МИНИСТЕРСТВО ОБРАЗОВАНИЯ СВЕРДЛОВСКОЙ ОБЛАСТИ
ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ СВЕРДЛОВСКОЙ ОБЛАСТИ
«ЕКАТЕРИНБУРГСКИЙ МОНТАЖНЫЙ КОЛЛЕДЖ»

РАБОЧАЯ ПРОГРАММА ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

ПМ.01 «Проектирование и разработка информационных ресурсов»
для специальности
09.02.09 «Веб-разработка»

2026 г.

Чистяков Валерий Николаевич
я подтверждаю этот документ своей удостоверяющей подписью
место подписания
2026.03.30 12:06:35+05'00'

Рабочая программа ПМ.01 «Проектирование и разработка информационных ресурсов» разработана на основе Федерального государственного образовательного стандарта среднего профессионального образования (далее – ФГОС СПО) по специальности 09.02.09 «Веб-разработка», примерной рабочей программы профессионального модуля, учебного плана специальности.

Организация-разработчик: ГАПОУ СО «Екатеринбургский монтажный колледж»

Разработчики:
Софьина Н.А. - преподаватель ГАПОУ СО «Екатеринбургский монтажный колледж», ВКК;
Грибова А.В - преподаватель ГАПОУ СО «Екатеринбургский монтажный колледж», ВКК;
Раковская Ю.В. - преподаватель ГАПОУ СО «Екатеринбургский монтажный колледж», 1КК.

Рассмотрена методическим объединением Информационных технологий

Руководитель МО  Софьина Н.А.

УТВЕРЖДАЮ

Заместитель директора по учебной работе

 Л. С. Хоринова

«13» марта 2026 г.

СОДЕРЖАНИЕ

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ 4
1.1. Область применения программы 4
1.2. Цель и планируемые результаты освоения профессионального модуля 4
1.3. Количество часов, отводимое на освоение профессионального модуля: 7
2. СТРУКТУРА И СОДЕРЖАНИЕ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ 8
2.1. Структура профессионального модуля 8
2.2. Тематический план и содержание профессионального модуля 9
3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ 15
3.1. Требования к минимальному материально-техническому обеспечению 15
3.2. Информационное обеспечение обучения 15
3.3. Общие требования к организации образовательного процесса 16
3.4. Кадровое обеспечение образовательного процесса 16
4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ 17

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

ПМ.01 Проектирование и разработка информационных ресурсов

1.1. Область применения программы

Рабочая программа профессионального модуля является частью программы подготовки специалистов среднего звена в соответствии с ФГОС СПО по специальности 09.02.09 «Веб-разработка».

1.2. Цель и планируемые результаты освоения профессионального модуля

В результате изучения профессионального модуля обучающийся должен освоить вид деятельности «Проектирование и разработка информационных ресурсов» и соответствующие ему общие компетенции и профессиональные компетенции:

1.1.1. Перечень общих компетенций и личностных результатов реализации программы воспитания

Код	Наименование общих компетенций
ОК 01.	Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам
ОК 02.	Использовать современные средства поиска, анализа и интерпретации информации, и информационные технологии для выполнения задач профессиональной деятельности
ОК 03.	Планировать и реализовывать собственное профессиональное и личностное развитие, предпринимательскую деятельность в профессиональной сфере, использовать знания по финансовой грамотности в различных жизненных ситуациях
ОК 04.	Эффективно взаимодействовать и работать в коллективе и команде
ОК 05.	Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста
ОК 06.	Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей, в том числе с учетом гармонизации межнациональных и межрелигиозных отношений, применять стандарты антикоррупционного поведения
ОК 07.	Содействовать сохранению окружающей среды, ресурсосбережению, применять знания об изменении климата, принципы бережливого производства, эффективно действовать в чрезвычайных ситуациях
ОК 08.	Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности
ОК 09.	Пользоваться профессиональной документацией на государственном и иностранном языках.

1.1.2. Перечень профессиональных компетенций

Код	Наименование видов деятельности и профессиональных компетенций

ПК 1.1.	Проектировать информационные ресурсы.
ПК. 1.2.	Разрабатывать интерфейсы пользователя.
ПК 1.3.	Интегрировать программный код в соответствующую инфраструктуру.
ПК 1.4.	Использовать систему контроля версий в процессе коллективной (параллельной) разработки.
ПК 1.5.	Выполнять процедуры тестирования программного кода.

1.1.3. В результате освоения профессионального модуля обучающийся должен:

Владеть навыками	– Проектирования компонентов информационных систем и ресурсов; – разработки прототипов пользовательских интерфейсов; – организации запросов с использованием нейронных сетей, с целью получения исходного кода для интеграции в проект; – интеграции программного кода в соответствующий участках проекта; – оптимизации заимствованного кода; – работы с системой контроля версий, в том числе при коллективной разработке; – разработки тестовых сценариев программного средства; – тестирование информационного ресурса в соответствии с планом тестирования; документирования результатов тестирования; – разработки тестовых сценариев программного средства; – применения техник тест-дизайна (EP, BVA, Pairwise, Decision Table); – создания тестовой документации (Test Plan, тест-кейсы, чек-листы, матрицы прослеживаемости); – автоматизации модульного, интеграционного и UI тестирования; – тестирования информационного ресурса в соответствии с планом тестирования; – документирования результатов тестирования с отчетами покрытия кода.

Уметь	– применять методы системного анализа; – интерпретировать бизнес-требования заказчика для разработки концептуальной модели информационного ресурса; – разрабатывать концептуальную модель информационного ресурса средствами графических нотаций; – разрабатывать прототипы пользовательских интерфейсов с использованием UI/UX подхода; – выполнять поисковые запросы с использованием нейронных сетей (искусственный интеллект); – осуществлять адаптацию заимствованного кода в соответствующих участках проекта; – встраивать в существующий проект готовый код. – создавать, клонирования, развития репозиториев хранения кода; – создавать ветки репозитория и управления изменениями кода; – решать конфликты версий кода. – выбирать и комбинировать техники тестирования информационных ресурсов; тестировать информационный ресурс с использованием тест-планов; – применять инструменты подготовки тестовых данных; – работать с инструментами подготовки тестовых данных; – создавать отчет по результатам тестирования; – выбирать и комбинировать техники тестирования информационных ресурсов; – тестировать информационный ресурс с использованием тест-планов; – применять инструменты подготовки тестовых данных; – работать с инструментами автоматизации тестирования (модульное, API, UI)-
Знать	– основы теории системного анализа и построения концептуальных моделей информационных ресурсов средствами графических нотаций; – понятия, классификацию информационных систем и ресурсов; – этапы, принципы и особенности проектирования информационных систем и ресурсов; – архитектуру информационных систем и ресурсов; – модели процесса разработки информационных систем и ресурсов; – принципы проектирования пользовательских интерфейсов; – элементы управления пользовательского интерфейса; – базовых принципов «общения» с искусственным интеллектом; – теории анализа веб-приложений и веб-ресурсов; – принципы и алгоритмы аудита веб-приложений и веб-ресурсов; – принципы устройства систем хранения версий кода; – современные методики тестирования информационных ресурсов; – теорию тестирования ПО (уровни, виды, методы); – техники тест-дизайна ; – принципы создания тестовой документации; – архитектуру API и принципы REST тестирования; – принципы модульного тестирования и TDD; – современные методики тестирования информационных ресурсов;

	– принципы автоматизации тестирования (функциональная декомпозиция, Data Driven, BDD).

1.3. Количество часов, отводимое на освоение профессионального модуля:

Вид учебной работы	**Объем часов**
Всего объем образовательной нагрузки	**402**
в том числе:	
Объем образовательной нагрузки	
теоретическое обучение	56
практические занятия	98
учебная практика	72
производственная практика	144
Самостоятельная работа	32
Промежуточная аттестация проводится в форме дифференцированного зачета	

2. СТРУКТУРА И СОДЕРЖАНИЕ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

2.1. Структура профессионального модуля

Коды профессиональных общих компетенций	Наименования разделов профессионального модуля**	Суммарныйобъем нагрузки, час.	Занятия во взаимодействии с преподавателем, час					Самостоятельная работа
			Обучение по МДК			Практики		
			Всего	Лабораторных и практических занятий	Курсовых работ (проектов) *	Учебная	Производственная(если предусмотрена рассредоточенная практика)	
1	2	3	4	5	6	7	8	9
ПК 1.1 ОК 1-9	МДК 01.01 Проектирование информационных ресурсов	58	48	30				10
ПК 1.2. ПК 1.4. ОК 1-9	МДК 01.02 Разработка интерфейсов пользователя	68	56	38				12
ПК 1.3. ПК 1.5. ОК 1-9	МДК 01.03 Тестирование информационных ресурсов	60	50	30				10
ПК 1.1 – 1.5, ОК 1-9	УП.01.01 Проектирование и разработка информационных ресурсов	72				72		
ПК 1.1 – 1.5, ОК 1-9	Производственнаяпрактика	144					144	
	Всего:	**402**	**370**	**98**		**2**	**144**	**32**

2.2. Тематический план и содержание профессионального модуля

Наименование разделов профессионального модуля (ПМ), междисциплинарных курсов (МДК) и тем	Содержание учебного материала, лабораторные работы и практические занятия, самостоятельная работа обучающихся, курсовая работ (проект)	Объем часов	Формируемые компетенции
1	**2**	**3**	**4**
РАЗДЕЛ 1.			
МДК 01.01 Проектирование информационных ресурсов		***58***	
Тема 1. **Основные понятия технологии**	***Содержание***	**8**	**ПК 1.1** **ОК 1-9**
	Информационные системы (ИС): основные понятия, классификация, структура и области применения. Понятие жизненного цикла ПО ИС. Процессы жизненного цикла. Эффективность ИС. Стадии и этапы проектирования ИС. Типовое проектирование ИС, типовое проектное решение (ТПР).	2	
	Современные платформы для создания, исполнения и Управления ИС. Архитектура ИС. Понятие архитектуры информационных систем. Типы архитектур. Значение программного обеспечения в информационных системах. Характеристики качества программного обеспечения. Фреймворки.	2	
	Тематика практических занятий	**4**	
	Разработка архитектуры приложения согласно принципу единственности ответственности. Разработка архитектуры приложения согласно принципу открытости/закрытости.	2	
	Разработка архитектуры приложения согласно принципу разделения интерфейсов. Разработка архитектуры приложения.	2	
Тема 2. **Применение UML для проектирования бизнес-процессов и информационных ресурсов**	***Содержание***	**10**	**ПК 1.1** **ОК 1-9**
	Анализ и моделирование бизнес-процессов. Унифицированный язык визуального моделирования (UML). Основные понятия. Концептуальная модель UML. Средства описания статических аспектов поведения системы. Классы, свойства классов. Диаграммы классов.	2	

