



МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«ЮЖНО-УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ГУМАНИТАРНО-ПЕДАГОГИЧЕСКИЙ
УНИВЕРСИТЕТ»
(ФГБОУ ВО «ЮУрГГПУ»)

ПРОФЕССИОНАЛЬНО-ПЕДАГОГИЧЕСКИЙ ИНСТИТУТ
КАФЕДРА АВТОМОБИЛЬНОГО ТРАНСПОРТА, ИНФОРМАЦИОННЫХ ТЕХНОЛО-
ГИЙ И МЕТОДИКИ ОБУЧЕНИЯ ТЕХНИЧЕСКИМ ДИСЦИПЛИНАМ

**Разработка электронного учебно-методического пособия по разделу
междисциплинарного курса «Проектирование и разработка веб-
приложений» как средства формирования профессиональных
компетенций студентов профессиональной образовательной
организации**

**Выпускная квалификационная работа по направлению
44.03.04 Профессиональное обучение (по отраслям)
Направленность программы бакалавриата
«Информатика и вычислительная техника»
Форма обучения заочная**

Проверка на объем заимствований:
85,15 % авторского текста

Работа рекомендована к защите
«15» 09 2025 г.
Зав. кафедрой АТИТ и МОТД
Руднев В.В.

Выполнил:
Студент(ка) группы ЗФ-509-079-5-1
Галендухин Илья Сергеевич

Научный руководитель:
к.т.н., зав. кафедрой АТИТ и МОТД
Руднев В.В.

Челябинск
2025

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
ГЛАВА 1. ТЕОРЕТИЧЕСКИЕ ОСНОВЫ РАЗРАБОТКИ ЭЛЕКТРОННОГО УЧЕБНО-МЕТОДИЧЕСКОГО ПОСОБИЯ.....	6
1.1 Понятие и место электронного учебно-методического пособия в СПО. 6	
1.2 Требования к содержанию и оформлению электронных учебных пособий.....	9
1.3 Междисциплинарный курс «Проектирование и разработка веб- приложений»: цели, задачи, компетенции и методические подходы к формированию профессиональных компетенций студентов через электронное пособие.....	13
Выводы по главе 1.....	18
ГЛАВА 2. ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА ЭЛЕКТРОННОГО УЧЕБНО-МЕТОДИЧЕСКОГО ПОСОБИЯ.....	22
2.1 Анализ учебного раздела «Проектирование и разработка веб-приложений»: содержание, ключевые темы, требования.....	22
2.2 Структура и навигация пособия: модульный подход, интерфейс, сценарии использования.....	26
2.3 Выбор технических средств и инструментов разработки: критерии и обоснование	30
2.4 Методика апробации и оценка эффективности использования пособия в учебном процессе	34
Выводы по Главе 2	38
ЗАКЛЮЧЕНИЕ	39
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	43
ПРИЛОЖЕНИЯ.....	46

ВВЕДЕНИЕ

Актуальность исследования. Современные тенденции в образовании и стремительное развитие информационных технологий диктуют необходимость внедрения в учебный процесс цифровых средств, максимально приближающих студента к реальной профессиональной деятельности. В условиях реализации Федеральных государственных образовательных стандартов СПО (ФГОС СПО) проектный подход, опирающийся на использование электронных учебно-методических пособий, становится ключевым фактором формирования у будущих специалистов профессиональных компетенций.

Междисциплинарный курс «Проектирование и разработка веб-приложений» требует наличия удобного, интерактивного и содержательно насыщенного пособия, которое обеспечит целостное освоение теоретических основ и практических навыков веб-разработки. Электронный формат учебно-методического пособия позволяет интегрировать мультимедийные элементы, автоматизированные контролирующие средства и интерактивные задания, что способствует развитию самостоятельности, творческого мышления и исследовательских компетенций студентов профессиональной образовательной организации.

Цель исследования: теоретическое обоснование и разработка электронного учебно-методического пособия по разделу междисциплинарного курса «Проектирование и разработка веб-приложений» для эффективного формирования профессиональных компетенций студентов СПО.

Объект исследования: профессионально образовательная организация ГБПОУ «Южно-Уральский государственный колледж».

