспекты. Содержание практического занятия: спецификация интерфейсов взаимодействия с интеллектуальными сервисами; разработка методики валидации качества

	предсказаний; документирование ограничений, метрик и этических аспектов; оформление отчёта с контрактом взаимодействия, методикой валидации, протоколом атрибуции генерированного контента и планом мониторинга дрейфа.
ПЗ 29-30	Обеспечение качества и безопасность цифровых продуктов. Содержание практического занятия: проведение статического анализа спецификаций и аудит зависимостей; разработка программы испытаний на соответствие требованиям информационной безопасности и защиты персональных данных; выявление уязвимостей; оформление отчёта с ПМИ по безопасности, результатами аудита, отчётом о статическом анализе и плане устранения рисков.
ПЗ 31-32	Интеграция инженерных артефактов в выпускную квалификационную работу. Содержание практического занятия: структурирование накопленных инженерных материалов (диаграммы, ТЗ, ПМИ, спецификации, чек-листы) для включения в пояснительную записку; проверка оформления на соответствие требованиям нормоконтроля; подготовка графической части; оформление отчёта с оглавлением приложений ВКР, чек-листом самопроверки, протоколом нормоконтроля и планом доработок.

Очно-заочная форма обучения (полный срок)

ПЗ 1	Основы программной инженерии и профессиональная практика разработчика. Содержание практического занятия: анализ структуры инженерных процессов согласно стандартам ИСО и ГОСТ; распределение ролей в проектной команде; разработка устава учебного проекта; формирование чек-листа инженерной культуры (воспроизводимость артефактов, документирование, контроль версий); оформление отчёта с матрицей ответственности и реестром нормативных требований к лицензированию и информационной безопасности.
ПЗ 2-3	Инженерия программных требований. Содержание практического занятия: сбор и классификация функциональных и нефункциональных требований; приоритизация и верификация; разработка технического задания в соответствии с ГОСТ 34.602-89; построение матрицы трассируемости требований; оформление комплекта документации с разделами: общие сведения, назначение, требования к системе, состав работ, контроль и приёмка, документирование.
ПЗ 4-5	Архитектура и проектирование программного обеспечения. Содержание практического занятия: декомпозиция системы на модули и компоненты; разработка пакета архитектурных диаграмм (UML: компонентов, развёртывания, последовательностей; логический и физический виды); фиксация архитектурных решений; оценка рисков и метрик технической сложности; оформление отчёта со спецификацией интерфейсов взаимодействия и обоснованием выбранного архитектурного стиля.
ПЗ 6	Конструирование программного обеспечения и стандарты кодирования. Содержание практического занятия: трансляция архитектурных решений в работоспособный код; настройка статического анализа и автоматизированных проверок синтаксиса; расчёт метрик качества (цикломатическая сложность, объём дублирования, технический долг); реализация практик безопасного программирования; оформление отчёта о соответствии стандартам кодирования, результатах линтинга и плане рефакторинга.
ПЗ 7-8	Тестирование и обеспечение качества программного обеспечения. Содержание практического занятия: проектирование уровней и видов тестирования; применение техник эквивалентного разбиения и анализа граничных значений; разработка Программы и методики испытаний (ПМИ); расчёт метрик покрытия кода; оформление отчёта с комплектом документов: ПМИ, протокол тестового прогона, реестр дефектов, план устранения.
ПЗ 9	Управление конфигурациями и системами контроля версий.

	Содержание практического занятия: настройка репозитория распределённой системы контроля версий; реализация стратегии ветвления; стандартизация сообщений коммитов; настройка автоматической генерации журналов изменений; версионирование зависимостей и конфигураций; оформление отчёта со схемой ветвления, шаблоном коммита, реестром версий артефактов и инструкцией по работе с репозиторием.
ПЗ 10	Сопровождение программного обеспечения и управление релизами. Содержание практического занятия: классификация изменений и оценка стоимости сопровождения; планирование выпуска стабильной версии; разработка релизного чек-листа; настройка процедуры отката и выпуска срочных исправлений; анализ инцидентов; оформление отчёта с релизным чек-листом, инструкцией по развёртыванию, шаблоном журнала изменений и планом пост-релизного анализа.
ПЗ 11	Управление проектами разработки и экономика программного обеспечения. Содержание практического занятия: построение иерархической структуры работ (ИСР); оценка трудоёмкости и распределение ресурсов; разработка календарного плана (диаграмма Ганта); идентификация и анализ рисков; расчёт экономической эффективности (общая стоимость владения, окупаемость инвестиций); оформление отчёта с реестром рисков, планом реагирования и финансовым обоснованием проекта.
ПЗ 12-13	Особенности проектирования цифровых продуктов: от концепции к реализации. Содержание практического занятия: анализ пользовательских сценариев и бизнес-ограничений; разработка диаграмм пользовательских потоков и состояний интерфейса; спецификация правил адаптивности и информационной архитектуры; интеграция требований доступности; оформление отчёта с пакетом проектных диаграмм, спецификацией состояний и матрицей соответствия стандартам доступности.
ПЗ 14-15	Версионирование и управление артефактами цифрового продукта. Содержание практического занятия: организация структуры репозитория для хранения проектных артефактов; настройка семантического версионирования для компонентов интерфейса; привязка изменений к задачам в трекере; автоматизация документирования обновлений; оформление отчёта со схемой версионирования, шаблоном журнала изменений и инструкцией по синхронизации проектных и кодовых баз.
ПЗ 16-17	Спецификации передачи дизайна в разработку. Содержание практического занятия: документирование компонентов, состояний и правил адаптивности; подготовка машиночитаемых и визуальных спецификаций; разработка чек-листа полноты и однозначности передачи; организация процесса обратной связи; оформление отчёта с комплектом спецификаций передачи, протоколом согласования и реестром выявленных расхождений.
ПЗ 18-19	Тестирование цифровых продуктов: валидация пользовательского опыта. Содержание практического занятия: разработка сценариев юзабилити-тестирования и проверки доступности; составление Программы и методики испытаний цифрового интерфейса; проведение автоматизированных и ручных проверок; расчёт метрик выполнения задач и уровня ошибок; оформление отчёта с ПМИ, протоколами тестирования, аудитом доступности и рекомендациями по устранению дефектов.
ПЗ 20-21	Релиз-менеджмент цифровых продуктов: подготовка, деплой, мониторинг. Содержание практического занятия: подготовка чек-листа готовности цифрового релиза; настройка конвейера сборки и развёртывания в изолированные окружения; планирование пост-релизного мониторинга и сбора метрик; разработка процедуры отката; оформление отчёта с релизным чек-листом, конфигурацией окружений, журналом выпуска и планом мониторинга.
ПЗ 22-23	Интеграция ИИ-компонентов в цифровые продукты: инженерные аспекты.

	Содержание практического занятия: спецификация интерфейсов взаимодействия с интеллектуальными сервисами; разработка методики валидации качества предсказаний; документирование ограничений, метрик и этических аспектов; оформление отчёта с контрактом взаимодействия, методикой валидации, протоколом атрибуции генерированного контента и планом мониторинга дрейфа.
ПЗ 24-25	Обеспечение качества и безопасность цифровых продуктов. Содержание практического занятия: проведение статического анализа спецификаций и аудит зависимостей; разработка программы испытаний на соответствие требованиям информационной безопасности и защиты персональных данных; выявление уязвимостей; оформление отчёта с ПМИ по безопасности, результатами аудита, отчётом о статическом анализе и плане устранения рисков.