	Прецеденты и связи между ними. Средства описания динамических аспектов поведения системы. Описания взаимодействий между объектами. Диаграммы последовательности. Кооперативные диаграммы.	2	
	Методология применения UML для проектирования ИС (RUP- Rational Unified Process). Концепции RUP. Архитектура RUP. Автоматизированное проектирование информационных систем на основе CASE- технологии. Назначение CASE-средств. Состав и классификация CASE-средств. Технология внедрения CASE-средств. Проектирование логической и физической модели ИС и модели баз данных.	2	
	Тематика практических занятий	**10**	
	Анализ предметной области и формирование требований для разработки типовой ИС виртуальной организации. Постановка задачи. Исследование и выбор инструментальных средств моделирования ИС.	2	
	Построение диаграммы классов.	2	
	Построение диаграммы деятельности.	2	
	Построение диаграммы прецедентов.	2	
	Построение диаграммы последовательности.	2	
Тема 3. **Разработка проектных документов**	***Содержание***	**10**	**ПК 1.1** **ОК 1-9**
	Разработка технического задания (ТЗ). Общие положения. Описание назначения и целей создания (развития) системы. Разработка ТЗ.	2	
	Характеристика объекта автоматизации. Формирование требований к системе. Разработка инструкции пользователя ИС. Разработка документации для технических специалистов.	2	
	Тематика практических занятий	**6**	
	Разработка технического задания.	2	
	Разработка инструкции пользователя.	2	
	Разработка технической документации.	2	
Тема 4. **Дизайн интерфейса информационных ресурсов**	***Содержание***	**12**	**ПК 1.1** **ОК 1-9**
	Понятие интерфейса ИС. Основные принципы разработки интерфейса ИС. Этапы проектирования.	2	
	Дизайн визуальных компонентов интерфейса ИС. Виды визуальных компонент. Принципы расположения Управляющих компонентов интерфейса. Цветовое и шрифтовое оформление интерфейса.	2	

	Тематика практических занятий	**10**	
	Создание прототипа интерфейса ИС.	2	
	Создание руководства по стилю.	2	
	Проектирование управляющих элементов интерфейса.	2	
	Разработка пользовательских сценариев.	2	
	Разработка навигационной схемы.	2	
Самостоятельная работа: – Анализ предметной области – Изучение и применение метода персонажей – Проектирование экранных форм приложения (создание прототипа приложения) – Создание технического задания на разработку информационной системы		**10**	
Итоговая аттестация в форме дифференцированного зачёта			
РАЗДЕЛ 2.			
МДК 01.02 Разработка интерфейсов пользователя		***68***	
Тема 1. Основы веб-технологий и вёрстки	***Содержание***	**16**	**ПК 1.2. ПК 1.4. ОК 1-9**
	Введение в веб-технологии: HTML, CSS, JavaScript. Основы HTML: разметка	2	
	Основы CSS: стилизация элементов. Позиционирование элементов в CSS.	2	
	Тематика практических занятий	**12**	
	Создание базовой структуры HTML-страницы.	2	
	Вёрстка статьи с использованием семантических тегов.	2	
	Создание таблицы и формы для сбора данных.	2	
	Стилизация текста и базовых элементов.	2	
	Работа с блочной моделью (box model).	2	
	Позиционирование элементов.	2	
Тема 2. Адаптивная и отзывчивая вёрстка	***Содержание***	**18**	**ПК 1.2. ПК 1.4. ОК 1-9**
	Flexbox: гибкий макет. Grid Layout: двумерная вёрстка.	2	
	Адаптивный дизайн: медиа-запросы. Отзывчивые изображения и шрифты.	2	
	Тематика практических занятий	**14**	
	Вёрстка навигационного меню с Flexbox.	2	
	Создание галереи карточек с Flexbox.	2	
	Вёрстка макета страницы с Grid Layout.	2	
	Создание сложной сетки с Grid.	2	

	Адаптация макета под мобильные устройств.	2	
	Использование отзывчивых изображений.	2	
	Отзывчивая типографика.	2	
Тема 3. Интерактивность и JavaScript	***Содержание***	**16**	**ПК 1.2. ПК 1.4. ОК 1-9**
	Основы JavaScript для интерфейсов. Обработка событий в JavaScript.	2	
	Динамическое изменение интерфейса с JS.	2	
	Анимации и переходы в CSS и JS.	2	
	Тематика практических занятий	**10**	
	Взаимодействие с DOM: поиск и изменение элементов.	2	
	Обработка кликов и наведения мыши.	2	
	Создание динамического списка задач.	2	
	Анимация элементов с CSS.	2	
	JavaScript-анимации.	2	
Тема 4. Компоненты интерфейса и фреймворки	***Содержание***	**6**	**ПК 1.2. ПК 1.4. ОК 1-9**
	Создание UI-компонентов: кнопки, карточки, модальные окна. Формы и	2	
	CSS-фреймворки: Bootstrap.	2	
	Тематика практических занятий	**2**	
	Вёрстка страницы с использованием сетки Bootstrap.	2	
Самостоятельная работа: – Создание персональной веб-страницы – Создание интерактивного слайдера изображений – Редизайн существующего сайта		**12**	
Итоговая аттестация в форме дифференцированного зачёта			
РАЗДЕЛ 3.			
МДК 01.03 Тестирование информационных ресурсов		60	
Тема 1. Модульное тестирование	***Содержание***	**10**	
	Введение в теорию тестирования	2	**ПК 1.2. ПК 1.3. ПК 1.4. ОК 1-9**
	Тестовая документация в QA	2	
	Техники тест-дизайна.	2	
	Понятие программной ошибки.	2	

	Модульное тестирование. Процесс модульного тестирования, его задачи и цели.	2
	Тематика практических занятий	**8**
	Формирование тест-кейсов и чек-листа. Создание чек-листа и базовых тест-кейсов на примере реального продукта для покрытия основных сценариев.	2
	Создание библиотеки классов с применением Test Driven Development. Написание unit-тестов до реализации основной логики (red-green-refactor).	2
	Проектирование тест-кейсов с параметризацией при unit-тестировании	2
	Обработка ошибок и исключений в unit-тестах. Документирование результатов тестирования.	2
Тема 2. Интеграционное **тестирование**	***Содержание***	**10**
	Интеграционное и API-тестирование.	2
	Автоматизация тестирования. Основные инструменты автоматизации тестирования.	2
	Автоматизация нефункционального тестирования при использовании различных интерфейсов (API, CLI, GUI и другие).	2
	UI автоматизация и Selenium принципы.	2
	Документирование процесса тестирования и контроль качества	2
	Тематика практических занятий	**22**
	Настройка тестового окружения для API.	2
	Сценарии API-тестирования с мокингом	2
	Автоматизация: функциональная декомпозиция с параметризацией	2
	Автоматизация: Data Driven подход	2
	Автоматизация тестирования через ключевые слова (Keyword-Driven Testing).	2
	Автоматизация: BDD для приемочного тестирования	2
	Нефункциональное тестирование и документирование дефектов	2

	Автоматизация UI-тестирования с Selenium	2	
	Создание устойчивых UI-тестов с Selenium для регрессии.	2	
	Сквозные тесты (End-to-End, E2E) с Selenium	2	
	Настройка сборки отчетов при тестировании	2	
1. Самостоятельная работа: 2. Провести обследование интерфейса сервиса для оценки эффективности решения задач пользователя через заполнение и отправку формы. Выполнить верификацию соответствия UI/UX стандартам usability, выявить дефекты удобства использования и задокументировать их в баг-репортах с конструктивными рекомендациями. 3. Выполнить тестирование экранной формы методом попарного тестирования для покрытия всех критических комбинаций параметров минимальным набором тест-кейсов. Научиться создавать структурированную тестовую документацию по образцу, обеспечивая ясность, полноту, проверяемость, логичность и неизбыточность тест-кейсов. 4. Разработать чек-лист для тестирования базовых API операций (Create, Read, Update, Delete). Покрыть типовые сценарии: валидные данные, граничные значения, ошибки валидации, отсутствующие обязательные поля. 5. Разделить входные данные произвольной API формы на классы эквивалентности и создать минимальный набор тест-кейсов. Применить технику ЕР для полей: текст (короткий/длинный), числа (валидные/инвалидные), email (корректный/некорректный). 6. Определить граничные значения для числовых параметров API (min, max, min+1, max-1) и создать тест-кейсы. Протестировать поведение системы на границах допустимых диапазонов с документацией ожидаемых кодов ответа (200/400/422). 7. Тестирование статус-кодов REST API. Создать чек-лист проверки HTTP статус-кодов для типовых API операций без эндпоинтов: 200/201 (успех), 400/401/403/404/422 (ошибки), 500 (сервер). Документировать ожидаемые заголовки Content-Type и форматы ошибок (JSON/XML). 8. Разработать контракт API (OpenAPI YAML) для типичного CRUD ресурса и создать тест-кейсы по схеме. Проверить соответствие ответов API контрактам: обязательные поля, типы данных, nullable свойства, примеры ответов. 9. Создать автотесты Selenium для типичной веб-формы (login/register): поиск элементов (ID/class/XPath), заполнение полей, клики, валидация сообщений. Настроить тестовое окружение 10. Автоматизировать тест-кейсы попарного тестирования формы регистрации через Selenium. Сохранить результаты в отчет с скриншотами ошибок валидации.	**10**		
Итоговая аттестация в форме дифференцированного зачёта			

3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

3.1. Требования к минимальному материально-техническому обеспечению

Реализация профессионального модуля предполагает наличие лаборатории программного обеспечения и сопровождения компьютерных систем.

Реализация профессионального модуля предполагает обязательную производственную практику.

Оборудование и технологическое оснащение рабочих мест:

- Автоматизированные рабочие места обучающихся;
- Автоматизированное рабочее место преподавателя;
- Интерактивная доска;
- Программное обеспечение общего и профессионального назначения.

3.2. Информационное обеспечение обучения

Основные источники:

1. Литвинская, О. С., Администрирование информационных ресурсов : учебное пособие / О. С. Литвинская, Л. А. Васин. — Москва : КноРус, 2024. — 227 с. — ISBN 978-5-406-12343-0. — URL: https://book.ru/book/951856. — Текст : электронный.
2. Шитов, В. Н., Проектирование и разработка интерфейсов пользователя : учебное пособие / В. Н. Шитов, К. Е. Успенский. — Москва : КноРус, 2023. — 294 с. — ISBN 978-5-406-10392-0. — URL: https://book.ru/book/945981. — Текст : электронный.
3. Шитов, В. Н., Устройство и функционирование информационной системы : учебник / В. Н. Шитов. — Москва : КноРус, 2024. — 333 с. — ISBN 978-5-406-12882-4. — URL: https://book.ru/book/953436. — Текст : электронный.
4. Япарова, Ю. А., Информационные технологии. Практикум с примерами решения задач : учебно-практическое пособие / Ю. А. Япарова. — Москва : КноРус, 2022. — 226 с. — ISBN 978-5-406-09774-8. — URL: https://book.ru/book/943670 (дата обращения: 10.03.2024). — Текст : электронный.
5. Орлов, С. А. Технологии разработки программного обеспечения : учебник для вузов / С. А. Орлов, Б. Я. Цилькер. - 4-е изд. - Санкт-Петербург : Питер, 2021. - 608 с.
6. Гагарина Л.Г., Кокорева Е.В., Сидорова-Виснадул Б.Д. Технология разработки программного обеспечения.- М: Издательство: Инфра-М, 2022
7. Галина Иванова. Технология программирования. Проектирование. Комплексное тестирование. Программная документация. Издательство: Кнорус, 2022 г.