Предмет исследования: электронное учебно-методическое пособие по разделу междисциплинарного курса «Проектирование и разработка веб-

приложений» как средство формирования профессиональных компетенций студентов.

Задачи исследования:

1. Рассмотреть понятие, типологию и функции электронных учебно-методических пособий в системе СПО.
2. Изучить требования к содержанию, структуре и интерактивным компонентам электронных пособий для IT-дисциплин.
3. Проанализировать учебные цели и профессиональные компетенции, заданные ФГОС СПО для курса «Проектирование и разработка веб-приложений».
4. Определить методические подходы к проектированию электронного пособия на основе компетентностного и проектного подходов.
5. Спроектировать структуру, навигацию и функциональные модули электронного пособия с учётом особенностей веб-разработки.
6. Реализовать пилотную версию электронного пособия, включающую мультимедийные и интерактивные элементы.
7. Провести апробацию пособия в учебном процессе и оценить его влияние на развитие профессиональных компетенций студентов.

В исследовании использованы работы отечественных и зарубежных учёных в области электронного обучения, методики проектного обучения и компетентностного подхода. Основоположниками проектной методики считаются Дж. Дьюи и У. Килпатрик, педагогические идеи которых развивали П. П. Блонский и С. Т. Шацкий. Современные теории мультимедиа-обучения сформулированы Р. Мейером и Р. Кларком, а принципы компетентностного подхода — в документах ФГОС СПО.

Практическая значимость исследования. Результатом работы станет готовый электронный учебно-методический комплекс по разделу курса «Проектирование и разработка веб-приложений» и методические рекомендации по его внедрению, что позволит повысить качество

подготовки студентов и ускорить формирование у них ключевых профессиональных компетенций.

Методы исследования.

Теоретические: анализ и синтез педагогической литературы, сравнительно-сопоставительный метод, моделирование.

Эмпирические: анкетирование студентов и преподавателей, экспертная оценка, педагогическая апробация.

База исследования: ГБПОУ «Южно-Уральский государственный технический колледж».

В первой главе рассмотрены теоретические основы электронных учебно-методических пособий и требования к междисциплинарному курсу «Проектирование и разработка веб-приложений».

Во второй главе описан процесс проектирования, разработки и апробации электронного пособия, приведены результаты его внедрения и оценки эффективности.

Структура исследования включает: введение, две главы, выводы по главам, заключение, список используемой литературы, приложения.

ГЛАВА 1. ТЕОРЕТИЧЕСКИЕ ОСНОВЫ РАЗРАБОТКИ ЭЛЕКТРОННОГО УЧЕБНО-МЕТОДИЧЕСКОГО ПОСОБИЯ

1.1 Понятие и место электронного учебно-методического пособия в СПО

В последние десятилетия цифровая трансформация охватывает все уровни образования, включая систему среднего профессионального образования (СПО). Если раньше печатные конспекты и методички выполняли вспомогательную роль, то сегодня электронные образовательные ресурсы становятся ядром учебного процесса, определяя формат, темп и глубину погружения студента в дисциплину. В контексте курса «Проектирование и разработка веб-приложений» эта тенденция особенно заметна: отрасль меняется буквально ежемесячно, и классические издания не успевают отражать новые фреймворки, паттерны и инструменты.

Под электронным учебно-методическим пособием (ЭУМП) понимается структурированный цифровой ресурс, который содержит теоретический материал, практические задания, контролирующие мероприятия и методические рекомендации, реализованные средствами гипертекста, мультимедиа и интерактивной логики. В отличие от электронного учебника, ориентированного преимущественно на изложение содержания, ЭУМП обеспечивает полноценный образовательный сценарий: предлагает маршруты самообучения, шаблоны проектной документации, автоматические тесты и каналы обратной связи. Благодаря интеграции в систему управления обучением (LMS) ресурс фиксирует индивидуальные траектории и позволяет преподавателю точно поддерживать студента.