3.3. Образовательные технологии

Очная форма обучения (полный срок)

№	Тема занятия	Вид учебного занятия	Форма / Методы интерактивного обучения	% учебного времени
1	2	3	4	5
1.	Основы программной инженерии и профессиональная практика разработчика	ПЗ	Метод проектов	100
2.	Инженерия программных требований	ПЗ	Метод проектов	100
3.	Архитектура и проектирование программного обеспечения	ПЗ	Метод проектов	100
4.	Конструирование программного обеспечения и стандарты кодирования	ПЗ	Метод проектов	100
5.	Тестирование и обеспечение качества программного обеспечения	ПЗ	Метод проектов	100
6.	Управление конфигурациями и системами контроля версий	ПЗ	Метод проектов	100
7.	Сопровождение программного обеспечения и управление релизами	ПЗ	Метод проектов	100
8.	Управление проектами разработки и экономика программного обеспечения	ПЗ	Метод проектов	100
9.	Особенности проектирования цифровых продуктов: от концепции к реализации	ПЗ	Метод проектов	100
10.	Версионирование и управление артефактами цифрового продукта	ПЗ	Метод проектов	100
11.	Спецификации передачи дизайна в разработку (handoff)	ПЗ	Метод проектов	100
12.	Тестирование цифровых продуктов: валидация дизайна и пользовательского опыта	ПЗ	Метод проектов	100
13.	Релиз-менеджмент цифровых продуктов: подготовка, деплой, мониторинг	ПЗ	Метод проектов	100
14.	Интеграция ИИ-компонентов в цифровые продукты: инженерные аспекты	ПЗ	Метод проектов	100
15.	Обеспечение качества и безопасность	ПЗ	Метод проектов	100

	цифровых продуктов			
16.	Интеграция инженерных артефактов	ПЗ	Метод проектов	100
Итого %				100%

Очно-заочная форма обучения (полный срок)

№	Тема занятия	Вид учебного занятия	Форма / Методы интерактивного обучения	% учебного времени
1	2	3	4	5
1.	Основы программной инженерии и профессиональная практика разработчика	ПЗ	Метод проектов	100
2.	Инженерия программных требований	ПЗ	Метод проектов	100
3.	Архитектура и проектирование программного обеспечения	ПЗ	Метод проектов	100
4.	Конструирование программного обеспечения и стандарты кодирования	ПЗ	Метод проектов	100
5.	Тестирование и обеспечение качества программного обеспечения	ПЗ	Метод проектов	100
6.	Управление конфигурациями и системами контроля версий	ПЗ	Метод проектов	100
7.	Сопровождение программного обеспечения и управление релизами	ПЗ	Метод проектов	100
8.	Управление проектами разработки и экономика программного обеспечения	ПЗ	Метод проектов	100
9.	Особенности проектирования цифровых продуктов: от концепции к реализации	ПЗ	Метод проектов	100
10.	Версионирование и управление артефактами цифрового продукта	ПЗ	Метод проектов	100
11.	Спецификации передачи дизайна в разработку (handoff)	ПЗ	Метод проектов	100
12.	Тестирование цифровых продуктов: валидация дизайна и пользовательского опыта	ПЗ	Метод проектов	100
13.	Релиз-менеджмент цифровых продуктов: подготовка, деплой, мониторинг	ПЗ	Метод проектов	100
14.	Интеграция ИИ-компонентов в цифровые продукты: инженерные аспекты	ПЗ	Метод проектов	100
15.	Обеспечение качества и безопасность цифровых продуктов	ПЗ	Метод проектов	100
Итого %				30%

Раздел 4. Организация самостоятельной работы обучающихся

4.1. Организация самостоятельной работы обучающихся

№	Тема дисциплины	№ вопросов	№ рекомендуемой литературы
1	2	3	4
	Основы программной инженерии и профессиональная практика разработчика	1	1-11
2.	Инженерия программных требований	2	1-11
3.	Архитектура и проектирование программного обеспечения	3	1-11
4.	Конструирование программного обеспечения и стандарты кодирования	4	1-11
5.	Тестирование и обеспечение качества программного обеспечения	5	1-11
6.	Управление конфигурациями и системами контроля версий	6	1-11
7.	Сопровождение программного обеспечения и управление релизами	7	1-11
8.	Управление проектами разработки и экономика программного обеспечения	8	1-11
9.	Особенности проектирования цифровых продуктов: от концепции к реализации	9	1-11
10.	Версионирование и управление артефактами цифрового продукта	10	1-11
11.	Спецификации передачи дизайна в разработку (handoff)	11	1-11
12.	Тестирование цифровых продуктов: валидация дизайна и пользовательского опыта	12	1-11
13.	Релиз-менеджмент цифровых продуктов: подготовка, деплой, мониторинг	13	1-11
14.	Интеграция ИИ-компонентов в цифровые продукты: инженерные аспекты	14	1-11
15.	Обеспечение качества и безопасность цифровых продуктов	15	1-11
16.	Интеграция инженерных артефактов	16	1-11

Перечень вопросов, выносимых на самостоятельную работу обучающихся

1. Каковы фундаментальные отличия программной инженерии от эмпирического программирования и какие нормативно-правовые стандарты регулируют профессиональную практику разработчика?
2. Каким образом обеспечивается сквозная трассируемость программных требований на протяжении всего жизненного цикла продукта и какие методы формализации исключают двусмысленность спецификаций?
3. Какие архитектурные паттерны и принципы модульного проектирования обеспечивают баланс между слабой связанностью компонентов и высокой производительностью системы?
4. Как стандарты кодирования и инструменты статического анализа влияют на снижение технического долга и повышение воспроизводимости программных сборок?
5. Каковы критерии выбора уровней и видов тестирования в зависимости от архитектуры системы и как рассчитываются метрики покрытия кода для обеспечения заданного качества?
6. Каким образом стратегии ветвления и семантическое версионирование обеспечивают

- обратную совместимость и минимизируют риски интеграции в распределённых командах?
7. Какие процедуры классификации изменений и стратегии управления релизами позволяют оптимизировать затраты на сопровождение программного продукта в пост-релизный период?
 8. Как методы оценки трудоёмкости и модели расчёта общей стоимости владения влияют на обоснование выбора методологии управления проектами разработки?
 9. Каковы принципы трансформации концептуальных пользовательских сценариев в формализованные требования к информационной архитектуре и адаптивным интерфейсам?
 10. Каким образом организуются процессы синхронизации и контроля версий неструктурированных артефактов в едином репозитории с обеспечением воспроизводимости сборок?
 11. Какие стандарты и форматы документации обеспечивают однозначную интерпретацию проектных решений командой разработки и минимизируют потери информации при передаче спецификаций?
 12. Как методология проектирования тестовых сценариев валидирует соответствие цифрового интерфейса требованиям цифровой доступности и юзабилити?
 13. Каковы инженерные процедуры подготовки, развёртывания и пост-релизного мониторинга цифровых продуктов в изолированных окружениях с учётом стратегий отката?
 14. Каким образом проектируются интерфейсы взаимодействия и методы валидации для недетерминированных ИИ-компонентов с обеспечением прозрачности и учётом дрейфа данных?
 15. Какие подходы к статическому анализу и аудиту зависимостей обеспечивают соответствие цифровых продуктов нормативным требованиям информационной безопасности и защиты данных?
 16. Каковы критерии отбора, структурирования и нормативного оформления инженерных артефактов для обеспечения академической целостности и практической значимости выпускной квалификационной работы?

4.2. Перечень учебно-методического обеспечения самостоятельной работы обучающихся

Самостоятельная работа обучающихся обеспечивается следующими учебно-методическими материалами:

1. Указаниями в рабочей программе по дисциплине (п.4.1.)
2. Лекционные материалы в составе учебно-методического комплекса по дисциплине
3. Заданиями и методическими рекомендациями по организации самостоятельной работы обучающихся в составе учебно-методического комплекса по дисциплине.
4. Глоссарием по дисциплине в составе учебно-методического комплекса по дисциплине.

Раздел 5. Фонд оценочных средств для проведения текущего контроля и промежуточной аттестации обучающихся

Фонд оценочных средств по дисциплине представляет собой совокупность контролирующих материалов, предназначенных для измерения уровня достижения обучающимися установленных результатов образовательной программы. ФОС по дисциплине используется при проведении оперативного контроля и промежуточной аттестации обучающихся. Требования к структуре и содержанию ФОС дисциплины регламентируются Положением о фонде оценочных материалов по программам высшего образования – программам бакалавриата, магистратуры.