Дополнительная литература:

1. Попов, А. А., Эргономика пользовательских интерфейсов в информационных системах : учебное пособие / А. А. Попов. — Москва : КноРус, 2023. — 304 с. — ISBN 978-5-406-10630-3. — URL: https://book.ru/book/945705. — Текст : электронный.
2. Климова, Г. Л., Основы дизайн-мышления для ИТ-специалистов : учебник / Г. Л. Климова. — Москва : КноРус, 2024. — 222 с. — ISBN 978- 5-406-13007-0. — URL: https://book.ru/book/953658. — Текст : электронныйШитов, В. Н. Разработка информационного контента (по отраслям) : учебное пособие / В.Н. Шитов. — Москва : ИНФРА-М, 2022. — 178 с.
3. Кузниченко, М. А. Основы баз данных : учебно-методическое пособие / М. А. Кузниченко. - 2-е изд., стер. - Москва : Флинта, 2022. - 102 с. - ISBN 978-5-9765-5139-8.

4. Морозова, Ю. В. Тестирование программного обеспечения : учебное пособие / Ю. В. Морозова. - Томск : Эль-Контент, 2019.

5. Галиаскаров, Э. Г. Анализ и проектирование систем с использованием UML : учебное пособие для вузов / Э. Г. Галиаскаров, А. С. Воробьев. — Москва : Издательство Юрайт, 2022. — 125 с. — (Высшее образование). — ISBN 978-5-534-14903-6.

6. Лаврищева, Е. М. Программная инженерия. Парадигмы, технологии и CASE-средства : учебник для вузов / Е. М. Лаврищева. — 2-е изд., испр. — Москва : Издательство Юрайт, 2022. — 280 с. — (Высшее образование). — ISBN 978-5-534-01056-5.

3.3. Общие требования к организации образовательного процесса

Освоению данного модуля предшествует изучение дисциплин общеобразовательного, социально-гуманитарного цикла, а также общепрофессиональной дисциплины: «Операционные системы и среды».

В рамках учебных курсов предусмотрены практические работы с использованием персональных компьютеров.

Производственная практика проводится концентрированно. Раздел модуля «Производственная практика» является обязательным и представляет собой вид учебных занятий, непосредственно ориентированных на профессионально-практическую подготовку обучающихся.

Практики могут проводиться в сторонних организациях или в отделениях и в кабинетах образовательного учреждения, обладающих необходимым кадровым и материально-техническим потенциалом. Аттестация по итогам практики осуществляется на основе отчета о проделанной работе и публичной его защиты.

3.4. Кадровое обеспечение образовательного процесса

Реализация образовательной программы обеспечивается педагогическими работниками образовательной организации, а также лицами, привлекаемыми к реализации образовательной программы на иных условиях, в том числе из числа руководителей и работников организаций, направление деятельности которых соответствует области профессиональной деятельности 06 Связь, информационные и коммуникационные технологии (имеющих стаж работы в данной профессиональной области не менее одного года).

Квалификация педагогических работников образовательной организации должна квалификационным требованиям, указанным в квалификационных справочниках и (или) профессиональных стандартах (при наличии).

Педагогические работники, привлекаемые к реализации образовательной программы, должны получать дополнительное профессиональное образование по программам повышения квалификации не реже одного раза в три года с учетом расширения спектра профессиональных компетенций, в том числе в форме стажировки в организациях, направление деятельности которых соответствует области профессиональной деятельности, 06 Связь, информационные и коммуникационные технологии, а также в других областях профессиональной деятельности и (или) сферах профессиональной деятельности при условии соответствия полученных компетенций требованиям к квалификации педагогического работника.

Доля педагогических работников (в приведенных к целочисленным значениям ставок), имеющих опыт деятельности не менее одного года в организациях, направление деятельности которых соответствует области профессиональной деятельности, 06 Связь, информационные и коммуникационные технологии, в общем числе педагогических работников, обеспечивающих освоение обучающимися профессиональных модулей образовательной программы, должна быть не менее 25 процентов.

4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

Код и наименование профессиональных и общих компетенций,	**Критерии оценки**	**Методы оценки**
ОК 01. Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам	Обоснованность постановки цели, выбора и применения методов и способов решения профессиональных задач. Адекватная оценка и самооценка эффективности и качества выполнения профессиональных задач.	Наблюдение за выполнением работ
ОК 02. Использовать современные средства поиска, анализа и интерпретации информации, и информационные технологии для выполнения задач профессиональной деятельности	Использование различных источников, включая электронные ресурсы, медиаресурсы, Интернет-ресурсы, периодические издания по специальности для решения профессиональных задач. Эффективность использования информационно- коммуникационных технологий в профессиональной деятельности согласно формируемым умениям и получаемому практическому опыту.	Наблюдение за выполнением работ на практике
ОК 03. Планировать и реализовывать собственное профессиональное и личностное развитие, предпринимательскую деятельность в профессиональной сфере, использовать знания по финансовой грамотности в различных жизненных ситуациях	Демонстрация ответственности за принятые решения обоснованность самоанализа и коррекция результатов собственной работы.	Наблюдение за выполнением работ на практике
ОК 04. Эффективно взаимодействовать и работать в коллективе и команде	Взаимодействовать с обучающимися, преподавателями и мастерами в ходе обучения, с руководителями учебной и производственной практик. Обоснованность анализа работы членов команды.	Наблюдение за выполнением работ на практике
ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей	Демонстрировать грамотность устной и письменной речи, ясность формулирования и изложения мыслей	Наблюдение за выполнением работ на практике

ОК 06. Проявлять гражданско- патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей, в том числе с учетом гармонизации межнациональных и межрелигиозных отношений,	Соблюдение норм поведения во время учебных занятий и прохождения учебной и производственной практик.	Наблюдение за выполнением работ на практике
ОК 07. Содействовать сохранению окружающей среды, ресурсосбережению, применять знания об изменении климата, принципы бережливого производства, эффективно действовать в	Оказывать первую помощь	Наблюдение за выполнением работ на практике
ОК 08. Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической	Эффективность использовать средств физической культуры для сохранения и укрепления здоровья при выполнении профессиональной деятельности.	Наблюдение за выполнением работ на практике
ОК 09. Пользоваться профессиональной документацией на государственном и иностранном языках	Эффективность использования информационно-коммуникационных технологий в профессиональной деятельности согласно формируемым умениям и получаемому практическому опыту	Наблюдение за выполнением работ на практике

ПК 1.1. Проектировать компоненты информационных ресурсов	Оценка «**отлично**» - изучены требования заказчика по результатам анкет и интервью; изучены типовые решения, обосновано, выбрано и согласовано с заказчиком оптимальное решение; построена графическая нотация описания бизнес процессов. Оценка «**хорошо**» - изучены требования заказчика по результатам анкет и интервью; изучены типовые решения, выбрано и согласовано с заказчиком оптимальное решение; построена графическая нотация с некоторыми недочетами Оценка «**удовлетворительно**» - изучены требования заказчика по результатам анкет и интервью; изучены типовые решения, выбрано и согласовано с заказчиком одно решение; графическая нотация содержит ряд неверных решений	Дифференцированный зачет в форме: практическое задание по постановке задачи по обработке информации в заданной сфере деятельности, анализу предметной области, сбору и обработке исходной информации.
ПК 1.2. Разрабатывать интерфейсы пользователя	Оценка **«отлично»** - разработан и реализован отзывчивый дизайн веб – приложения с использованием специальных графических и редакторов, применением относительных размеров, контрольных точек и вложенных объектов; макет корректно отображается на различных устройствах; заданные элементы интегрированы в дизайн оптимальным образом; разработанный дизайн полностью соответствует современным стандартам. Оценка **«хорошо»** - разработан и реализован отзывчивый дизайн веб – приложения с использованием специальных графических редакторов, применением нескольких методов; макет корректно отображается на большинстве устройств; заданные элементы интегрированы в общий дизайн; разработанный дизайн соответствует современным стандартам. Оценка **«удовлетворительно»** - разработан и реализован отзывчивый дизайн веб – приложения с использованием специальных графических редакторов, применением нескольких методов; большинство заданных элементов интегрировано в дизайн; макет корректно отображается на одном устройстве; разработанный дизайн в основном соответствует современным стандартам.	Дифференцированный зачет в форме: практическое задание по тестированию информационной системы. Наблюдение за выполнением различных видов работ.

ПК 1.3. Интегрировать программный код в соответствующую инфраструктуру	Оценка **«отлично»** - В результате интеграции программного кода, приложение функционирует правильно, согласно заявленным требованиям. Новые функции доступны. Система работает без сбоев. Оценка **«хорошо»** - В результате интеграции программного кода, приложение функционирует правильно, но не обеспечивает возможности выполнения всех регламентных функций, описанных в требовании к разработке веб-приложения. Оценка **«удовлетворительно»** - В результате интеграции программного кода, приложение функционирует частично и не обеспечивает возможности выполнения всех регламентных функций, описанных в требовании к разработке веб-приложения.	Дифференцированный зачет в форме: практическое задание по тестированию информационной системы. Наблюдение за выполнением различных видов работ.
ПК 1.4. Использовать систему контроля версий в процессе коллективной (параллельной) разработки	Оценка «**отлично**» - результаты тестирования сохранены в системе контроля версий; по результатам тестирования сделаны выводы и внесены предложения по рефакторингу кода; выполнена отладка приложения; результаты отладки сохранены в системе контроля версий; сделаны выводы по результатам отладки. Оценка «**хорошо**» - результаты тестирования сохранены в системе контроля версий; по результатам тестирования сделаны выводы; выполнена отладка приложения; результаты отладки сохранены в системе контроля версий; сделаны выводы по результатам отладки. Оценка «**удовлетворительно**» - результаты тестирования сохранены в системе контроля версий; выполнена отладка приложения; результаты отладки сохранены в системе контроля версий.	Дифференцированный зачет в форме: практическое задание по анализу интересов клиента (изложенным в задании); разработке и оформлению алгоритма решения задачи по обработке информации. Наблюдение за выполнением различных видов работ.

ПК 1.5. **Выполнять процедуры тестирования программного кода**	Оценка «**отлично**» - выполнено тестирование веб – приложения в соответствии с тест – планом; результаты тестирования сохранены в системе контроля версий; по результатам тестирования сделаны выводы и внесены предложения по рефакторингу кода; выполнена отладка приложения; результаты отладки сохранены в системе контроля версий; сделаны выводы по результатам отладки. Оценка «**хорошо**» - выполнено тестирование веб – приложения в соответствии с тест – планом; результаты тестирования сохранены в системе контроля версий; по результатам тестирования сделаны выводы; выполнена отладка приложения; результаты отладки сохранены в системе контроля версий; сделаны выводы по результатам отладки. Оценка «**удовлетворительно**» - выполнено тестирование веб – приложения в соответствии с тест – планом; результаты тестирования сохранены в системе контроля версий; выполнена отладка приложения; результаты отладки сохранены в системе контроля версий.	Дифференцированный зачет в форме: практическое задание по анализу интересов клиента (изложенным в задании); разработке и оформлению алгоритма решения задачи по обработке информации. Наблюдение за выполнением различных видов работ.