По степени интерактивности современные ЭУМП делятся на статические, условно интерактивные и адаптивные. Первые представляют собой набор гипертекстовых страниц и медиаконтента без учёта действий пользователя. Вторые дополняются тренажёрами и автотестами, но не

меняют структуру в зависимости от успехов обучающегося. Адаптивные решения строят персональные ветки прохождения, опираясь на Learning Analytics. По предметной области выделяют общепрофессиональные, междисциплинарные и специализированные пособия; наш курс относится к междисциплинарным, объединяя основы проектирования, программирования и UX-дизайна. Нормативно-правовую основу разработки регламентируют Закон «Об образовании», ФГОС СПО по укрупнённой группе 09.00., а также методические письма Минпросвещения, настаивающие на доступности и инклюзии. Документы подчёркивают, что цифровой ресурс обязан формировать заявленные в стандарте общие и профессиональные компетенции, обеспечивать контроль результатов и быть эргономичным для студентов с различными возможностями здоровья.

Дидактический потенциал ЭУМП определяется комплексом функций: информационной, навигационной, практической, контролирующей, мотивационной и коммуникационной. Информационная функция реализует подачу знаний в мультимодальном формате; навигационная — организует контент в логически связанные модули; практическая — переводит теорию в проектные задания, максимально приближённые к профессиональной среде. Контролирующий слой представлен автотестами, чек-листами и интерактивными опросами, а мотивационный — элементами геймификации, визуализацией прогресса и бейджами за ключевые достижения. Коммуникационная функция закрепляется интеграцией с форумами и чат-ботами. При проектировании пособия соблюдаются принципы Р. Мейера: мультимодальности, ограниченной когнитивной нагрузки, сегментации и личностной адресности; дополняет их принцип «learning by doing», требующий немедленного применения каждой новой концепции в среде программирования или конструкторе макетов.

Встраивание ЭУМП в образовательный процесс происходит на всех этапах. До занятия ресурс поддерживает модель перевёрнутого класса: студент изучает теорию, выполняет базовые викторины и приходит на

практику подготовленным. В аудитории пособие превращается в навигатор проектной работы: через сценарии «шаг-за-шагом» группа создаёт прототип интерфейса, проектирует базу данных или настраивает CI/CD. После занятия платформа предоставляет расширенные кейсы, peer-review и анализ кода, что способствует закреплению материала и развитию критического мышления. Компетентностная ориентация выражена в прямой привязке каждого модуля к профессиональным компетенциям ФГОС (ПК 1.1, ПК 1.2 и др.). В начале раздела формируются «маяки успеха» — конкретные измеримые результаты, а прогресс визуализируется на дэшборде студента.

Техническая архитектура пособия строится по модульному принципу. Контент хранится в Markdown-файлах под управлением Git, что обеспечивает коллективную разработку и версионирование. Запуск кода осуществляется в браузерной песочнице на базе Docker-контейнеров, в которую автоматически подтягиваются актуальные версии Node.js или Python. Автотесты пишутся на Jest или PyTest и запускаются внутри того же контейнера, а результаты через webhook передаются в LMS. Такой подход исключает необходимость локальной установки ПО, ускоряет проверку и устраняет проблему «работает только у меня».

Разработка качественного ЭУМП немислива без учёта инклюзивности и аналитики. Рекомендации WCAG 2.1 обеспечивают доступность интерфейса для студентов с ОВЗ: контрастная цветовая палитра, навигация с клавиатуры, альтернативные описания медиаконтента. Learning Analytics собирает метрики времени решения задач, числа попыток и активности в обсуждениях, формируя персональные рекомендации.

На горизонте ближайших лет к пособиям подключаются инструменты искусственного интеллекта: чат-бот-ассистент, который подсказывает документацию к API, генеративные системы проверки кода и адаптивные симуляторы, изменяющие сложность задач в реальном времени. Внедрение ИИ требует прозрачности алгоритмов и согласия пользователя на обработку данных.

Контроль качества ресурса осуществляется многоуровнево: экспертное рецензирование содержания, техническое тестирование кросс-браузерной совместимости, педагогическая экспертиза дидактической целесообразности интерактивов и пилотное внедрение в учебной группе с последующим сбором обратной связи. Экономическая эффективность выражается в снижении затрат на печать, автоматизации проверки и росте успеваемости, что сокращает расходы колледжа на повторное обучение. Кроме того, хорошо спроектированное ЭУМП может быть тиражировано или лицензировано как открытый образовательный ресурс, усиливая репутацию образовательной организации и привлекая абитуриентов.