5.1. Паспорт фонда оценочных средств Очная форма обучения (полный срок) Очно-заочная форма обучения (полный срок)

№	Контролируемые разделы (темы) дисциплины	Оценочные средства			
		Л	ПЗ (ЛЗ, СЗ)	СРО	Код индикатора и дескриптора достижения компетенций
1	2	3	4	5	6
1	Основы программной инженерии и профессиональная практика разработчика	ЛС	МП,33	ПРВ	ИД-1 ПК-4.1 ИД-2 ПК-4.2
2	Инженерия программных требований	ЛС	МП,33	ПРВ	ИД-3 ПК-4.1 ИД-4 ПК-4.2
3	Архитектура и проектирование программного обеспечения	ЛС	МП,33	ПРВ	ИД-5 ПК-4.1 ИД-6 ПК-4.2
4	Конструирование программного обеспечения и стандарты кодирования	ЛС	МП,33	ПРВ	ИД-3 ПК-4.1 ИД-6 ПК-4.2
5	Тестирование и обеспечение качества программного обеспечения	ЛС	МП,33	ПРВ	ИД-3 ПК-4.1 ИД-4 ПК-4.2
6	Управление конфигурациями и системами контроля версий	ЛС	МП,33	ПРВ	ИД-5 ПК-4.1 ИД-6 ПК-4.2
7	Сопровождение программного обеспечения и управление релизами	ЛС	МП,33	ПРВ	ИД-3 ПК-4.1 ИД-6 ПК-4.2
8	Управление проектами разработки и экономика программного обеспечения	ЛС, УО	УО,33	ПРВ, 33	ИД-5 ПК-4.1 ИД-6 ПК-4.2
9	Особенности проектирования цифровых продуктов: от концепции к реализации	ЛС	МП,33	ПРВ	ИД-1 ПК-4.1 ИД-2 ПК-4.2
10	Версионирование и управление артефактами цифрового продукта	ЛС	МП,33	ПРВ	ИД-3 ПК-4.1 ИД-4 ПК-4.2
11	Спецификации передачи дизайна в разработку (handoff)	ЛС	МП,33	ПРВ	ИД-5 ПК-4.1 ИД-6 ПК-4.2
12	Тестирование цифровых продуктов: валидация дизайна и пользовательского опыта	ЛС	МП,33	ПРВ	ИД-3 ПК-4.1 ИД-6 ПК-4.2
13	Релиз-менеджмент цифровых продуктов: подготовка, деплой, мониторинг	ЛС	МП,33	ПРВ	ИД-3 ПК-4.1 ИД-4 ПК-4.2
14	Интеграция ИИ-компонентов в цифровые продукты: инженерные аспекты	ЛС	МП,33	ПРВ	ИД-5 ПК-4.1 ИД-6 ПК-4.2
15	Обеспечение качества и безопасность цифровых продуктов	ЛС	УО,33	ПРВ, 33	ИД-3 ПК-4.1 ИД-6 ПК-4.2
16	Интеграция инженерных артефактов	ЛС, УО	УО,33	ПРВ, 33	ИД-5 ПК-4.1 ИД-6 ПК-4.2

Условные обозначения оценочных средств (Столбцы 3, 4, 5):

ЗЗ – Защита выполненных заданий (творческих, расчетных и т.д.), представление презентаций;

Т – Тестирование компьютерное;

УО – Устный (фронтальный, индивидуальный, комбинированный) опрос;

КР – Контрольная работа (аудиторные или домашние, индивидуальные, парные или групповые контрольные, самостоятельные работы, диктанты и т.д.);

К – Коллоквиум;

ПРВ – Проверка рефератов, отчетов, рецензий, аннотаций, конспектов, графического материала, эссе, переводов, решений заданий, выполненных заданий в электронном виде и т.д.;

ДИ – Деловая игра;

РИ – Ролевая игра;

КМ – Кейс-метод;

КС – Круглый стол;

КСМ – Компьютерная симуляция;

МШ – Метод мозгового штурма;

ЛС – Лекция-ситуация;

ЛК – Лекция-конференция;

ЛВ – Лекция-визуализация;

ПЛ – Проблемная лекция;

Д – Дискуссия, полемика, диспут, дебаты;

П – Портфолио;

ПВУ – Просмотр видеоуроков;

МП – Метод проектов.

5.2. Оценочные средства текущего контроля

Перечень практических (семинарских) заданий

Тема 1. Основы программной инженерии и профессиональная практика разработчика

Цель практики и её задачи: Сформировать представление о профессиональной среде разработчика, нормативной базе и инженерной культуре. *Задачи:* проанализировать стандарты ИСО/ГОСТ, распределить роли в команде, разработать устав проекта, сформировать чек-лист инженерной культуры.

Входные данные:

- Описание учебного проекта (цель, состав команды, технические ограничения)
- Выдержки из стандартов ИСО/МЭК 12207, ГОСТ 34.601-90, ГОСТ Р ИСО/МЭК 27001
- Шаблон устава проекта и матрицы ответственности в текстовом редакторе

Инструменты: Р7-Офис / Яндекс.Документы / Свободный текстовый процессор

Пошаговое задание:

1. Проанализировать описание проекта: выделить ключевые артефакты, этапы, ответственных
2. Распределить роли в команде (аналитик, архитектор, разработчик, тестировщик, руководитель)
3. Разработать устав проекта: цели, границы, критерии успеха, ограничения, риски
4. Сформировать чек-лист инженерной культуры: воспроизводимость, документирование, контроль версий, ревью
5. Составить реестр нормативных требований: лицензирование ПО, защита данных, информационная безопасность
6. Оформить отчёт с матрицей ответственности и обоснованием выбранных стандартов

Итоговый результат:

.pdf / .docx документ с уставом проекта, матрицей ответственности и реестром нормативных требований / Фамилия_ПЗ1_УставПроекта

.txt черновик распределения ролей и заметки по нормативным требованиям / Фамилия_ПЗ1_Черновик

Критерии оценки (чек-лист):

- [] Устав проекта содержит измеримые цели, чёткие границы и критерии успеха
- [] Матрица ответственности покрывает все ключевые артефакты и этапы проекта
- [] Чек-лист инженерной культуры включает не менее пяти практик с пояснениями
- [] Реестр нормативных требований привязан к конкретным статьям ГОСТ/ИСО

Тема 2. Инженерия программных требований

Цель практики и её задачи: Освоить методику сбора, классификации и формализации требований. *Задачи:* выявить функциональные и нефункциональные требования, приоритизировать их, разработать техническое задание по ГОСТ 34.602-89, построить матрицу трассируемости.

Входные данные:

- Интервью с условным заказчиком (сценарий использования, бизнес-ограничения)
- Примеры технических заданий из методических материалов
- Шаблон ТЗ по ГОСТ 34.602-89 в текстовом редакторе

Инструменты: Р7-Офис / Яндекс.Документы / Свободный текстовый процессор

Пошаговое задание:

1. Проанализировать сценарий использования: выделить действующих лиц, цели, ограничения
2. Сформулировать не менее 10 функциональных и 5 нефункциональных требований
3. Приоритизировать требования методом MoSCoW, обосновать выбор
4. Разработать техническое задание по структуре ГОСТ 34.602-89: общие сведения, назначение, требования к системе, состав работ, контроль и приёмка

5. Построить матрицу трассируемости: связь бизнес-цель → требование → компонент системы

6. Оформить комплект документации с оглавлением и ссылками на приложения

Итоговый результат:

.pdf / .docx техническое задание по ГОСТ 34.602-89 с матрицей трассируемости /
Фамилия_ПЗ2_ТЗ

.txt черновик требований и заметки по приоритизации / Фамилия_ПЗ2_Черновик

Критерии оценки (чек-лист):

- [] Требования сформулированы однозначно, проверяемо, без двусмысленности
- [] Структура ТЗ соответствует разделам ГОСТ 34.602-89
- [] Матрица трассируемости покрывает все требования и компоненты
- [] Приоритизация обоснована бизнес-ценностью и техническими ограничениями

Тема 3. Архитектура и проектирование программного обеспечения

Цель практики и её задачи: Научиться декомпозиции системы и документированию архитектурных решений. *Задачи:* выделить модули и компоненты, разработать пакет диаграмм, специфицировать интерфейсы, оценить риски.

Входные данные:

- Техническое задание из ПЗ 2
- Примеры UML-диаграмм и архитектурных описаний из методических материалов
- Шаблон отчёта по архитектурному проектированию

Инструменты: Draw.io / PlantUML / P7-Офис

Пошаговое задание:

1. Декомпонировать систему на модули и компоненты согласно требованиям ТЗ
2. Разработать диаграмму компонентов: границы модулей, интерфейсы, зависимости
3. Построить диаграмму развёртывания: узлы, артефакты, каналы связи
4. Специфицировать интерфейсы взаимодействия: сигнатуры методов, форматы данных, коды ответов
5. Оценить риски архитектурных решений: точки отказа, масштабируемость, сопровождаемость
6. Оформить отчёт с пакетом диаграмм и обоснованием выбранного архитектурного стиля

Итоговый результат:

.pdf / .docx отчёт с диаграммами компонентов, развёртывания и спецификацией интерфейсов /
Фамилия_ПЗ3_Архитектура

.txt черновик декомпозиции и заметки по оценке рисков / Фамилия_ПЗ3_Черновик

Критерии оценки (чек-лист):

- [] Диаграммы выполнены в нотации UML, содержат необходимые элементы и связи
- [] Спецификация интерфейсов включает сигнатуры, форматы и примеры вызовов
- [] Оценка рисков содержит не менее трёх сценариев с планом смягчения
- [] Выбор архитектурного стиля обоснован требованиями к масштабируемости и сопровождаемости

Тема 4. Конструирование программного обеспечения и стандарты кодирования

Цель практики и её задачи: Освоить трансляцию архитектурных решений в код и применение стандартов качества. *Задачи:* реализовать прототип модуля, настроить статический анализ, рассчитать метрики качества, оформить отчёт.