МИНИСТЕРСТВО ОБРАЗОВАНИЯ СВЕРДЛОВСКОЙ ОБЛАСТИ
ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ СВЕРДЛОВСКОЙ ОБЛАСТИ
«ЕКАТЕРИНБУРГСКИЙ МОНТАЖНЫЙ КОЛЛЕДЖ»

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
МДК 01.01 Проектирование информационных ресурсов
для специальности
09.02.09 «Веб-разработка»

Екатеринбург
2026

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Данный фонд оценочных средств подготовлен для сопровождения МДК 01.01 «Проектирование информационных ресурсов», направленного на формирование профессиональных компетенций в области проектирования и разработки информационных систем.

Основная цель пособия — предоставление обучающимся комплекса теоретических знаний и практических навыков, необходимых для освоения ключевых аспектов проектирования информационных ресурсов, включая разработку архитектуры приложений, моделирование бизнес-процессов и создание пользовательских интерфейсов.

Структура пособия

Пособие построено на основе двух взаимосвязанных частей, обеспечивающих комплексное освоение дисциплины.

Теоретическая часть охватывает фундаментальные основы проектирования информационных ресурсов. В ней рассматриваются ключевые аспекты создания информационных систем: от базовых понятий и классификации до современных методологий проектирования. Особое внимание уделяется изучению жизненного цикла информационных систем, принципам построения архитектуры приложений, основам UML-моделирования и проектированию пользовательских интерфейсов. Теоретический блок также включает изучение стандартов разработки проектной документации и современных платформ для создания информационных систем.

Практическая часть направлена на закрепление теоретических знаний через выполнение практических заданий и разработку реальных проектов. Она включает работу над анализом предметной области, построением UML-диаграмм, разработкой технической документации и проектированием интерфейсов.

Завершающий этап обучения предусматривает проведение итоговой аттестации в форме дифференцированного зачёта, который позволяет оценить уровень освоения материала и сформированные компетенции.

Учебные цели и задачи

Основные учебные цели включают:

- Формирование системных знаний в области проектирования информационных ресурсов и информационных систем
- Освоение современных методологий и инструментов проектирования информационных систем
- Развитие профессиональных компетенций в сфере анализа и моделирования бизнес-процессов
- Приобретение практических навыков разработки проектной документации
- Формирование умений по проектированию пользовательских интерфейсов
- Развитие способности к самостоятельному проектированию информационных систем

Задачи курса состоят в следующем:

- Освоение теоретических основ проектирования информационных систем
- Изучение принципов построения архитектуры приложений
- Освоение методологии UML-моделирования
- Формирование навыков работы с современными CASE-средствами
- Развитие умений по разработке технической документации
- Освоение принципов проектирования пользовательских интерфейсов
- Приобретение практических навыков анализа предметной области
- Формирование компетенций в области разработки требований к информационным системам
- Развитие способности к созданию прототипов информационных систем

- Освоение методов оценки эффективности информационных систем

Контроль знаний
Контроль знаний осуществляется поэтапно:
- Текущие проверки знаний в форме вопросов и опросов.
- Практические работы с последующим обсуждением.
- Итоговая аттестация в форме дифференцированного зачета.

Оценочная деятельность
Оценка проводится по следующим критериям:
- Качество выполнения практических работ.
- Уровень владения материалами курса.
- Активность и инициативность на занятиях.
- Результативность индивидуальной проектной работы.

В итоге данный фонд оценочных средств обеспечит качественное сопровождение образовательного процесса по проектированию информационных ресурсов, поможет освоить современные технологии и методологии разработки информационных систем, а также подготовит выпускников к успешной профессиональной деятельности в сфере информационных технологий. Обучающиеся получат необходимые компетенции для анализа, проектирования и разработки информационных систем, научатся эффективно применять современные инструменты и методологии проектирования, что позволит им успешно решать профессиональные задачи в условиях реального производства.

Краткое изложение планов хода лекционных занятий:
1. Информационные системы (ИС): основные понятия, классификация, структура и области применения. Понятие жизненного цикла ПО ИС. Процессы жизненного цикла. Эффективность ИС. Стадии и этапы проектирования ИС. Типовое проектирование ИС, типовое проектное решение (ТПР).
*Образовательный результа*т: понимает основные понятия, классификацию и структуру информационных систем; владеет представлениями о жизненном цикле ПО ИС и его процессах, стадиях и этапах проектирования; умеет оценивать эффективность ИС; знает принципы типового проектирования ИС и суть типового проектного решения (ТПР).

2. Современные платформы для создания, исполнения и управления ИС. Архитектура ИС. Понятие архитектуры информационных систем. Типы архитектур. Значение программного обеспечения в информационных системах. Характеристики качества программного обеспечения. Фреймворки.

Образовательный результат: знает современные платформы для создания, исполнения и управления ИС; понимает понятие архитектуры ИС и различия между её типами (монолитной, клиент-серверной, микросервисной и др.); оценивает значение и ключевые характеристики качества ПО в ИС; умеет подбирать подходящие фреймворки для решения задач проектирования.

3. Анализ и моделирование бизнес-процессов. Унифицированный язык визуального моделирования (UML). Основные понятия. Концептуальная модель UML. Средства описания статических аспектов поведения системы. Классы, свойства классов. Диаграммы классов.

Образовательный результат: владеет основами анализа и моделирования бизнес-процессов с применением UML; понимает концептуальную модель UML и средства описания

статических аспектов системы; умеет определять классы и их свойства, строить диаграммы классов для формализации структуры ИС.

4. Прецеденты и связи между ними. Средства описания динамических аспектов поведения системы. Описания взаимодействий между объектами. Диаграммы последовательности. Кооперативные диаграммы.
Образовательный результат: понимает концепцию прецедентов и способы описания связей между ними; умеет моделировать динамические аспекты поведения системы и взаимодействия между объектами с помощью UML; строит диаграммы последовательности и кооперативные диаграммы; интерпретирует их для анализа логики работы ИС.

5. Методология применения UML для проектирования ИС (RUP — Rational Unified Process). Концепции RUP. Архитектура RUP. Автоматизированное проектирование информационных систем на основе CASE-технологии. Назначение CASE-средств. Состав и классификация CASE-средств. Технология внедрения CASE-средств. Проектирование логической и физической модели ИС и модели баз данных.
Образовательный результат: знает концепции и архитектуру методологии RUP, умеет применять UML в рамках этой методологии для проектирования ИС; понимает назначение, состав и классификацию CASE-средств, владеет технологией их внедрения; проектирует логическую и физическую модели ИС, а также модели баз данных с использованием CASE-технологий.

6. Разработка технического задания (ТЗ). Общие положения. Описание назначения и целей создания (развития) системы. Разработка ТЗ.
Образовательный результат: понимает структуру и содержание технического задания, знает нормативные требования к его оформлению; умеет описывать назначение и цели создания (развития) ИС; составляет ТЗ, согласовывает его с заинтересованными сторонами и использует как основу для дальнейшего проектирования.

7. Характеристика объекта автоматизации. Формирование требований к системе. Разработка инструкции пользователя ИС. Разработка документации для технических специалистов.
Образовательный результат: умеет анализировать объект автоматизации и формулировать требования к ИС с учётом бизнес-задач; разрабатывает инструкцию пользователя с соблюдением принципов юзабилити; составляет техническую документацию для специалистов (администраторов, разработчиков), обеспечивает её полноту и однозначность; согласовывает и утверждает комплект документации.

8. Понятие интерфейса ИС. Основные принципы разработки интерфейса ИС. Этапы проектирования.
Образовательный результат: понимает сущность интерфейса ИС и его роль в системе; знает основные принципы разработки интерфейсов (пользователь-центричность, доступность, консистентность, минимизация усилий пользователя); владеет этапами проектирования интерфейса (анализ, прототипирование, дизайн, тестирование); учитывает особенности целевой аудитории и задачи пользователей при разработке.

9. Дизайн визуальных компонентов интерфейса ИС. Виды визуальных компонент. Принципы расположения управляющих компонентов интерфейса. Цветовое и шрифтовое оформление интерфейса.
Образовательный результат: различает виды визуальных компонентов интерфейса (кнопки, поля ввода, меню, карточки, слайдеры, модальные окна и т.д.); применяет принципы расположения управляющих элементов для обеспечения удобства использования; подбирает

цветовые схемы и шрифты с учётом требований доступности (WCAG) и корпоративного стиля; создаёт гармоничный и функциональный дизайн интерфейса; оценивает визуальную композицию и читаемость элементов; использует инструменты дизайна (Figma, Adobe XD и др.) для прототипирования.

Краткое изложение планов хода практических занятий и примерные задания:
1. Разработка архитектуры приложения согласно принципу единственности ответственности. Разработка архитектуры приложения согласно принципу открытости/закрытости
Ход занятия: разбор принципов SOLID (SRP и OCP); анализ примера плохо спроектированного кода; рефакторинг кода с соблюдением SRP (разделение классов по обязанностям) и OCP (добавление функционала через наследование/интерфейсы без изменения существующего кода).
Задание: взять готовый фрагмент кода (например, класс OrderProcessor, который обрабатывает заказ, сохраняет в БД и отправляет email) и перепроектировать его: выделить отдельные классы OrderService, OrderRepository, EmailService; затем добавить новый тип заказа (например, DiscountedOrder) без изменения базового класса.

2. Разработка архитектуры приложения согласно принципу разделения интерфейсов. Разработка архитектуры приложения
Ход занятия: объяснение принципа ISP; выявление «толстых» интерфейсов в примере кода; разделение интерфейсов на более мелкие и специализированные; демонстрация влияния ISP на гибкость архитектуры.
Задание: дан интерфейс IMachine с методами Print(), Scan(), Fax() — реализовать разделение на IPrinter, IScanner, IFax; создать классы SimplePrinter и MultiFunctionDevice, реализующие только необходимые интерфейсы.

3. Анализ предметной области и формирование требований для разработки типовой ИС виртуальной организации. Постановка задачи. Исследование и выбор инструментальных средств моделирования ИС
Ход занятия: сбор информации о виртуальной организации (цели, процессы, пользователи); формулирование бизнес-требований и функциональных требований; обзор инструментов моделирования (PlantUML, Visual Paradigm, Lucidchart, draw.io); сравнение возможностей и выбор подходящего инструмента.
Задание: описать предметную область интернет-магазина; сформулировать 5–7 функциональных и 3–5 нефункциональных требований; выбрать инструмент моделирования и обосновать выбор.

4. Построение диаграммы классов
Ход занятия: повторение синтаксиса UML для диаграмм классов; определение классов, атрибутов, методов и отношений (ассоциация, агрегация, наследование); построение диаграммы в выбранном инструменте.
Задание: построить диаграмму классов для системы «Библиотека»:
классы Book, Member, Loan, Library; указать атрибуты и методы; обозначить отношения между классами.

5. Построение диаграммы деятельности
Ход занятия: изучение элементов диаграммы деятельности (начальное/конечное состояние, действия, решения, потоки); моделирование бизнес-процесса; построение диаграммы для сценария «Оформление заказа».
Задание: создать диаграмму деятельности для процесса «Бронирование отеля»: шаги - выбор дат, проверка доступности, ввод данных гостя, подтверждение бронирования, оплата.