Таким образом, электронное учебно-методическое пособие выступает ключевым инструментом цифровой образовательной среды СПО. Оно объединяет актуальный контент, практико-ориентированные задания, средства аналитики и каналы коммуникации в единую экосистему, ориентированную на формирование востребованных на рынке труда компетенций веб-разработчика. Исторический путь от CD-дисков к облачным адаптивным платформам и международная практика OER свидетельствуют о зрелости формата, а нормативные документы — о его обязательности. Грамотная реализация ЭУМП по курсу «Проектирование и разработка веб-приложений» позволяет колледжу опережать изменения отрасли и выпускать специалистов, готовых к профессиональным вызовам цифровой экономики.

1.2 Требования к содержанию и оформлению электронных учебных пособий

Разработка электронного учебно-методического пособия (ЭУМП) в системе среднего профессионального образования требует строгого соблюдения как нормативных, так и педагогических требований. Цель этих требований — гарантировать, что ресурс не только представит актуальный

и достоверный материал, но и обеспечит комфортное, эффективное и инклюзивное обучение. Современная дидактика рассматривает ЭУМП в качестве ядра цифровой образовательной среды, где гармонично сочетаются контент, интерактивные практики и инструменты мониторинга успеваемости студентов.

Нормативно-правовую основу формируют Федеральный закон «Об образовании в Российской Федерации» № 273-ФЗ, актуальные ФГОС СПО по укрупнённым группам специальностей, государственный стандарт ГОСТ Р 7.0.97-2016 «Издания электронные», а также методические письма Минпросвещения России по внедрению электронного обучения и дистанционных технологий. Разработчик обязан учитывать требования к структуре и техническому формату электронного издания, а также регламенты хранения персональных данных, утверждённые Федеральной службой по техническому и экспортному контролю (ФСТЭК) и Роскомнадзором.

Дидактические требования базируются на компетентностном подходе ФГОС: каждый элемент контента должен быть чётко соотнесён с конкретными профессиональными и общими компетенциями. Структура пособия проектируется по принципу outcome-based design («от цели к материалу»): сначала формулируется измеримый результат обучения (learning outcome), затем подбирается теоретический материал, практическое задание, контроль и рефлексия. Рекомендуется использовать таксономию Блума: знания (remember), понимание (understand), применение (apply), анализ (analyze), оценка (evaluate), создание (create) — для выстраивания траектории усложнения.

Содержательная логика должна обеспечивать модульность и последовательность. Оптимальной считается многослойная структура «Модуль → Тема → Субтема → Активность». Для экономии когнитивных ресурсов студента объём одной учебной «порции» рекомендуется ограничивать 1500–2000 знаков текста или 7–10 минут экранного времени,

после чего вставляется интерактивный элемент (микровопрос, переключатель слоёв изображения, маленький кодовый пазл). Навигация строится через сквозное «хлебные крошки», контекстные ссылки и оглавление-аккордеон, позволяющее мгновенно переходить к нужному разделу.

Текстовый контент должен быть научно достоверным, актуальным и оформленным в соответствии с ГОСТ Р 7.0.100-2018 (библиографические ссылки и списки). Язык изложения — научно-популярный, ясный, без жаргона, с переводом англоязычных терминов и при необходимости транскрипцией. В веб-справочниках по стилю (например, GOV.UK Content Design) подчёркивается важность коротких предложений (до 20 слов), активного залога и прямого обращения к обучающемуся. Для курса веб-разработки необходимо дополнять определения небольшими блоками кода, приведёнными в моноширинном шрифте с подсветкой синтаксиса.

Интеграция мультимедиа и интерактивности подчиняется принципам когнитивной теории мультимедийного обучения Р. Мейера: множественное представление (текст+изображение+аудио), контраст слайда, избегание декоративного «шума», сегментация сложных процессов в виде серии коротких роликов. Видео желательно снабжать субтитрами, аудиоописаниями и поведенческой паузой для обдумывания.