Входные данные:

- Спецификация интерфейсов из ПЗ 3

- Руководство по стилю кодирования для выбранного языка
- Шаблон отчёта по качеству кода

Инструменты: Выбранный язык программирования (Python/Java/C++), линтер (pylint/Checkstyle), P7-Офис

Пошаговое задание:

1. Реализовать прототип одного модуля согласно спецификации интерфейсов
2. Настроить линтер: правила именования, форматирования, документирования
3. Провести статический анализ: выявить нарушения стандартов, «запахи кода»
4. Рассчитать метрики качества: цикломатическая сложность, объём дублирования, покрытие комментариями
5. Разработать план рефакторинга: приоритетные участки, целевые значения метрик, сроки
6. Оформить отчёт о соответствии стандартам, результатах анализа и плане улучшений

Итоговый результат:

.pdf / .docx отчёт о качестве кода с метриками и планом рефакторинга /
 Фамилия_ПЗ4_КачествоКода
 .исходный код прототипа с комментариями и конфигурацией линтера / Фамилия_ПЗ4_Код

Критерии оценки (чек-лист):

- [] Прототип модуля реализован согласно спецификации и проходит базовые проверки
- [] Настройки линтера покрывают не менее 10 правил стиля и качества
- [] Метрики качества рассчитаны корректно, целевые значения обоснованы
- [] План рефакторинга содержит конкретные действия, приоритеты и ожидаемый эффект

Тема 5. Тестирование и обеспечение качества программного обеспечения

Цель практики и её задачи: Освоить методику проектирования тестов и документирования процессов контроля качества. *Задачи:* спроектировать уровни тестирования, разработать Программу и методику испытаний (ПМИ), рассчитать метрики покрытия.

Входные данные:

- Техническое задание и спецификации из ПЗ 2–3
- Примеры ПМИ из методических материалов
- Шаблон ПМИ в текстовом редакторе

Инструменты: P7-Офис / Яндекс.Документы / Свободный текстовый процессор

Пошаговое задание:

1. Определить цели и объекты тестирования: функциональность, надёжность, производительность
2. Спроектировать уровни тестирования: модульное, интеграционное, системное, приёмочное
3. Применить техники тест-дизайна: эквивалентное разбиение, анализ граничных значений
4. Разработать Программу и методику испытаний: сценарии, входные/выходные данные, критерии успеха
5. Рассчитать метрики покрытия кода: по операторам, ветвям, путям
6. Оформить комплект документов: ПМИ, протокол тестового прогона, реестр дефектов, план устранения

Итоговый результат:

.pdf / .docx Программа и методика испытаний с протоколом и реестром дефектов /
 Фамилия_ПЗ5_ПМИ
 .txt черновик тест-сценариев и заметки по расчёту покрытия / Фамилия_ПЗ5_Черновик

Критерии оценки (чек-лист):

- [] ПМИ содержит полные сценарии с входными/выходными данными и критериями успеха
- [] Техники тест-дизайна применены корректно, покрывают основные классы эквивалентности
- [] Метрики покрытия рассчитаны с пояснением методики и интерпретацией результатов
- [] Реестр дефектов включает приоритизацию и план устранения с оценкой трудоёмкости

Тема 6. Управление конфигурациями и системами контроля версий

Цель практики и её задачи: Освоить настройку репозитория, стратегии ветвления и стандартизацию коммитов. *Задачи:* инициализировать репозиторий, реализовать стратегию ветвления, настроить автоматизацию журналов изменений.

Входные данные:

- Исходный код и документация из ПЗ 3–5
- Инструкция по работе с системой контроля версий
- Шаблон сообщения коммита и реестра версий

Инструменты: Система контроля версий (Git), Р7-Офис / Яндекс.Документы

Пошаговое задание:

1. Инициализировать репозиторий: создать структуру папок, настроить `.gitignore` и `.gitattributes`
2. Реализовать стратегию ветвления: создать ветки `feature`, `develop`, `release`, `main`
3. Выполнить тестовое слияние: разрешить конфликт, зафиксировать решение
4. Стандартизировать сообщения коммитов: структура, привязка к задачам, тип изменения
5. Настроить автоматическую генерацию журнала изменений: тегирование, формат `changelog`
6. Оформить отчёт со схемой ветвления, шаблоном коммита и инструкцией по работе с репозиторием

Итоговый результат:

`.pdf` / `.docx` отчёт со схемой ветвления, шаблоном коммита и инструкцией /
Фамилия_ПЗ6_Конфигурации
`.репозиторий` с историей коммитов и файлами конфигурации / Фамилия_ПЗ6_Репозиторий

Критерии оценки (чек-лист):

- [] Структура репозитория логична, `.gitignore` исключает временные и чувствительные файлы
- [] Стратегия ветвления реализована корректно, конфликты разрешены документированно
- [] Шаблон сообщения коммита включает тип, описание, ссылку на задачу
- [] Инструкция по работе с репозиторием понятна новому участнику команды

Тема 7. Сопровождение программного обеспечения и управление релизами

Цель практики и её задачи: Освоить планирование релизов, процедуры отката и пост-релизный анализ. *Задачи:* классифицировать изменения, разработать релизный чек-лист, настроить процедуру отката.

Входные данные:

- Техническое задание, спецификации, результаты тестирования из ПЗ 2–5
- Примеры релизных чек-листов и пост-мортем отчётов
- Шаблон журнала изменений и плана мониторинга

Инструменты: Р7-Офис / Яндекс.Документы / Свободный текстовый процессор

Пошаговое задание:

1. Классифицировать изменения: корректирующие, адаптивные, совершенствующие, превентивные
2. Оценить стоимость сопровождения: трудоёмкость, риски, ресурсы
3. Разработать релизный чек-лист: артефакты, тестирование, документация, коммуникация
4. Настроить процедуру отката: триггеры, шаги, верификация, информирование
5. Спланировать пост-релизный мониторинг: метрики, алерты, периодичность отчётов
6. Оформить отчёт с релизным чек-листом, инструкцией по откату и планом мониторинга

Итоговый результат:

`.pdf` / `.docx` релизный чек-лист, инструкция по откату и план мониторинга / Фамилия_ПЗ7_Релиз
`.txt` черновик классификации изменений и заметки по оценке стоимости / Фамилия_ПЗ7_Черновик

Критерии оценки (чек-лист):

- [] Классификация изменений покрывает все типы, каждый тип проиллюстрирован примером
- [] Релизный чек-лист включает не менее 10 пунктов с критериями проверки
- [] Процедура отката содержит конкретные шаги, триггеры и роли ответственных
- [] План мониторинга привязан к бизнес-метрикам и техническим индикаторам

Тема 8. Управление проектами разработки и экономика программного обеспечения

Цель практики и её задачи: Освоить методы планирования, оценки и экономического обоснования. *Задачи:* построить ИСР, оценить трудоёмкость, разработать диаграмму Ганта, рассчитать экономические показатели.