6. Построение диаграммы прецедентов
Ход занятия: определение акторов и прецедентов; установление отношений include и extend; построение диаграммы прецедентов для информационной системы.
Задание: разработать диаграмму прецедентов для системы «Онлайн-обучение»: акторы — студент, преподаватель, администратор; прецеденты — просмотр курсов, сдача теста, создание курса, оценка заданий.

7. Построение диаграммы последовательности
Ход занятия: разбор элементов диаграммы последовательности (объекты, линии жизни, сообщения); моделирование взаимодействия объектов во времени; построение диаграммы для типичного сценария.
Задание: создать диаграмму последовательности для сценария «Оплата заказа»: объекты — User, PaymentService, OrderService, NotificationService; показать последовательность вызовов и ответов.

8. Разработка технического задания
Ход занятия: структура ТЗ (назначение, требования, состав, условия эксплуатации, технико-экономические показатели); заполнение разделов на примере выбранной системы; согласование содержания с «заказчиком».
Задание: составить ТЗ для ИС «Учёт оборудования в компании»: включить разделы «Введение», «Назначение системы», «Требования к функционалу», «Требования к надёжности», «Состав и содержание работ».

9. Разработка инструкции пользователя
Ход занятия: принципы написания понятных инструкций; структура документа (введение, шаги работы, скриншоты, FAQ); примеры удачных и неудачных формулировок.
Задание: написать инструкцию пользователя для мобильного приложения «Трекер привычек»: разделы — «Регистрация», «Добавление привычки», «Отслеживание прогресса», «Настройки уведомлений».

10. Разработка технической документации
Ход занятия: виды технической документации (API-документация, архитектура, развёртывание); стандарты оформления (Swagger, Markdown); составление описания модулей и интерфейсов.
Задание: подготовить фрагмент технической документации для REST API интернет-магазина: описать эндпоинты /api/products, /api/orders, методы GET/POST, форматы запросов и ответов (JSON), коды статусов.

11. Создание прототипа интерфейса ИС
Ход занятия: инструменты прототипирования (Figma, Adobe XD); принципы низко- и высокодетализированных прототипов; создание каркаса и интерактивного прототипа.
Задание: сделать интерактивный прототип главной страницы ИС «Корпоративный портал»: включить меню, виджет новостей, календарь событий, поиск; настроить переходы между экранами.

12. Создание руководства по стилю
Ход занятия: компоненты дизайн-системы (цвета, шрифты, отступы, кнопки, иконки); правила оформления UI-кита; экспорт стилей для разработчиков.

Задание: разработать руководство по стилю для веб-приложения «Финансовый калькулятор»: задать палитру (3–4 цвета), шрифты (основной и заголовок), размеры отступов (8px-сетка), стили кнопок (основная, вторичная, отключённая).

13. Проектирование управляющих элементов интерфейса
Ход занятия: типы управляющих элементов (кнопки, переключатели, слайдеры, поля ввода); принципы доступности и юзабилити; размещение элементов на экране.
Задание: спроектировать форму «Регистрация пользователя» с полями: имя, email, пароль, подтверждение пароля, чекбокс согласия; добавить валидацию и подсказки; разместить элементы с учётом читаемости и логики заполнения.

14. Разработка пользовательских сценариев
Ход занятия: понятие user story и use case; структура сценария (актор, цель, шаги, результат); связь сценариев с функциональными требованиями.
Задание: написать 3 пользовательских сценария для ИС «Онлайн-банк»: «Перевод между счетами», «Оплата коммунальных услуг», «Просмотр истории операций»; для каждого указать актора, цель и пошаговые действия.

15. Разработка навигационной схемы
Ход занятия: типы навигации (горизонтальная, вертикальная, хлебные крошки, табы); проектирование user flow; визуализация структуры сайта/приложения.
Задание: создать навигационную схему для веб-сайта университета: главные разделы — «О вузе», «Абитуриентам», «Студентам», «Преподавателям», «Наука»; построить карту сайта и user flow для сценария «Подача заявки на стипендию».

Критерии оценивания.
Критерии оценивания практических заданий:

Корректность выполнения задания
- правильность применения принципов проектирования в архитектуре приложения;
- соответствие разработанной архитектуры поставленной задаче;
- отсутствие критических ошибок, нарушающих работу системы или логику взаимодействия компонентов.

Качество моделирования
- корректность построения диаграмм UML (классов, деятельности, прецедентов, последовательности) согласно синтаксису и семантике UML;
- полнота отображения сущностей, отношений и взаимодействий;
- логическая связность элементов диаграммы и их соответствие предметной области.

Полнота выполнения
- выполнение всех пунктов задания;
- наличие всех требуемых элементов (классов, методов, атрибутов, сообщений, сценариев и т.д.);
- завершённость работы: от постановки задачи до итогового результата.

Соответствие стандартам и методологиям
- соблюдение принципов проектирования ПО;
- использование общепринятых нотаций и шаблонов при моделировании;
- соответствие структуры и содержания документации принятым стандартам.

Качество технической документации

- чёткость и однозначность формулировок в ТЗ, инструкции пользователя и технической д окументации;
- структурированность документа (разделы, подзаголовки, списки);
- наличие примеров, скриншотов или схем, если это требуется по заданию.

Юзабилити и дизайн интерфейса (для прототипов и UI-задач)
- удобство навигации и расположения элементов;
- читаемость текста и контрастность цветов;
- согласованность стилей (шрифты, отступы, кнопки);
- учёт принципов доступности.

Логичность и последовательность
- правильная последовательность шагов в пользовательских сценариях и диаграммах деят ельности;
- логическая связь между элементами архитектуры и интерфейса;
- обоснованность выбора инструментов, методов и решений.

Функциональность прототипа
- работоспособность интерактивных элементов (кнопки, формы, переходы);
- реализация основных сценариев использования;
- отсутствие ошибок, блокирующих взаимодействие с прототипом.

Самостоятельность и оригинальность
- независимость выполнения задания;
- оригинальность подхода к решению задачи;
- отсутствие плагиата и прямого копирования решений из открытых источников без перер аботки.

Оформление и представление результатов
- соблюдение требований к формату представления (расширение файла, структура проекта , архивирование);
- аккуратность и чистота оформления диаграмм, кода, документации;
- наличие пояснений, комментариев или легенды, если это необходимо;
- корректность именования файлов и элементов.

Временные показатели
- соблюдение установленных сроков сдачи работы;
- эффективность использования времени на этапах анализа, проектирования и реализации;
- пунктуальность при промежуточных отчётах (если предусмотрены).
- Практическое применение
- реалистичность предложенного решения для реальной ИС;
- возможность использования результатов в профессиональной деятельности (например, в недрение прототипа, использование ТЗ для разработки);
- масштабируемость и гибкость архитектуры/интерфейса для будущих доработок.

Балльная шкала оценивания практических работ:
«Отлично» (5 баллов)
Работа выполнена полностью и без ошибок.
Строго соблюдены все требования к оформлению и стандартам.

Глубокое понимание материала и принципов проектирования.
Оптимальное и эффективное решение задачи.
Работа сдана в срок.
Присутствуют оригинальные подходы или дополнительные улучшения.

«Хорошо» (4 балла)
Работа выполнена полностью, есть 1–2 незначительные ошибки.
В оформлении — мелкие недочёты (опечатки, небольшие нарушения структуры).
Хорошее понимание материала, некоторые аспекты раскрыты не до конца.
Решение эффективное, возможны небольшие улучшения.
Есть элементы самостоятельного подхода.

«Удовлетворительно» (3 балла)
Выполнено 60–80 % задания, есть существенные ошибки.
Оформление требует доработки (нарушена структура, стиль не соблюдается).
Понимание материала частичное, затруднения с объяснением деталей.
Решение неоптимальное (избыточность, дублирование, неудобный интерфейс).
Минимальное использование собственных решений.

«Неудовлетворительно» (2 балла)
Выполнено менее 60 % задания или работа неверна.
Грубые ошибки (нарушена логика, пропущены ключевые функции, диаграммы нечитаемы).
Оформление не соответствует требованиям (хаотичная структура, нечитаемый код).
Отсутствует понимание материала, не может объяснить решение.
Решение неэффективно или отсутствует.
Плагиат или копирование без понимания сути.

МИНИСТЕРСТВО ОБРАЗОВАНИЯ СВЕРДЛОВСКОЙ ОБЛАСТИ

ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ СВЕРДЛОВСКОЙ ОБЛАСТИ
«ЕКАТЕРИНБУРГСКИЙ МОНТАЖНЫЙ КОЛЛЕДЖ»

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
МДК 01.02 Разработка интерфейсов пользователя
для специальности
09.02.09 «Веб-разработка»

Екатеринбург
2026

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Данный фонд оценочных средств подготовлен для сопровождения МДК 01.02 «Разработка интерфейсов пользователя», направленного на формирование практических навыков и компетенций в области создания современных веб-интерфейсов и пользовательских интерфейсов.

Основной целью пособия является предоставление обучающимся полного набора теоретических сведений и практических инструкций, необходимых для освоения ключевых аспектов разработки пользовательских интерфейсов: HTML-вёрстки, CSS-стилизации, создания адаптивных макетов, работы с JavaScript и современными фреймворками.

Пособие охватывает все основные направления разработки интерфейсов, включая базовые технологии вёрстки, создание адаптивных дизайнов, внедрение интерактивных элементов и работу с UI-компонентами, что позволит сформировать у обучающихся комплексные профессиональные компетенции в сфере веб-разработки.

Структура пособия

Пособие построено на основе двух взаимосвязанных частей, обеспечивающих комплексное освоение дисциплины по разработке пользовательских интерфейсов.

Теоретическая часть охватывает фундаментальные основы веб-разработки и проектирования интерфейсов. В ней рассматриваются базовые технологии HTML, CSS и JavaScript, принципы адаптивной вёрстки, современные методологии создания пользовательских интерфейсов. Особое внимание уделяется изучению основ блочной модели, работе с Flexbox и Grid Layout, а также принципам создания интерактивных элементов.

Практическая часть направлена на закрепление теоретических знаний через выполнение практических заданий и разработку реальных проектов. Она включает работу над созданием базовой структуры веб-страниц, разработкой адаптивных макетов, внедрением интерактивных элементов и использованием современных фреймворков.

Завершающий этап обучения предусматривает проведение итоговой аттестации, которая позволит оценить уровень освоения материала и сформированные профессиональные компетенции в области разработки пользовательских интерфейсов.