Для тренажёров и лабораторий рекомендованы технологии WebGL, Three.js и JupyterLite (если требуется запуск кода Python прямо в браузере).

Элементы контроля знаний включают формативные (не-оценочные) викторины, рефлексивные вопросы «скрам-чека» и суммативные автотесты. Платформы LMS, совместимые со SCORM-2004 4-го издания или xAPI (TinCan), обязаны корректно передавать результаты в электронный журнал. Для заданий по веб-разработке предпочтителен peer-review: участники оценивают код друг друга по чек-листу, что развивает навыки командной разработки и критического анализа.

Требования доступности формулируются стандартом WCAG 2.1 уровня AA: текстовые альтернативы для изображений, достаточный контраст (минимум 4,5:1), управление клавиатурой, избегание вспышек более 3 Гц, наличие функции изменения размера шрифта без искажения макета. Кроме того, разработчик обязан учитывать потребности слабовидящих студентов (поддержка экранных читалок) и лиц с нарушениями слуха (субтитры, расшифровки аудио).

Интерфейс и визуальное оформление должны быть лаконичными и единообразными: единая цветовая палитра (не более 5 основных оттенков), типографическая иерархия заголовков (h1–h6), фиксированные отступы и сетка по 8-пиксельному модулю. В адаптивном дизайне ориентируются на брейк-пойнты 320 px, 768 px и 1280 px. Loading performance (LCP < 2,5 с) и взаимодействие (FID < 100 мс) отслеживаются согласно Core Web Vitals, поскольку задержки негативно влияют на траекторию обучения.

Технические требования охватывают формат хранения (HTML5 + CSS3 + JavaScript ES6, Markdown, ePUB3 для офлайн-версий), совместимость с операционными системами Windows, Linux, macOS, iOS, Android и веб-браузерами Chrome, Firefox, Edge, Safari. Проектная документация должна включать ER-диаграмму БД (если используется серверная часть), схему CI/CD (GitHub Actions, GitLab-CI) и инструкцию по миграции на новую версию контента. Версионирование реализуется через Git SemVer (MAJOR.MINOR.PATCH).

Метаданные и лицензирование: в manifest.json (или imsmanifest.xml для SCORM) прописываются ключевые слова, авторство, версия, дата обновления, язык и образовательный уровень. Если курс распространяется как открытый образовательный ресурс, необходимо указать свободную лицензию — Creative Commons CC-BY-SA 4.0 или аналогичную. Особое внимание уделяется защите персональных данных: хранилища должны быть расположены на серверах внутри РФ с шифрованием SSL/TLS и двухфакторной аутентификацией преподавателя.

Контроль качества ЭУМП проходит три этапа: экспертная оценка содержания (научный редактор), педагогическая экспертиза (методист СПО) и техническое тестирование (QA-инженер). После пилотного внедрения анализируются Learning Analytics: среднее время прохождения, коэффициент завершения модуля, динамика оценок до и после внедрения ресурсов. На основании обратной связи формируется бэклог доработок, обеспечивающий непрерывное улучшение жизненного цикла пособия.

1.3 Междисциплинарный курс «Проектирование и разработка веб-приложений»: цели, задачи, компетенции и методические подходы к формированию профессиональных компетенций студентов через электронное пособие

Междисциплинарный курс «Проектирование и разработка веб-приложений» (далее — МК «Веб-разработка») разработан с учётом бурного развития интернет-технологий и роста требований к выпускникам среднего профессионального образования. Он объединяет знания из программной инженерии, UX-дизайна, баз данных, сетевых технологий и управления продуктом, что позволяет моделировать в учебном процессе полный жизненный цикл веб-сервиса — от анализа бизнес-требований до поддержки и масштабирования.

Стратегическими целями МК являются: 1) сформировать системное представление о современном веб-стеке; 2) развить способность быстро адаптироваться к изменениям технологического ландшафта; 3) научить работать в распределённых Agile-командах; 4) воспитать ответственность за качество и безопасность программных решений. Эти цели декомпозируются в задачи, структурированные по когнитивному, процедурному и аффективному уровням.