Входные данные:

- Устав проекта и ТЗ из ПЗ 1–2
- Примеры диаграмм Ганта и расчётов экономической эффективности
- Шаблон реестра рисков и финансового обоснования

Инструменты: P7-Офис / Яндекс.Документы / Табличный процессор

Пошаговое задание:

1. Построить иерархическую структуру работ (ИСР): декомпозиция до исполняемых задач
2. Оценить трудоёмкость задач: экспертный метод, аналогии, буферизация рисков
3. Разработать диаграмму Ганта: сроки, зависимости, критический путь, резервы
4. Идентифицировать риски: вероятность, влияние, план реагирования, ответственные
5. Рассчитать экономические показатели: общая стоимость владения, окупаемость инвестиций
6. Оформить отчёт с ИСР, диаграммой Ганта, реестром рисков и финансовым обоснованием

Итоговый результат:

.pdf / .docx план управления проектом с диаграммой Ганта и расчётами /
Фамилия_ПЗ8_УправлениеПроектом
.таблицы расчётов и черновик реестра рисков / Фамилия_ПЗ8_Расчёты

Критерии оценки (чек-лист):

- [] ИСР декомпозирована до уровня задач, не требующих дальнейшего разбиения
- [] Диаграмма Ганта отражает реальные зависимости и учитывает резерв времени
- [] Реестр рисков содержит количественные оценки и конкретные планы реагирования
- [] Экономические расчёты обоснованы допущениями и привязаны к бизнес-целям

Тема 9. Особенности проектирования цифровых продуктов: от концепции к реализации

Цель практики и её задачи: Освоить трансформацию пользовательских сценариев в проектные спецификации. *Задачи:* проанализировать сценарии, разработать диаграммы потоков и состояний, специфицировать адаптивность.

Входные данные:

- Пользовательские сценарии и бизнес-требования из ПЗ 2
- Примеры диаграмм пользовательских потоков и спецификаций адаптивности
- Шаблон отчёта по проектированию цифрового интерфейса

Инструменты: Draw.io / PlantUML / P7-Офис

Пошаговое задание:

1. Проанализировать пользовательские сценарии: выделить роли, цели, точки принятия решений
2. Разработать диаграмму пользовательских потоков: этапы, ветвления, состояния
3. Построить диаграмму состояний интерфейса: переходы, триггеры, визуальные изменения
4. Специфицировать правила адаптивности: брейкпоинты, плотность пикселей, ориентация

5. Интегрировать требования доступности: контраст, навигация с клавиатуры, альтернативные тексты
6. Оформить отчёт с пакетом диаграмм, спецификацией состояний и матрицей соответствия стандартам доступности

Итоговый результат:

.pdf / .docx отчёт с диаграммами потоков, состояний и спецификацией адаптивности /
Фамилия_ПЗ9_Проектирование
.txt черновик сценариев и заметки по требованиям доступности / Фамилия_ПЗ9_Черновик

Критерии оценки (чек-лист):

- [] Диаграммы потоков и состояний выполнены в единой нотации, покрывают основные сценарии
- [] Спецификация адаптивности включает не менее трёх брейкпоинтов с правилами для каждого
- [] Требования доступности привязаны к конкретным пунктам ГОСТ Р 52872-2019 или аналогов
- [] Матрица соответствия стандартам содержит проверку по каждому требованию

Тема 10. Версионирование и управление артефактами цифрового продукта

Цель практики и её задачи: Освоить семантическое версионирование и синхронизацию проектных артефактов. *Задачи:* организовать структуру репозитория, настроить версионирование компонентов, автоматизировать документирование изменений.

Входные данные:

- Артефакты проектирования из ПЗ 9
- Инструкция по семантическому версионированию
- Шаблон журнала изменений и схемы синхронизации

Инструменты: Система контроля версий (Git), P7-Офис / Яндекс.Документы

Пошаговое задание:

1. Организовать структуру репозитория: папки для макетов, спецификаций, кода, документации
2. Настроить семантическое версионирование: правила инкремента major/minor/patch для компонентов
3. Привязать изменения к задачам в трекере: формат сообщений, метаданные, статусы
4. Автоматизировать генерацию журнала изменений: парсинг коммитов, группировка по версиям
5. Разработать инструкцию по синхронизации проектных и кодовых баз: триггеры, роли, проверки
6. Оформить отчёт со схемой версионирования, шаблоном changelog и инструкцией по синхронизации

Итоговый результат:

.pdf / .docx схема версионирования, шаблон changelog и инструкция по синхронизации /
Фамилия_ПЗ10_Версионирование
.репозиторий с историей версий и конфигурационными файлами / Фамилия_ПЗ10_Репозиторий

Критерии оценки (чек-лист):

- [] Структура репозитория логична, разделена по типам артефактов
- [] Правила семантического версионирования покрывают все типы изменений компонентов
- [] Шаблон changelog включает обязательные разделы: изменения, миграции, депрекация
- [] Инструкция по синхронизации содержит чек-лист проверок перед слиянием

Тема 11. Спецификации передачи дизайна в разработку

Цель практики и её задачи: Освоить подготовку машиночитаемых и визуальных спецификаций

для передачи в разработку. Задачи: документировать компоненты, специфицировать адаптивность, разработать чек-лист полноты передачи.

Входные данные:

- Спецификации компонентов и состояний из ПЗ 9
- Примеры handoff-документации и чек-листов передачи
- Шаблон спецификации компонента в текстовом редакторе

Инструменты: Р7-Офис / Яндекс.Документы / Свободный текстовый процессор

Пошаговое задание:

1. Документировать компоненты: назначение, пропсы, состояния, примеры использования
2. Специфицировать правила адаптивности: брейкпоинты, плотность, ориентация, поведение
3. Подготовить машиночитаемые спецификации: JSON-токены, контракты API, метаданные
4. Разработать чек-лист полноты передачи: визуал, состояния, доступность, анимации, экспорт
5. Организовать процесс обратной связи: фиксация расхождений, итеративное уточнение, протоколирование
6. Оформить отчёт с комплектом спецификаций, протоколом согласования и реестром расхождений

Итоговый результат:

.pdf / .docx комплект спецификаций компонентов и чек-лист передачи /
Фамилия_ПЗ11_Спецификации
.файлы JSON-токенов и черновик чек-листа / Фамилия_ПЗ11_Черновик

Критерии оценки (чек-лист):

- [] Спецификации компонентов содержат все обязательные разделы и примеры использования
- [] Правила адаптивности покрывают не менее трёх типов устройств с конкретными значениями
- [] Чек-лист передачи включает не менее 10 пунктов с критериями проверки
- [] Протокол согласования фиксирует расхождения и план их устранения

Тема 12. Тестирование цифровых продуктов: валидация пользовательского опыта

Цель практики и её задачи: Освоить методику тестирования юзабилити и доступности цифровых интерфейсов. Задачи: разработать сценарии тестирования, составить ПМИ, провести проверки, оформить отчёт.

Входные данные:

- Спецификации компонентов и требования доступности из ПЗ 9–11
- Примеры ПМИ для цифровых интерфейсов
- Шаблон протокола тестирования и аудита доступности

Инструменты: Р7-Офис / Яндекс.Документы / Инструменты автоматизированного тестирования

Пошаговое задание:

1. Разработать сценарии юзабилити-тестирования: задачи, метрики успеха, критерии завершения
2. Составить сценарии проверки доступности: навигация с клавиатуры, контраст, альтернативные тексты
3. Разработать Программу и методику испытаний цифрового интерфейса: цели, объекты, сценарии, критерии
4. Провести автоматизированные и ручные проверки: зафиксировать результаты, выявить дефекты
5. Рассчитать метрики выполнения задач: время, уровень ошибок, удовлетворённость
6. Оформить отчёт с ПМИ, протоколами тестирования, аудитом доступности и рекомендациями по устранению

Итоговый результат:

.pdf / .docx ПМИ цифрового интерфейса с протоколами и рекомендациями /
Фамилия_ПЗ12_ПМИ_Интерфейс

Критерии оценки (чек-лист):

- [] Сценарии тестирования покрывают ключевые пользовательские задачи и требования доступности
- [] ПМИ содержит полные сценарии с входными/выходными данными и критериями успеха
- [] Аудит доступности привязан к конкретным пунктам ГОСТ Р 52872-2019 или аналогов
- [] Рекомендации по устранению дефектов приоритизированы и оценены по трудоёмкости

Тема 13. Релиз-менеджмент цифровых продуктов: подготовка, деплой, мониторинг

Цель практики и её задачи: Освоить подготовку цифрового релиза, настройку конвейера развёртывания и пост-релизный мониторинг. *Задачи:* разработать чек-лист готовности, настроить окружения, спланировать мониторинг.