Учебные цели и задачи

Основные учебные цели включают:

- Формирование системных знаний в области веб-разработки и создания пользовательских интерфейсов
- Освоение современных технологий HTML, CSS и JavaScript для разработки интерфейсов
- Развитие профессиональных компетенций в сфере адаптивной вёрстки и создания интерактивных элементов
- Приобретение практических навыков работы с современными фреймворками и инструментами разработки
- Формирование умений по проектированию и реализации пользовательских интерфейсов
- Развитие способности к самостоятельной разработке веб-приложений

Задачи курса направлены на:

- Освоение теоретических основ веб-технологий и принципов вёрстки
- Изучение современных методологий создания адаптивных интерфейсов
- Освоение техник работы с CSS-фреймворками и библиотеками

- Формирование навыков создания интерактивных элементов с использованием JavaScr ipt
- Развитие умений по разработке адаптивных макетов
- Освоение принципов работы с DOM-деревом
- Приобретение практических навыков создания UI-компонентов
- Формирование компетенций в области разработки интерактивных веб-приложений
- Развитие способности к созданию современных пользовательских интерфейсов

Контроль знаний

- Контроль знаний осуществляется поэтапно:
- Текущие проверки знаний в форме вопросов и опросов.
- Практические работы с последующим обсуждением.
- Итоговая аттестация в форме дифференцированного зачета.

Оценочная деятельность

- Оценка проводится по следующим критериям:
- Качество выполнения практических работ.
- Уровень владения материалами курса.
- Активность и инициативность на занятиях.
- Результативность индивидуальной проектной работы.

В итоге данный фонд оценочных средств обеспечит качественное сопровождение образовательного процесса по разработке пользовательских интерфейсов, поможет освоить современные веб-технологии и методологии создания интерактивных приложений, а также подготовит выпускников к успешной профессиональной деятельности в сфере веб-разработки. Обучающиеся получат необходимые компетенции для создания современных веб-интерфейсов, научатся эффективно применять HTML, CSS и JavaScript, освоят принципы адаптивной вёрстки и работы с современными фреймворками, что позволит им успешно решать профессиональные задачи в условиях реального производства.

Краткое изложение планов хода лекционных занятий:

1. Введение в веб-технологии: HTML, CSS, JavaScript. Основы HTML: разметка контента.
Образовательный результат: понимает базовые принципы веб-разработки; знает структуру HTML-документа; умеет создавать базовую разметку контента; владеет основными HTML-тегами для структурирования информации; способен применять семантические элементы разметки.
2. Основы CSS: стилизация элементов. Позиционирование элементов в CSS.
Образовательный результат: знает основы каскадных таблиц стилей; умеет применять стили к HTML-элементам; владеет техниками позиционирования элементов; понимает принципы работы с блочной моделью; способен создавать простые стили для веб-страниц.
3. Flexbox: гибкий макет. Grid Layout: двумерная вёрстка.
Образовательный результат: понимает принципы работы Flexbox и Grid Layout; умеет создавать гибкие макеты с помощью Flexbox; владеет техниками двумерной вёрстки с Grid; способен эффективно использовать современные методы позиционирования элементов.
4. Адаптивный дизайн: медиа-запросы. Отзывчивые изображения и шрифты.
Образовательный результат: знает принципы адаптивной вёрстки; умеет создавать медиа-запросы; владеет техниками работы с отзывчивыми изображениями; способен настраивать адаптивную типографику; понимает принципы создания мобильных версий сайтов.
5. Основы JavaScript для интерфейсов. Обработка событий в JavaScript.

Образовательный результат: понимает основы JavaScript для веб-интерфейсов; знает принципы работы с DOM; умеет обрабатывать события; владеет базовыми методами манипуляции элементами; способен создавать интерактивные элементы.
6. Динамическое изменение интерфейса с JS.
Образовательный результат: знает методы динамического изменения контента; умеет работать с DOM-элементами; владеет техниками модификации интерфейса; способен создавать интерактивные компоненты с помощью JavaScript.
7. Анимации и переходы в CSS и JS.
Образовательный результат: понимает принципы создания анимаций; знает техники CSS-переходов; умеет создавать анимации с помощью JavaScript; владеет инструментами для реализации интерактивных эффектов.
8. Создание UI-компонентов: кнопки, карточки, модальные окна. Формы и валидация.
Образовательный результат: знает структуру UI-компонентов; умеет создавать базовые элементы интерфейса; владеет техниками валидации форм; способен разрабатывать интерактивные компоненты.
9. CSS-фреймворки: Bootstrap.
Образовательный результат: понимает принципы работы CSS-фреймворков; знает основные компоненты Bootstrap; умеет применять готовые решения; владеет техниками интеграции фреймворка в проект; способен создавать адаптивные макеты с использованием Bootstrap.

Краткое изложение планов хода практических занятий и примерные задания:
1. Создание базовой структуры HTML-страницы.

• Ход занятия: изучение структуры HTML-документа; создание базовой разметки; подключение CSS-файла; настройка метаданных.
• Задание: создать HTML-страницу с заголовком, подзаголовком, абзацем текста и списком; добавить мета-теги и подключить стили.

2. Вёрстка статьи с использованием семантических тегов.

• Ход занятия: работа с семантическими элементами HTML; создание структуры статьи; использование заголовков, абзацев, списков.
• Задание: сверстать статью о веб-разработке с использованием тегов article, section, header, footer, aside.

3. Создание таблицы и формы для сбора данных.

• Ход занятия: создание HTML-таблиц; разработка форм с различными типами полей ввода; валидация данных.
• Задание: создать форму обратной связи с полями для имени, email, сообщения; добавить таблицу с данными пользователей.

4. Стилизация текста и базовых элементов.

• Ход занятия: работа с CSS-свойствами для текста; настройка шрифтов, цветов, отступов.
• Задание: стилизовать текст статьи, добавить декоративные элементы, настроить типографику.

5. Работа с блочной моделью.

• Ход занятия: изучение margin, padding, border; настройка размеров элементов.

• Задание: создать несколько блоков с разными значениями margin и padding, продемонстрировать их взаимодействие.

6. Позиционирование элементов.

• Ход занятия: изучение static, relative, absolute, fixed позиционирования.
• Задание: расположить элементы в разных позициях, создать всплывающую подсказку.

7. Вёрстка навигационного меню с Flexbox.

• Ход занятия: создание гибкого меню; выравнивание элементов; настройка отступов.
• Задание: сверстать горизонтальное и вертикальное меню с помощью Flexbox.

8. Создание галереи карточек с Flexbox.

• Ход занятия: работа с flex-контейнером; настройка flex-свойств.
• Задание: создать адаптивную галерею изображений с карточками товаров.

9. Вёрстка макета страницы с Grid Layout.

• Ход занятия: создание сетки; настройка колонок и строк; размещение элементов.
• Задание: сверстать макет страницы с использованием Grid.

10. Создание сложной сетки с Grid.

• Ход занятия: работа со сложными шаблонами; использование minmax; auto-fit/auto-fill.
• Задание: создать адаптивную сетку для каталога товаров.

11. Адаптация макета под мобильные устройства.

• Ход занятия: создание медиа-запросов; настройка точек разрыва.
• Задание: адаптировать существующий макет под мобильные устройства.

12. Использование отзывчивых изображений.

• Ход занятия: работа с атрибутами srcset и sizes; настройка background-image.
• Задание: добавить отзывчивые изображения разных размеров.

13. Отзывчивая типографика.

• Ход занятия: использование vw единиц; настройка размеров шрифтов.
• Задание: создать масштабируемую типографику для разных устройств.

14. Взаимодействие с DOM.

• Ход занятия: изучение методов querySelector; работа с элементами через JavaScript.
• Задание: написать скрипт для изменения стилей элементов при клике.

15. Обработка кликов и наведения мыши.

• Ход занятия: работа с событиями click, mouseover; добавление обработчиков.

• Задание: создать интерактивные элементы с обработчиками событий.

16. Создание динамического списка задач.

• Ход занятия: работа с массивами; создание элементов через JavaScript.
• Задание: реализовать todo-лист с возможностью добавления и удаления задач.

17. Анимация элементов с CSS.

• Ход занятие: создание анимаций через @keyframes; настройка переходов.
• Задание: анимировать появление элементов, создать слайдер.

18. JavaScript-анимации.

• Ход занятия: работа с requestAnimationFrame; создание плавных анимаций.
• Задание: реализовать анимацию перемещения элементов.

19. Вёрстка страницы с Bootstrap.

• Ход занятия: изучение сетки Bootstrap; работа с компонентами.
• Задание: сверстать адаптивную страницу с использованием готовых компонентов Bootstrap.

Критерии оценивания
Критерии оценивания практических заданий:

Корректность HTML-разметки
- правильность использования семантических тегов;
- валидность кода;
- соблюдение структуры документа;
- отсутствие ошибок в разметке.
- Качество CSS-стилизации
- корректность применения стилей;
- соблюдение принципов каскадности;
- оптимизация кода;
- отсутствие избыточных стилей.

Функциональность интерфейса
- работоспособность всех интерактивных элементов;
- корректная обработка пользовательских действий;
- отсутствие ошибок при взаимодействии;
- выполнение всех заданных функций.
- Адаптивность и отзывчивость
- корректное отображение на разных устройствах;
- работа медиа-запросов;
- масштабируемость элементов;
- сохранение функциональности при изменении размера окна.

Качество JavaScript-кода
- правильность реализации логики;
- эффективность алгоритмов;
- отсутствие ошибок выполнения;
- соблюдение стандартов написания кода.

Юзабилити и дизайн
- удобство использования интерфейса;
- читаемость контента;
- согласованность стилей;
- соблюдение принципов доступности.
- Оптимизация и производительность
- скорость загрузки страницы;
- оптимизация изображений;
- минимизация ресурсов;
- корректная работа анимаций.

Соблюдение стандартов
- соответствие современным веб-стандартам;
- использование актуальных технологий;
- соблюдение рекомендаций по безопасности;
- корректное использование фреймворков.

Самостоятельность выполнения
- оригинальность решения;
- независимость выполнения;
- способность к анализу и исправлению ошибок;
- творческий подход к решению задач.

Оформление проекта
- структурированность кода;
- наличие комментариев;
- корректное именование файлов;
- соблюдение требований к структуре проекта.

Балльная шкала оценивания практических работ:
«Отлично» (5 баллов)
- Работа выполнена полностью, без ошибок в HTML, CSS и JavaScript.
- Код валидный, структурированный и оптимизированный.
- Полностью соблюдены все требования к адаптивности и кроссбраузерности.
- Интерфейс интуитивно понятен и соответствует современным стандартам UX.
- Работа сдана в срок.
- Документация полная и корректная.

«Хорошо» (4 балла)
- Работа выполнена полностью, допущены 1–2 незначительные ошибки в коде.
- Есть мелкие недочёты в стилизации или адаптивности.
- Оформление соответствует требованиям, но есть небольшие нарушения стиля.
- Интерфейс работает корректно, но требует небольших улучшений.
- Решение эффективное, но возможны оптимизации.

«Удовлетворительно» (3 балла)
- Выполнено 60–80% задания, присутствуют существенные ошибки в коде.
- Нарушена структура проекта или отсутствует часть функциональности.
- Адаптивность работает частично или некорректно.
- Интерфейс неудобен в использовании.
- Отсутствуют необходимые комментарии в коде.
- Частично нарушены требования к оформлению.

«Неудовлетворительно» (2 балла)

- Выполнено менее 60% задания или работа содержит критические ошибки.
- Нарушена базовая функциональность или логика работы.
- Отсутствует адаптивность или она реализована неверно.
- Интерфейс неработоспособен или не соответствует заданию.
- Грубые нарушения в структуре кода.
- Отсутствие необходимой документации.
- Обнаружен плагиат или копирование без понимания сути.