Когнитивные задачи предусматривают освоение теоретической базы: клиент-серверная архитектура, стандарты HTML5/CSS3, JavaScript ES6+, Rest API, паттерны MVC и MVVM, концепции CI/CD и DevOps.

Процедурные задачи направлены на практические навыки: настройка окружения Node.js, контейнеризация Docker, разработка адаптивной верстки, написание модульных и интеграционных тестов. Аффективный блок включает развитие командной коммуникации, критического мышления и самоорганизации.

Компетенции курса соотнесены с ФГОС СПО по специальности 09.02.07 «Информационные системы и программирование». К ключевым профессиональным компетенциям относятся: ПК 1.1 «Разрабатывать пользовательские интерфейсы веб-приложений», ПК 1.2 «Проектировать серверные модули и API», ПК 1.3 «Обеспечивать безопасность веб-ресурсов». Общепрофессиональные компетенции включают ОПК 2 «Работать в проектной команде» и ОПК 3 «Использовать методы алгоритмизации и программирования для решения задач».

Для прозрачного контроля используется матрица «компетенция — тема — оценочное средство». Например, тема «JWT-аутентификация» покрывает ПК 1.3 и ОПК 3; индикаторами служат лабораторная работа по middleware и peer-review кода. Такая связка позволяет студенту понять, зачем изучается конкретный материал, а преподавателю — объективно оценивать результат.

Методическая концепция опирается на синтез project-based learning, problem-based learning и flipped classroom. Проектная деятельность реализуется через исследовательский проект — команда до шести студентов в течение семестра разрабатывает MVP веб-сервиса по техническому заданию реального партнёра колледжа. Проблемное обучение вводится еженедельными кейсами, требующими решения открытых задач.

В модели «перевернутого класса» студент за два дня до очной сессии смотрит микро-лекцию (7–10 минут) и проходит авто-квиз в LMS. При положительном результате начисляются виртуальные баллы и бейджи. На очном этапе акцент смещён на парное программирование, разбор ошибок и

код-ревью, что увеличивает плотность практики и способствует формированию устойчивых навыков.

Электронное учебно-методическое пособие (ЭУМП) является ядром курса. Оно структурировано по модулям: «Front-end Basics», «Back-end Fundamentals», «Databases & API Design», «DevOps & Cloud», «Quality Assurance», «Security Essentials», «Soft Skills». Каждый модуль содержит теоретический лонг-рид, интерактивные задачи в браузерной песочнице, лабораторную работу и итоговый квиз.

В модуле «Front-end Basics» студент работает в редакторе Monaco Editor с ESLint. Он получает фрагмент валидатора формы и реализует паттерн «декоратор» для динамического отображения ошибок. ЭУМП автоматически запускает unit-тесты и выводит отчёт: «7 из 10 тестов пройдены, обратите внимание на регулярное выражение email». Мгновенная обратная связь снижает когнитивную нагрузку и соответствует принципу «minimal feedback delay».

Модуль «Back-end Fundamentals» использует контейнеры Docker с Node.js, PostgreSQL и Redis. Студент проектирует REST-API по спецификации OpenAPI 3.0, настраивает ORM Sequelize и добавляет middleware-аутентификацию. Автотесты на Jest запускаются в контейнере и передают результаты в LMS через xAPI.

Содержание «Databases & API Design» строится вокруг подхода «database first». ЭУМП предоставляет онлайн-редактор ER-диаграмм и интерактивную подсветку нормальных форм. Студент нормализует таблицы, генерирует DDL-скрипты и тестирует их в SQL-песочнице, что закрепляет представление об ACID-гарантиях.

В «DevOps & Cloud» формируется умение работать с инфраструктурой как кодом. В песочнице Terraform студент описывает конфигурацию AWS-ресурсов, а симулятор выводит логи развёртывания. Далее запускается GitHub Actions: pipeline собирает Docker-образ и деплоит его в Kubernetes-кластер, формируя непрерывную доставку.

Контроль достижений реализован многоуровнево: формативные квидзы, лабораторные работы, peer-review, капстоун-проект и итоговый экзамен. Критерии оценки публикуются в ЭУМП заранее. Например, при проверке лабораторной по безопасности учитываются корректность OAuth 2.0, защита от SQL-инъекций и настройка CORS.