Входные данные:

- Спецификации, результаты тестирования и ПМИ из ПЗ 11–12
- Примеры релизных чек-листов и конфигураций окружений
- Шаблон плана мониторинга и журнала выпуска

Инструменты: Р7-Офис / Яндекс.Документы / Инструменты настройки окружений

Пошаговое задание:

1. Подготовить чек-лист готовности релиза: визуальная целостность, соответствие спецификациям, тестирование
2. Настроить конфигурации окружений: dev/stage/prod, переменные среды, секреты
3. Разработать процедуру отката: триггеры, шаги, верификация, информирование
4. Спланировать пост-релизный мониторинг: метрики производительности, пользовательского поведения, алерты
5. Разработать журнал выпуска: версия, изменения, миграции, ответственные, дата
6. Оформить отчёт с релизным чек-листом, конфигурацией окружений, планом мониторинга и журналом выпуска

Итоговый результат:

.pdf / .docx релизный чек-лист, конфигурация окружений и план мониторинга / Фамилия_ПЗ13_Релиз

.файлы конфигураций и черновик журнала выпуска / Фамилия_ПЗ13_Черновик

Критерии оценки (чек-лист):

- [] Чек-лист готовности включает не менее 10 пунктов с критериями проверки
- [] Конфигурации окружений разделены, секреты защищены, переменные документированы
- [] Процедура отката содержит конкретные шаги, триггеры и роли ответственных
- [] План мониторинга привязан к бизнес-метрикам и техническим индикаторам

Тема 14. Интеграция ИИ-компонентов в цифровые продукты: инженерные аспекты

Цель практики и её задачи: Освоить спецификацию интерфейсов и валидацию ИИ-компонентов. *Задачи:* разработать контракт взаимодействия, методику валидации, документировать ограничения и этические аспекты.

Входные данные:

- Описание ИИ-компонента (задача, входные/выходные данные, метрики качества)
- Примеры контрактов API и методик валидации ИИ-моделей
- Шаблон протокола атрибуции и плана мониторинга дрейфа

Инструменты: Р7-Офис / Яндекс.Документы / Свободный текстовый процессор

Пошаговое задание:

1. Специфицировать интерфейс взаимодействия: форматы запросов/ответов, коды ошибок, таймауты
2. Разработать методику валидации качества предсказаний: метрики, пороговые значения, сценарии тестов
3. Документировать ограничения компонента: допустимые входные данные, области применения, риски
4. Описать этические аспекты: прозрачность, предвзятость, атрибуция генерированного контента
5. Разработать план мониторинга дрейфа: метрики, частота проверок, триггеры переобучения
6. Оформить отчёт с контрактом взаимодействия, методикой валидации, протоколом атрибуции и планом мониторинга

Итоговый результат:

.pdf / .docx контракт API ИИ-компонента, методика валидации и план мониторинга /
 Фамилия_ПЗ14_ИИ_Интеграция

.черновик спецификации и заметки по этическим аспектам / Фамилия_ПЗ14_Черновик

Критерии оценки (чек-лист):

- [] Контракт взаимодействия содержит полные сигнатуры, форматы и примеры вызовов
- [] Методика валидации включает метрики, пороговые значения и сценарии тестов
- [] Ограничения и этические аспекты документированы с привязкой к нормативным требованиям
- [] План мониторинга дрейфа содержит конкретные метрики, частоту и триггеры действий

Тема 15. Обеспечение качества и безопасность цифровых продуктов

Цель практики и её задачи: Освоить аудит безопасности и статический анализ спецификаций.
 Задачи: провести аудит зависимостей, разработать ПМИ по безопасности, выявить уязвимости, оформить план устранения.

Входные данные:

- Спецификации компонентов и требования безопасности из ПЗ 11–14
- Примеры ПМИ по безопасности и отчётов аудита
- Шаблон реестра уязвимостей и плана устранения

Инструменты: P7-Офис / Яндекс.Документы / Инструменты статического анализа безопасности

Пошаговое задание:

1. Провести статический анализ спецификаций: выявление уязвимостей в логике, данных, интерфейсах
2. Выполнить аудит зависимостей: проверка на известные уязвимости, лицензии, актуальность
3. Разработать Программу и методику испытаний по безопасности: сценарии, входные данные, критерии
4. Провести тестирование на основные классы уязвимостей: инъекции, аутентификация, авторизация
5. Оформить реестр уязвимостей: описание, критичность, влияние, план устранения, ответственные
6. Оформить отчёт с ПМИ по безопасности, результатами аудита, реестром уязвимостей и планом устранения

Итоговый результат:

.pdf / .docx ПМИ по безопасности, отчёт аудита и план устранения уязвимостей /
 Фамилия_ПЗ15_Безопасность

.реестр уязвимостей и черновик ПМИ / Фамилия_ПЗ15_Черновик

Критерии оценки (чек-лист):

- [] Статический анализ покрывает не менее трёх классов уязвимостей с примерами
- [] Аудит зависимостей включает проверку на уязвимости, лицензии и актуальность

- [] ПМИ по безопасности содержит полные сценарии с критериями успеха
- [] План устранения уязвимостей приоритизирован и оценен по трудоёмкости и рискам

Тема 16. Интеграция инженерных артефактов

Цель практики и её задачи: Освоить структурирование и нормативное оформление инженерных материалов для ВКР. Задачи: подготовить оглавление приложений, проверить оформление по нормоконтролю, оформить чек-лист самопроверки.

Входные данные:

- Накопленные инженерные артефакты из ПЗ 1–15 (ТЗ, диаграммы, ПМИ, спецификации)
- Требования к оформлению ВКР АНО ВО ВИБ (шаблон, ГОСТ, нормоконтроль)
- Шаблон оглавления приложений и чек-листа нормоконтроля

Инструменты: Р7-Офис / Яндекс.Документы / Свободный текстовый процессор

Пошаговое задание:

1. Структурировать накопленные артефакты: группировка по разделам ВКР, нумерация, ссылки
2. Подготовить оглавление приложений: названия, номера, страницы, типы артефактов
3. Проверить оформление на соответствие требованиям нормоконтроля: шрифты, отступы, библиография
4. Оформить графическую часть: диаграммы, таблицы, скриншоты — с подписями и ссылками в тексте
5. Составить чек-лист самопроверки: уникальность, цитирование, атрибуция ИИ-контента, метаданные
6. Оформить отчёт с оглавлением приложений, чек-листом нормоконтроля и планом доработок

Итоговый результат:

.pdf / .docx оглавление приложений ВКР, чек-лист нормоконтроля и план доработок /
 Фамилия_ПЗ16_ВКР_Подготовка
 .черновик оглавления и заметки по оформлению / Фамилия_ПЗ16_Черновик

Критерии оценки (чек-лист):

- [] Оглавление приложений структурировано, содержит ссылки на все артефакты ПЗ 1–15
- [] Оформление соответствует требованиям ГОСТ и шаблона ВКР АНО ВО ВИБ
- [] Графическая часть содержит подписи, нумерацию и ссылки в тексте пояснительной записки
- [] Чек-лист самопроверки покрывает уникальность, цитирование, атрибуцию и метаданные

5.4. Перечень вопросов промежуточной аттестации по дисциплине

Тема курсовой работы «Реализация этапов жизненного цикла цифрового дизайн-продукта (на примере проекта выпускной квалификационной работы)»

Рекомендуемая структура курсовой работы

Введение — актуальность применения ЖЦ ПО в цифровом дизайне, цель, задачи, объект/предмет исследования

Теоретический раздел — модели ЖЦ, их применимость к дизайн-разработке

Аналитический раздел — выбор модели ЖЦ под конкретный продукт ВКР

Практический раздел — реализация этапов ЖЦ для выбранного продукта:

- Сбор и формализация требований (ТЗ по ГОСТ 34.602-89)
- Проектирование и версионирование артефактов
- Контроль качества
- Подготовка к релизу и передача в разработку

Экономический раздел — оценка трудоёмкости, рисков, окупаемости дизайн-решений

Заключение — выводы, рекомендации по масштабированию процесса

Вопросы к экзамену:

1. Какие нормативные стандарты регламентируют профессиональную практику разработчика в РФ?
2. Каковы методы формализации функциональных и нефункциональных требований к программному продукту?
3. Как обеспечивается сквозная трассируемость требований на протяжении жизненного цикла ПО?
4. Какие архитектурные принципы обеспечивают слабую связанность и высокую сопровождаемость системы?
5. Какова роль диаграмм в документировании проектных решений?
6. Как стандарты кодирования влияют на читаемость и поддерживаемость исходного кода?
7. Какие метрики качества кода используются для оценки технического долга?
8. В чём методологическое различие между верификацией и валидацией программного продукта?
9. Как техники тест-дизайна оптимизируют покрытие кода при ограниченных ресурсах?
10. Каковы критерии выбора стратегии ветвления в распределённой системе контроля версий?
11. Как семантическое версионирование обеспечивает обратную совместимость компонентов?
12. Какие процедуры классификации изменений применяются на этапе сопровождения ПО?
13. Каковы этапы подготовки стабильной версии продукта к выпуску в продакшен?
14. Как методы оценки трудоёмкости влияют на точность календарного планирования проекта?
15. Какие метрики проектной деятельности используются для мониторинга эффективности команды?
16. Каковы принципы трансформации пользовательских сценариев в требования к информационной архитектуре?
17. Как требования цифровой доступности интегрируются в процесс проектирования интерфейса?
18. Как организуется структура репозитория для синхронизации визуальных и кодовых артефактов?
19. Каковы правила семантического версионирования применительно к изменениям компонентов интерфейса?
20. Какие форматы спецификаций обеспечивают машиночитаемость проектных данных?
21. Каковы критерии полноты и однозначности документации при передаче артефактов в разработку?
22. Как проектируются тестовые сценарии для валидации юзабилити и доступности интерфейса?
23. Какие метрики пользовательского опыта используются для оценки эффективности взаимодействия?
24. Каковы обязательные компоненты чек-листа готовности цифрового релиза к развёртыванию?
25. Как стратегии постепенного внедрения минимизируют риски при обновлении продукта?
26. Как специфицируются интерфейсы взаимодействия с недетерминированными ИИ-сервисами?
27. Какие методы валидации качества предсказаний применяются для ИИ-компонентов в продакшене?
28. Как проводится аудит зависимостей на наличие известных уязвимостей в цифровом продукте?
29. Какие подходы к статическому анализу спецификаций обеспечивают соответствие требованиям информационной безопасности?
30. Каковы критерии отбора и структурирования инженерных материалов для включения в пояснительную записку ВКР?
31. Какие требования нормоконтроля предъявляются к оформлению графической части выпускной работы?
32. Как нормативные стандарты влияют на выбор методологии управления проектом?
33. В чём заключается роль автора проекта на этапе авторского надзора за технической реализацией?

34. Как обеспечивается воспроизводимость программных сборок в командной разработке?
35. Каковы процедуры анализа и фиксации уроков после пост-релизных инцидентов?
36. Как рассчитывается общая стоимость владения цифровым продуктом на этапе экономического обоснования?
37. Какие этические аспекты учитываются при документировании ИИ-генерируемого контента в академической работе?
38. Как методы системного анализа применяются для декомпозиции проектных задач цифрового продукта?
39. Каковы критерии оценки готовности инженерных артефактов к защите выпускной квалификационной работы?

Раздел 6. Оценочные средства промежуточной аттестации (с ключами)

1. Выберите два правильных ответа. Какие два артефакта являются обязательными в методологии Scrum?

- а) Диаграмма Ганта
- б) Product Backlog
- в) Sprint Backlog
- г) Техническое задание по ГОСТ

Правильный ответ: б) в)

2. Установите соответствие между уровнями тестирования программного обеспечения и их целью:

А) Модульное	1) Проверка взаимодействия между отдельными компонентами системы
Б) Интеграционное	2) Тестирование наименьших проверяемых частей кода (функций, классов)
В) Системное	3) Оценка соответствия всей системы заявленным требованиям
Г) Приёмочное	4) Подтверждение готовности продукта к эксплуатации заказчиком

Правильный ответ: А-2, Б-1, В-3, Г-4

3. Разместите по порядку основные фазы жизненного цикла проекта разработки ПО (от начальной к завершающей):

- 1. Исполнение проекта и разработка кода
- 2. Инициация проекта и формирование устава
- 3. Закрытие проекта и передача продукта
- 4. Планирование требований, архитектуры и сроков
- 5. Мониторинг, контроль качества и управление изменениями

Правильный ответ: 2, 4, 1, 5, 3

4. Запишите термин (с маленькой буквы, в именительном падеже), обозначающий процесс автоматической сборки, тестирования и подготовки кода к развёртыванию при каждом изменении в репозитории. _____ — практика, обеспечивающая раннее выявление интеграционных ошибок.

Правильный ответ: непрерывная интеграция

5. Установите соответствие между стратегиями развёртывания ПО и их описанием:

А) Blue-Green	1) Постепенная замена старой версии новой с параллельной работой двух окружений
Б) Canary	2) Выпуск обновления на небольшую долю пользователей для проверки стабильности
В) Rolling Update	3) Последовательное обновление экземпляров сервиса без простоя
Г) Recreate	4) Полная остановка старой версии перед запуском новой

Правильный ответ: А-1, Б-2, В-3, Г-4

6. Выберите три правильных ответа. Какие из перечисленных характеристик относятся к нефункциональным требованиям к программному продукту?

- а) Система должна обрабатывать до 1000 запросов в секунду
- б) Пользователь может добавлять товары в корзину
- в) Время отклика интерфейса не должно превышать 2 секунд
- г) Система должна соответствовать требованиям цифровой доступности ГОСТ Р 52872-2019
- д) Администратор имеет возможность блокировать учетные записи

Правильный ответ: а) в) г)

7. Разместите по порядку типичные действия при работе по стратегии Git Flow для выпуска новой функции:

1. Слияние ветки **release/*** в **main** и **develop**
2. Создание ветки **feature/*** от **develop**
3. Фиксация версии тегом **vX.X.X** в **main**
4. Проведение ревью кода и слияние в **develop**
5. Создание ветки **release/*** от **develop** для финального тестирования

Правильный ответ: 2, 4, 5, 1, 3

8. Выберите два правильных ответа. Какие две практики относятся к концепции DevSecOps?

- а) Проверка зависимостей на известные уязвимости (SAST/DAST) на этапе CI
- б) Ручное тестирование безопасности только перед релизом
- в) Автоматический запуск анализаторов кода при каждом коммите
- г) Хранение секретов и ключей доступа в открытом виде в репозитории

Правильный ответ: а) в)

9. Запишите термин (с маленькой буквы, в именительном падеже), обозначающий архитектурный стиль, при котором система состоит из набора слабосвязанных, независимо развёртываемых сервисов. _____ — подход, обеспечивающий высокую масштабируемость и технологическую гибкость.

Правильный ответ: микросервисная архитектура

10. Установите соответствие между метрикой Agile-проекта и её описанием:

А) Velocity	1) График, показывающий оставшийся объём работы по спринту
Б) Burndown chart	2) Количество условных единиц (story points), выполненных командой за итерацию
В) Cycle time	3) Время от начала работы над задачей до её полного завершения
Г) Throughput	4) Количество задач, завершённых за единицу времени

Правильный ответ: А-2, Б-1, В-3, Г-4

11. Разместите по порядку этапы работы с требованиями к ПО в соответствии с классическим процессом:

1. Валидация и верификация требований
2. Сбор требований
3. Управление изменениями требований
4. Анализ и формализация требований в спецификации
5. Приоритизация и согласование с заказчиком

Правильный ответ: 2, 4, 5, 1, 3

12. Выберите два правильных ответа. Какие две характеристики входят в модель качества ПО ISO/IEC 25010?

- а) Функциональная пригодность
 - б) Эстетическая привлекательность
 - в) Сопровождаемость
 - г) Стоимость разработки
- Правильный ответ: а) в)**

13. Запишите термин (с большой буквы, в именительном падеже), обозначающий технологию упаковки приложения со всеми зависимостями в изолированный переносимый образ. _____ — стандарт де-факто для обеспечения воспроизводимости сред развёртывания.

Правильный ответ: Контейнеризация

14. Установите соответствие между инструментом анализа кода и его назначением:

А) Линтер (например, pylint)	1) Поиск уязвимостей и опасных паттернов в коде
Б) Статический анализатор безопасности (например, bandit)	2) Проверка соответствия кода стандартам оформления и выявление синтаксических ошибок
В) Инструмент покрытия кода (например, coverage)	3) Измерение доли строк/ветвей кода, выполненных при тестировании

Правильный ответ: А-2, Б-1, В-3

15. Укажите один правильный ответ. Какой инструмент управления проектами визуализирует задачи во времени, показывает зависимости и критический путь?