МИНИСТЕРСТВО ОБРАЗОВАНИЯ СВЕРДЛОВСКОЙ ОБЛАСТИ

ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ СВЕРДЛОВСКОЙ ОБЛАСТИ
«ЕКАТЕРИНБУРГСКИЙ МОНТАЖНЫЙ КОЛЛЕДЖ»

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
МДК 01.03 Тестирование информационных ресурсов

09.02.09 «Веб-разработка»

Екатеринбург
2026

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Данный фонд оценочных средств подготовлен для сопровождения МДК 01.03 Тестирование информационных ресурсов, направленного на формирование профессиональных компетенций в области полного цикла тестирования приложений.

Основная цель пособия — обеспечение качественного освоения студентами теоретических знаний и практических навыков в сфере тестирования программных продуктов, включая методы анализа, виды, методики, а также особенности организации процессов тестирования и фиксации результатов.

Структура пособия

Теоретическая часть

Посвящена изучению ключевых понятий и методов тестирования информационных систем, типам ошибок, критериям и принципам тестирования, видам (в том числе автоматизированным) и этапам тестирования, а также основам функционального и нефункционального тестирования. Цель — формирование основательного понимания процессов тестирования и подготовки к выполнению практических заданий.

Практическая часть

Направлена на закрепление теоретических знаний через выполнение лабораторных работ, практических заданий и проектов. Каждое практическое задание сопровождается детальным описанием плана выполнения, примерами и образцами решений, а также рекомендациями по самостоятельной проверке и оценке результата. Практика позволяет формировать навыки проведения тестирования разных видов, анализа результатов, документирования ошибок и работы с инструментами автоматизации тестирования.

Учебные цели и задачи

Основные учебные цели включают:

- освоение системного подхода к организации процессов тестирования;
- изучение видов тестирования и методик их применения на практике;
- развитие навыков выявления и анализа ошибок программных модулей;
- формирование умений документирования и отчётности по результатам тестирования;
- освоение инструментов и технологий автоматизации тестирования.

Задачи курса:

- изучить теоретические основы и методологию тестирования;
- овладеть навыками задания критериев, разработки и проведения тестов;
- осваивать проведение статического и динамического тестирования;
- совершенствовать умения работы с системами отслеживания ошибок и управления тестированием;
- формировать способности к анализу качества программных продуктов с применением автоматизированных средств.

Контроль знаний

Контроль знаний осуществляется поэтапно:

- Текущие проверки знаний в форме теоретических вопросов и опросов;
- Практические работы с последующим обсуждением и анализом результатов;
- Итоговая аттестация в форме дифференцированного зачета

Оценочная деятельность

Оценка проводится по следующим критериям:

- Качество выполнения практических работ по всем уровням тестирования;
- Уровень владения материалами курса и техниками тест-дизайна;

- Результативность работ по созданию тестовой документации;
- Качество анализа дефектов и формулировки рекомендаций по улучшению.

В итоге данный фонд оценочных средств обеспечит качественное сопровождение образовательного процесса по тестированию информационных систем, поможет освоить полный цикл тестирования от теории до автоматизации, а также подготовит выпускников к успешной профессиональной деятельности в сфере QA и автоматизации тестирования. Обучающиеся получат необходимые компетенции для проведения всех видов тестирования, научатся эффективно применять современные техники тест-дизайна и подходы автоматизации, освоят принципы создания профессиональной тестовой документации, что позволит им успешно решить профессиональные задачи в условиях реального производства.

Краткое изложение планов хода лекционных занятий:

1. Введение в теорию тестирования

Определение тестирования и QA. Различия verification/validation. Принципы тестирования (раннее тестирование, кластеризация ошибок, парадокс пестицидов). Роли участников процесса. Экономический эффект тестирования. Модели STLC Unit→Integration→System→Acceptance. Функциональное/нефункциональное тестирование. Black/White/Gray Box

Образовательный результат: Понимает роль тестирования в жизненном цикле разработки ПО; знает основные принципы и цели QA; различает тестирование, контроль качества и отладку; владеет базовой терминологией QA. Пирамида тестирования; понимает связь тестирования с SDLC, различает виды и методы тестирования; определяет объекты тестирования на каждом этапе.

2. Тестовая документация в QA

Образовательный результат: Знает структуру Test Plan (IEEE 829); умеет создавать чек-листы, тест-кейсы, матрицы прослеживаемости; понимает назначение каждого документа; приводит практические примеры.

3. Техники тест-дизайна

Образовательный результат: Владеет техниками тест-дизайна (EP, BVA, Decision Table, Pairwise, Cause/Effect, Error Guessing, Domain Testing); умеет оптимизировать тестовые наборы; знает критерии покрытия кода.

4. Понятие программной ошибки. Жизненный цикл бага

Образовательный результат: Понимает причины и классификацию ошибок ПО; знает полный lifecycle бага; умеет анализировать root cause; владеет структурой баг-репорта; различает severity/priority.

5. Модульное тестирование

TDD цикл. Тестовое окружение. Тестовые данные (fixtures). Принципы: изолированность, быстрота, повторяемость. Coverage метрики. AAA паттерн (Arrange-Act-Assert).

Образовательный результат: Понимает принципы и цели unit-тестирования; знает процесс TDD (Red-Green-Refactor); определяет критерии хорошего unit-теста

6. Интеграционное и API-тестирование

Типы интеграционных тестов (Big Bang, Top-Down, Bottom-Up, Sandwich). Отличия unit/integration тестирования. API-тестирование: REST принципы, HTTP статусы (200/201/401/422/429), заголовки. Сценарии: happy path, negative (auth/valid), edge cases

(pagination, rate limiting). Фикстуры БД (Testcontainers, in-memory). Мокинг (WireMock, внешние сервисы). Примеры реальных API дефектов.
Образовательный результат: Понимает типы интеграционных тестов и их применение; различает unit и integration тестирование; знает ключевые сценарии API-тестирования (happy path, ошибки авторизации/валидации, пагинация); определяет необходимость фикстур БД и мокинга внешних зависимостей.

7. Автоматизация тестирования

Автоматизация тестирования на разных этапах тестирования (smoke, regression, data validation). Автоматизация приемочного (ATDD) и регрессионного тестирования. Сравнительный анализ инструментов автоматизации.
Основные инструменты автоматизации тестирования: Регрессионные тест-сюиты. Матрица ROI автоматизации. Сравнение подходов: функциональная декомпозиция Data Driven и Keyword-Driven и BDD. Выбор инструментов по задачам.
Образовательный результат: Знает критерии автоматизации (стабильность, повторяемость, ROI); различает приемочное/регрессионное тестирование; определяет подходящие сценарии для автоматизации; проводит сравнительный анализ подходов к автоматизации.

8. Автоматизация нефункционального тестирования

Автоматизация тестирования при использовании различных интерфейсов (API, CLI, GUI и другие). Нефункциональное тестирование: тестирование производительности, безопасности, удобства использования.
Образовательный результат: Различает подходы автоматизации для API/CLI/GUI; понимает принципы нефункционального тестирования; определяет метрики производительности, безопасности, usability; знает особенности тестирования различных интерфейсов.

9. UI автоматизация и Selenium принципы

Selenium: поддерживаемые платформы. Компоненты Selenium Webdriver. Взаимодействие с браузером и UI. Тестовое окружение (headless, resolutions,pages). Page Object Model принципы при использовании Selenium. Проблемы UI автоматизации (timing, flakiness), anti-flakiness техники.Браузерная архитектура (WebDriver protocol). Кросс-браузерное тестирование. Стратегии локаторов (ID>CSS>XPath).
Образовательный результат: Понимает принципы UI автоматизации; знает архитектуру браузерной автоматизации; определяет поддерживаемые платформы и компоненты; различает тестовое и production окружение для UI-тестов.

10. Документирование процесса тестирования и контроль качества

Документирование процесса тестирования Контроль качества тестов, системы полноты, взаимодействие автотестов с тестируемой системой. Покрытие кода тестами Контроль качества тестов (maintainability, stability, coverage). Тестовая отчетность (Allure, HTML). Матрица traceability (requirements→tests→bugs). Test Impact Analysis. Self-healing тесты. CI/CD интеграция результатов тестирования.
Образовательный результат: Знает системы контроля качества тестов; понимает метрики полноты покрытия; определяет критерии качества автотестов; умеет анализировать взаимодействие тестов с системой и отчеты покрытия кода.

Краткое изложение планов хода практических занятий и примерные задания:

1. Формирование тест-кейсов и чек-листа.

- Ход занятия: Анализ реального продукта; определение объектов тестирования; создание чек-листа листа и базовых тест-кейсов на примере реального продукта для покрытия основных сценариев; разработка базовых тест-кейсов; проверка полноты покрытия.

- Задание: Для веб-формы регистрации создать чек-лист (20+ пунктов) и 10 базовых тест-кейсов по шаблону (ID, шаги, ожидаемый результат).

2. Создание библиотеки классов с применением Test Driven Development.
- Ход занятия: Изучение TDD цикла (Red-Green-Refactor); написание failing теста; минимальная реализация; рефакторинг; проверка стабильности.
- Задание: написать тест → реализовать → рефакторить до 100% покрытия.

3. Проектирование тест-кейсов с параметризацией при unit-тестировании
- Ход занятия: Анализ библиотеки классов; определение тестовой параметризации; создание наборов данных; анализ результатов.
- Задание: Для класса параметризовать тесты: валидные/невалидные, пустые поля (10+ комбинаций).

4. Обработка ошибок и исключений в unit-тестах.
- Ход занятия: тестирование исключений; граничные значения; документирование результатов; анализ покрытия ветвлений; документирование результатов тестирования.
- Задание: Дополнить тесты проверкой исключений и граничных длин строк; создать отчет о покрытии.

5. Настройка тестового окружения для API.
- Ход занятия: Создание базовых интеграционных тестов для проверки взаимодействия модулей (бэкенд+БД). Настройка фикстур. Тестовые сценарии и запуск автотестов в текстовом файле README.md. Инструменты анализа покрытия кода.
- Задание: Настроить БД и создать 5 интеграционных теста для API (CRUD); оформить README.md с инструкцией запуска.

6. Сценарии API-тестирования с мокингом
- Ход занятия: Тестирование сценариев: happy path (GET/POST 200), ошибки (401 Unauthorized, 422 ValidationError), пагинация (page/limit). Мокинг внешних зависимостей. Документирование тест-кейсов с 80%+ покрытием
- Задание: Создать тест-сюит для API эндпоинта с GET/POST, ошибками авторизации, пагинацией; мокинг внешнего сервиса; отчет покрытия.

7. Автоматизация: функциональная декомпозиция
- Ход занятия: Разбиение автотестов на переиспользуемые функции с параметризацией. азбиение автотестов на функции; рефакторинг дублирования кода.
- Задание: Разработать для формы 3 тест-кейса, используя общие функции (login, logout, error_check).