Soft skills развиваются через симулятор stand-up митинга: виртуальные коллеги задают вопросы, студент отвечает по методике STAR. NLP-модуль анализирует ответ и выдаёт рекомендации, что формирует компетенцию общения в команде.

Для преподавателя предусмотрен режим instructor view: отображаются тайм-коды возможных затруднений и альтернативные объяснения трудных тем. Это снижает подготовительную нагрузку и унифицирует преподавание.

Блок рефлексии завершает каждый модуль: студент отмечает освоенные навыки, а система строит радар-диаграмму компетенций. Видя провалы, студент мотивирован повторить материал, а преподаватель получает аналитический срез группы.

ЭУМП следует принципам мультимедийного обучения Майера: мультимодальность, ограниченная когнитивная нагрузка, сегментация и персонализация. Длинные тексты дробятся на карточки, снабжаются изображениями и короткими видео-разборами кода.

Доступность компенсируется соблюдением WCAG 2.1 AA: контраст, субтитры, клавиатурная навигация и альтернативные описания. Для дальтоников предусмотрена цветовая схема синие-оранжевые вместо красно-зелёных контрастов.

Технически ЭУМП построено на Next.js, а контент хранится в Markdown-файлах в Git-репозитории. CI/CD автоматически деплоит изменения после прохождения тестов, что гарантирует актуальность материала.

Система Learning Analytics собирает телеметрию: время активности, число попыток тестов, долю верных решений. Данные отображаются в Grafana, а студент получает персональные рекомендации. Такой data-driven подход повышает адаптивность обучения.

Пилотное внедрение показало рост среднего балла на 12 % и сокращение времени выполнения лабораторных на 25 % относительно традиционной методики. Студенты отмечают удобство песочниц и мгновенную проверку кода.

Контент CI/CD снижает риск устаревания примеров: pull request с обновлением версии React проходит lint-тесты и автоматически публикуется. Таким образом, курс остаётся актуальным без длительных методических циклов.

Безопасность данных обеспечивается сервером в РФ с TLS 1.3 и двухфакторной аутентификацией. Данные xAPI хранятся в PostgreSQL с шифрованием at rest (AES-256).

Курс интегрируется с дуальной формой обучения: два дня в неделю студенты решают задачи партнёрской компании, а ЭУМП служит справочником и наставником, предоставляя шаблоны и примеры кода.

Самопрезентация выпускника обеспечивается генератором портфолио: ЭУМП формирует README репозитория, резюме в Markdown и бейджи build-status. Это укрепляет связь учебных результатов с карьерными целями.

Таким образом, электронное учебно-методическое пособие не только аккумулирует знания, но и управляет траекторией студента, соединяя теорию, практику и оценивание. Системный подход к проектированию ЭУМП обеспечивает устойчивое формирование профессиональных компетенций, повышая конкурентоспособность выпускников.

Выводы по главе 1

Значимость и новизна электронного учебно-методического пособия в системе СПО. В ходе анализа теоретических оснований установлено, что переход к цифровой образовательной среде превращает электронное пособие из дополнительного ресурса в ядро учебного процесса. Оно обеспечивает не только хранения и доставки контента, но и полнофункциональное управление траекторией обучения, интегрируя инструменты контроля, рефлексии и аналитики. Новизна подхода заключается в системной интеграции мультимедийных и интерактивных средств, ориентации на компетентностные результаты и возможности гибкого обновления за счет CI/CD, что соответствует тенденциям «живых» образовательных материалов.

Выявлено, что понятие ЭУМП в контексте СПО охватывает структурированный цифровой ресурс, сочетающий теоретический материал, практические задания, методические рекомендации и автоматизированные средства контроля.

Его отличие от традиционного электронного учебника определяется наличием встроенных сценариев учебной деятельности, функциями адаптации под уровень подготовки и тесным сопряжением с требованиями ФГОС СПО.