- а) Канбан-доска
- б) Диаграмма Ганта
- в) Burndown-чарт
- г) Матрица Эйзенхауэра

Правильный ответ: б)

16. Разместите по порядку этапы эффективного процесса code review:

1. Автор вносит правки и обновляет запрос на слияние
2. Разработчик создаёт Pull/Merge Request с описанием изменений
3. Ревьюер проверяет код по чек-листу, оставляет комментарии
4. Автоматические проверки CI/CD проходят успешно
5. Ревьюер аппрувит изменения, код сливается в основную ветку

Правильный ответ: 2, 4, 3, 1, 5

17. Выберите два правильных ответа. Какие два требования необходимо учитывать при передаче дизайн-спецификаций в разработку с точки зрения программной инженерии?

- а) Машиночитаемый формат токенов (JSON/YAML)
- б) Только визуальные скриншоты без описания состояний
- в) Документирование правил адаптивности и доступности
- г) Отказ от версионирования макетов для ускорения процесса

Правильный ответ: а) в)

18. Укажите один правильный ответ. Какая диаграмма UML используется для отображения физических узлов, серверов и развёртывания артефактов в системе?

- а) Диаграмма классов
- б) Диаграмма последовательностей
- в) Диаграмма развёртывания
- г) Диаграмма состояний

Правильный ответ: в)

Раздел 7. Перечень учебной литературы, необходимой для освоения дисциплины

7.1. Обязательная литература

1. Воронина, В. В. Отладка программных решений : учебно-методическое пособие для студентов направлений подготовки бакалавриата 09.03.03 «Прикладная информатика», 09.03.04 «Программная инженерия» / В. В. Воронина, Е. Н. Эгов. — Ульяновск : Ульяновский государственный технический университет, 2024. — 103 с. — ISBN 978-5-9795-2383-5. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/149282.html>
2. Влацкая, И. В. Анализ программных реализаций : практикум / И. В. Влацкая. — Оренбург : Оренбургский государственный университет, ЭБС АСВ, 2024. — 97 с. — ISBN 978-5-7410-3309-8. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/153023.html>
3. Мейер, Б. Объектно-ориентированное программирование и программная инженерия : учебное пособие / Б. Мейер. — 4-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2024. — 284 с. — ISBN 978-5-4497-2464-9. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/133956.html>
4. Хабаров, А. Н. Разработка программных приложений : учебник / А. Н. Хабаров, А. Н. Ермакова. — Ставрополь : АГРУС, 2025. — 208 с. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/156614.html>

7.2. Дополнительная литература

5. Фаулер М., Рефакторинг: улучшение существующего кода, 2019
6. Буч Г., Объектно-ориентированный анализ и проектирование с примерами приложений, 2017
7. Брукс Ф., Мифический человеко-месяц, 2018
8. Басс Л., Клементс П., Кацман Р., Архитектура программного обеспечения: практика и принципы, 2020
9. Соммервилл И., Инженерия программного обеспечения, 2021

7.3. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

10. https://bronevichok.ru/static/swebok_2004_ru.html

Раздел 8. Материально-техническая база и информационные технологии

Материально-техническое обеспечение дисциплины «**Программная инженерия**» включает в себя учебные аудитории для проведения лекционных, лабораторных занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, самостоятельной работы обучающихся.

Учебные аудитории укомплектованы специализированной мебелью и техническими средствами обучения. Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети Интернет.

Дисциплина может реализовываться с применением дистанционных технологий обучения. Специфика реализации дисциплины с применением дистанционных технологий обучения устанавливается дополнением к рабочей программе. В части не противоречащей специфике, изложенной в дополнении к программе, применяется настоящая рабочая программа.

Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине с применением дистанционных образовательных технологий включает в себя: Компьютерная техника, расположенная в учебном корпусе Института (ул. Качинцев, 63, кабинет Центра дистанционного обучения):

- 1) Intel i3 3.4Ghz\ОЗУ 4Gb\500GB\RadeonHD5450
- 2) Intel PENTIUM 2.9GHz\ОЗУ 4GB\500GB
- 3) личные электронные устройства (компьютеры, ноутбуки, планшеты и иное), а также средства связи преподавателей и студентов.

Информационные технологии, необходимые для осуществления образовательного процесса по дисциплине с применением дистанционных образовательных технологий включают в себя:

- система дистанционного обучения (СДО) (Learning Management System) (LMS) Moodle (Modular Object-Oriented Dynamic Learning Environment);
- электронная почта;
- система компьютерного тестирования;
- Цифровой образовательный ресурс IPR SMART;
- система интернет-связи skype;
- телефонная связь;
- ПО для проведения конференций.

Обучение обучающихся инвалидов и обучающихся с ограниченными возможностями здоровья осуществляется посредством применения специальных технических средств в зависимости от вида нозологии.

При проведении учебных занятий по дисциплине используются мультимедийные комплексы, электронные учебники и учебные пособия, адаптированные к ограничениям здоровья обучающихся.

Лекционные аудитории оборудованы мультимедийными кафедрами, подключенными к звуковым колонкам, позволяющими усилить звук для категории слабослышащих обучающихся, а также проекционными экранами, которые увеличивают изображение в несколько раз и позволяют воспринимать учебную информацию обучающимся с нарушениями зрения.

При обучении лиц с нарушениями слуха используется усилитель слуха для слабослышащих людей Cyber Ear модель NAP-40, помогающий обучаемым лучше воспринимать учебную информацию.

Обучающиеся с ограниченными возможностями здоровья, обеспечены печатными и электронными образовательными ресурсами (программы, учебники, учебные пособия, материалы для самостоятельной работы и т.д.) в формах, адаптированных к ограничениям их здоровья и восприятия информации:

для лиц с нарушениями зрения:

- в форме электронного документа;
- в форме аудиофайла;

для лиц с нарушениями слуха:

- в печатной форме;
- в форме электронного документа;

для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме;
- в форме электронного документа;
- в форме аудиофайла.

Программное обеспечение, используемое на занятиях:

- Операционная система Windows,
- Архиватор 7-zip,
- Система тестирования,
- Microsoft Office 2007,
- Антивирус Касперский 6,
- Консультант+,
- Виртуальная машина VirtualBox,
- Виртуальная машина VirtualPC,
- Internet Explorer.
-

Раздел 9. Методические указания для обучающихся по освоению дисциплины

Для успешного усвоения материала дисциплины требуются значительное время, концентрация внимания и усилия: посещение лекционных занятий и конспектирование преподаваемого материала, работа с ним дома, самостоятельная проработка материала рекомендуемых учебников и учебных пособий при самостоятельной подготовке. Особое внимание следует обратить на выполнение практических работ, практических задач по СРО, тестовых вопросов.

При самостоятельной работе с учебниками и учебными пособиями полезно иметь под рукой справочную литературу (энциклопедии) или доступ к сети Интернет, так как могут встречаться новые термины, понятия, которые раньше обучающиеся не знали.

Цель практических занятий по дисциплине - закрепление знаний по определенной теме, приобретенных в результате прослушивания лекций, получения консультаций и самостоятельного изучения различных источников литературы. При выполнении данных работ обучающиеся должны будут глубоко изучить состав и принцип работы современных информационных систем. Получить практические навыки работы с современными ИС.

Перед практическим занятием обучающийся должен детально изучить теоретические материалы вопросов практики в учебниках, конспектах лекций, периодических журналах и прочее. Если при выполнении практического задания у обучающегося остаются неясности, то ему необходимо оперативно обратиться к преподавателю за уточнением.

После выполнения практического задания обучающиеся должны выполнить самостоятельную работу. Самостоятельная работа включает в себя индивидуальное задание по пройденной теме. Таким образом, каждый обучающийся выполняет только свой вариант задания.

При дистанционном выполнении практических работ обучающийся может самостоятельно приобрести операционные системы Windows XP, Windows 7, Windows 8, Windows 10 и пакет Microsoft Office или Open Office. Ответственность за установку и настройку программного обеспечения в данном случае ложится на обучающегося. Следует воспользоваться методическими указаниями по установке данных программных систем.

Результаты выполненных заданий оцениваются с учетом теоретических знаний по соответствующим разделам дисциплины, техники выполнения работы, объективности и обоснованности принимаемых решений в процессе работы с данными, качества оформления. Переход к выполнению следующего практического задания допускается только после отчета выполненной работы.

Учебно-методическое издание

Рабочая программа учебной дисциплины

Программная инженерия

(Наименование дисциплины в соответствии с учебным планом)

Сафонова Елена Владимировна

(Фамилия, Имя, Отчество составителя)