8. Автоматизация: Data Driven подход
- Ход занятия: Параметризация тестов данными из CSV/JSON (тестовые данные отделены от кода). Создание CSV/JSON с тестовыми данными (массовое тестирование); настройка параметризации; разделение логики и данных.
- Задание: Сгенерировать CSV-файлы с 200 комбинациями тестами; написать data-driven тест для проверки валидации формы одним тестом.

9. Автоматизация тестирования через ключевые слова (Keyword-Driven Testing).
- Ход занятия: Создание таблицы ключевых слов; разработка библиотеки функций; выполнение тестов из таблицы; анализ переиспользования. Автоматизация тестирования через ключевые слова для GUI/API. Таблицы ключевых слов + библиотека функций.

- Задание: Сгенерировать данные для таблицы Excel с 15 ключевыми словами (click, input, verify) и библиотеку функций; протестировать форму регистрации.

10. Автоматизация: BDD для приемочного тестирования
- Ход занятия: Given-When-Then сценарии. Связь бизнес-требований с автотестами регрессии. Написание файлов (Given-When-Then); реализация шагов; связь с бизнес-требованиями; запуск сценариев.
- Задание: Создать файл с 3 сценариями регистрации (успех/ошибка email/ошибка пароля); реализовать шаги Given/When/Then.

11. Нефункциональное тестирование и документирование дефектов
- Ход занятия: Тестирование безопасности (аутентификация) и отказоустойчивости, удобства использования, точек отказа системы. Написание баг-репортов по стандарту: шаги воспроизведения, ожидаемый/фактический результат, скриншоты, приоритет. Тестирование usability/performance/security; написание баг-репортов; приоритизация дефектов.
- Задание: Протестировать веб-форму на usability , найти 5 дефектов, оформить баг-репорты по стандарту IEEE 829.

12. Автоматизация UI-тестирования с Selenium
- Ход занятия: Установка Selenium WebDriver, поддерживаемые платформы. Компоненты WebDriver, взаимодействие с браузером/UI, тестовое окружение (Page Object Model) Настройка параллельное выполнение Selenium тестов в Chrome/Firefox/Edge. Проверка адаптивности UI (кросс-браузерное тестирование: 1920x1080, 375x667) и поведение форм.
- Задание: Автоматизировать тест логина в Chrome/Firefox на desktop/mobile разрешениях; настроить параллельный запуск.

13. Создание устойчивых UI-тестов с Selenium для регрессии.
- Ход занятия: Обработка динамических элементов, ожидания (WebDriverWait), работа с динамическими элементами (iframe/popups/dropdown); robust локаторы.
- Задание: Автоматизировать тест с модальным окном, dropdown и iframe; использовать WebDriverWait для всех взаимодействий.

14. Сквозные тесты (End-to-End, E2E) с Selenium
- Ход занятия: Создание E2E тестовых сценариев, охватывающих взаимодействие всех компонентов системы — UI → API → БД → обратная UI-верификация для контроля целостности критических бизнес-процессов; синхронизация слоев, cleanup данных; верификация бизнес-логики.
- Задание: Создать E2E тест: UI регистрация → API проверка БД → UI dashboard с данными пользователя.

15. Настройка сборки отчетов при тестировании
- Ход занятия: Настройка сборки отчетов при тестировании со скриншотами ошибок, тестовыми данными и результатом. Аанализ ошибок; метрики стабильности.
- Задание: Настроить создание отчетов для всех автотестов с скриншотами ошибок, тестовыми данными и историей запусков.

Критерии оценивания практических заданий по тестированию:
Корректность выполнения задания
- правильность применения техник тест-дизайна (EP, BVA, Pairwise, Decision Table) согласно специфике объекта тестирования;

- соответствие разработанных тест-кейсов и чек-листов заявленным требованиям и сценариям;
- отсутствие пропусков критических тест-сценариев .

Качество тест-дизайна

- корректность применения выбранных техник тест-дизайна с обоснованием выбора;
- оптимальность тестового набора (сокращение количества ТС без потери покрытия);
- полнота покрытия классов эквивалентности, граничных значений и комбинаций параметров;
- логическая связность тест-кейсов и их соответствие бизнес-логике.

Полнота выполнения

- выполнение всех пунктов задания (чек-листы, тест-кейсы, автотесты, документация);
- наличие тестов всех требуемых уровней (unit/integration/system/E2E по заданию);
- завершённость работы: от анализа требований до финальных отчетов и cleanup.

Соответствие стандартам тестирования

- соблюдение принципов создания тестовой документации (IEEE 829, ISTQB);
- использование общепринятых шаблонов тест-кейсов, баг-репортов, чек-листов;
- правильная структура тест-плана и матрицы прослеживаемости;
- соответствие жизненному циклу дефектов (New→Fixed→Verified→Closed).

Качество тестовой документации

- чёткость и однозначность формулировок в тест-кейсах и баг-репортах;
- структурированность документов ;
- наличие метрик покрытия кода (≥80% для автотестов), скриншотов ошибок;
- полнота README.md (запуск, окружение, инструкции).

Стабильность и надежность автотестов

- стабильность выполнения автотестов (≤5%);
- корректная обработка исключений и таймаутов;
- устойчивость к изменениям системы.

Качество покрытия и анализ

- достижение целевых метрик покрытия кода (≥80%);
- полнота покрытия бизнес-сценариев тестовой матрицей;
- корректный анализ результатов тестирования и дефектов;
- обоснованность рекомендаций по рефакторингу и улучшению.

Архитектура автотестов

- применение изученных подходов автоматизации (POM, Data Driven, BDD);
- функциональная декомпозиция и переиспользование кода тестов;
- правильная настройка тестового окружения ;
- разделение тестовых данных от логики тестов.

Логичность и последовательность

- правильная последовательность шагов в тест-кейсах и автотестах;
- логическая связь между уровнями тестирования (unit→integration→E2E);
- обоснованность выбора техник тест-дизайна и подходов автоматизации;
- корректная верификация ожидаемых результатов.

Функциональность автотестов

- работоспособность всех автотестов на всех уровнях;
- корректная реализация всех сценариев;
- отсутствие ложных срабатываний;
- успешная работа в CI/CD окружении.

Самостоятельность и оригинальность

- независимое выполнение задания без прямого копирования решений;

- оригинальность подхода к автоматизации и тест-дизайну;
- адаптация изученных методик под конкретную задачу;
- наличие собственных дополнений к базовому решению.

Оформление и представление результатов

- соблюдение требований к структуре проекта;
- аккуратность оформления отчетов ;
- наличие комментариев в коде автотестов и пояснений;
- корректное именование файлов, тестов, веток Git.

Временные показатели

- соблюдение установленных сроков сдачи лабораторных работ;
- эффективность использования времени на анализ и автоматизацию;
- пунктуальность при демонстрации промежуточных результатов.

Практическое применение

- применимость тест-кейсов и автотестов в реальных проектах;
- возможность интеграции в существующую CI/CD систему;
- масштабируемость тестовой архитектуры для будущих доработок;
- реальная ценность выявленных дефектов для разработки.

Балльная шкала оценивания практических работ:

Отлично (5) — выполнены все критерии, покрытие ≥90%, оригинальные решения
Хорошо (4) — выполнены основные критерии, покрытие 80-89%, стандартные подходы
Удовлетворительно (3) — базовое выполнение, покрытие 60-79%, без автотестов
Неудовлетворительно (2) — критические недочеты, покрытие <60%, незавершенность

Примерное задание для проведения дифференцированного зачета.

Задание 1 «Тестирование UI»

Необходимо создать проект UI-автотестов для страницы https://practice-automation.com/ .
Форма: https://practice-automation.com/form-fields/
События : https://practice-automation.com/click-events/
Модальные окна : https://practice-automation.com/popups/
Проанализировать функционал страниц .
На странице с формой в поле Message заполнить списком из раздела Automation Tools, который нужно получить средствами Selenium
Требования к реализации:

- полнота README.md (запуск, окружение, инструкции).
- правильная настройка тестового окружения (для обеспечения читаемости, переиспользования и поддержки кода требуется применить структурированный подход к автоматизации (рекомендуется паттерн Page Object Model);
- корректность применения выбранных техник тест-дизайна с обоснованием выбора;
- оптимальность тестового набора (сокращение количества ТС без потери покрытия) и полнота покрытия классов эквивалентности, граничных значений и комбинаций параметров;
- логическая связность тест-кейсов и их соответствие бизнес-логике.
- использование общепринятых шаблонов тест-кейсов, баг-репортов, чек-листов;
- корректная обработка исключений и таймаутов;
- устойчивость к изменениям системы.
- разделение тестовых данных от логики тестов.
- правильная последовательность шагов в тест-кейсах и автотестах,

- осмысленные названия тестов и шагов;
- корректное именование файлов, веток Git.

Задание 2 «Формирование документации к API»
На основании предложенной API создать подробную документацию.
Описание API
1. Описание API
Назначение
1.1. Общие принципы
Формат обмена: JSON (Content-Type: application/json)
Версионирование: /api/v1/ (совместимость с v1 гарантирована)
Аутентификация: JWT токены в заголовке Authorization: Bearer <token>

1.2. Группировка эндпоинтов
Аутентификация
POST /api/v1/auth/login
POST /api/v1/auth/refresh

Пользователи
GET /api/v1/users # Список пользователей
GET /api/v1/users/{id} # Пользователь по ID
POST /api/v1/users # Создание
PUT /api/v1/users/{id} # Полное обновление
PATCH /api/v1/users/{id} # Частичное обновление
DELETE /api/v1/users/{id} # Удаление

1.3. Спецификация эндпоинтов
POST /api/v1/users -Создание нового пользователя
Параметры запроса:

Поле	Тип	Обязательное
email	string	Да
name	string	Да
password	string	Да

Пример запроса:
POST /api/v1/users
```
{
  "email": "user@example.com",
  "name": "Иван Иванов",
  "password": "SecurePass123!"
}
```

Коды ответа:

201	Created: {"id": "uuid", "email":...}
400	Bad Request: неверные данные
409	Conflict: email уже существует
422	Validation Error: формат email
401	Unauthorized: требуется токен

3.1.4. Структуры данных (JSON Schema)
Users (ответ):

```
{
  "id": "uuid",
  "email": "string",
  "name": "string",
  "created_at": "ISO8601",
  "updated_at": "ISO8601"
}

Paginated Response:
{
  "data": {
    "items": [User],
    "total": "integer",
    "page": "integer",
    "size": "integer",
    "pages": "integer"
  }
}

Error Response:
{
  "error": {
    "code": "VALIDATION_ERROR",
    "message": "string",
    "details": { "field": ["error1", "error2"] },
    "timestamp": "ISO8601"
  }
}
```

1.5. Пагинация и фильтрация
Параметры (query string):
```
page=1        # Номер страницы (1-based)
size=10       # Размер страницы (1-100)
sort=-created_at  # Сортировка (поле или -поле)
filter[status]=active  # Фильтр по полю
search=Иван    # Полнотекстовый поиск
```

Пример: GET /api/v1/users?page=1&size=10&sort=-created_at&filter[role]=user

1.6. Безопасность API (Эндпоинты без авторизации):
- POST /api/v1/auth/login
- POST /api/v1/auth/refresh
- GET /api/v1/health