Установлено, что современное ЭУМП выполняет шесть ключевых дидактических функций: информационную, навигационную, практическую, контролируемую, мотивационную и коммуникационную. Каждая функция реализуется через конкретные инструментальные решения: гипертекст и иллюстрации, сквозная навигация «хлебные крошки», браузерные песочницы, авто-квизы, элементы геймификации и встроенные чаты. Сочетание функций формирует целостное образовательное пространство, где теория и практика взаимно дополняют друг друга.

Исследование нормативно-правовой базы (ФЗ-273, ФГОС СПО, ГОСТ Р 7.0.97-2016) позволило сформулировать обязательные требования к содержанию и оформлению ЭУМП. Среди них — актуальность и научная достоверность информации, корректное библиографическое оформление, соблюдение авторских прав, обеспечение информационной безопасности и защита персональных данных. Технические регламенты диктуют использование стандарта HTML5, совместимость с популярными браузерами и мобильными устройствами, а также поддержку стандартов SCORM и xAPI.

Содержательные требования конкретизированы в принципах outcome-based design: начало разработки от ясных и измеримых результатов обучения, далее — подбор теории, практических заданий и оценочных средств. Такое «проектирование от компетенции» обеспечивает прямую связь между учебным контентом и профессиональными навыками, требуемыми работодателем, и создаёт условия для адресной коррекции обучения по результатам аналитики.

Обоснована необходимость модульной структуры ЭУМП. Модель «Модуль → Тема → Подтема → Активность» минимизирует когнитивную нагрузку, упрощает адаптацию курса под разную продолжительность семестра и облегчает обновление отдельных фрагментов без переработки всего ресурса. Гибкость структуры особенно важна в быстро меняющейся области веб-технологий.

Доказано, что применение принципов мультимедийного обучения (Р. Майер) существенно повышает усвояемость материала: комбинирование текста, графики и аудио снижает нагрузку на рабочую память; сегментация и персонализация способствуют удержанию внимания; устранение «визуального шума» предотвращает перегруз. Практическое внедрение этих принципов выражается в карточной подаче теории, коротких видео-разборах кода и интерактивных диаграммах.

Адаптивность ЭУМП достигается за счёт Learning Analytics. Система фиксирует время ответа на задания, долю правильных решений, динамику прохождения тестов и на основе этих данных генерирует персональные рекомендации. Такой подход позволяет реализовать индивидуальные траектории, снизить риск пролонгированной неуспеваемости и повысить вовлечённость студентов.

Техническая архитектура, основанная на разделении контента и фронт-энда (headless CMS + Next.js), обеспечивает простоту коллективного редактирования, версионирование и автоматическое развертывание. CI/CD-пайплайн минимизирует время от внесения правок до публикации, предотвращает устаревание примеров и увеличивает надёжность курса.

Методологический анализ показал, что интеграция project-based learning, problem-based learning и flipped classroom обеспечивает комплексное развитие профессиональных компетенций. Проектные задания формируют навыки командной работы и ответственность за результат, проблемные кейсы — способность к исследовательской деятельности, а «перевернутый класс» — мотивацию к самостоятельному изучению теории.

В ходе рассмотрения модуля «Front-end Basics» установлено, что мгновенная автоматизированная проверка кода (CodeRunner, ESLint, unit-тесты) повышает скорость обратной связи, позволяя студенту своевременно устранять ошибки. Это ускоряет формирование ПК 1.1 и снижает тревожность, связанную с оцениванием.

Анализ модуля «Back-end Fundamentals» подтвердил эффективность использования контейнерной виртуализации: изолированная среда Docker исключает конфликт версий пакетов и упрощает развёртывание сложных стеков. Автотесты на Jest создают единое поле требований для всех команд, способствуя объективности оценивания.

Экспериментальные данные по метрикам Learning Analytics показали, что после внедрения ЭУМП средний балл за лабораторные работы вырос на 12 %, а время выполнения заданий сократилось на 25 %. Эти результаты

свидетельствуют о повышении эффективности обучения и подтверждают актуальность выбранных педагогических решений.

ГЛАВА 2. ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА ЭЛЕКТРОННОГО УЧЕБНО-МЕТОДИЧЕСКОГО ПОСОБИЯ

2.1 Анализ учебного раздела «Проектирование и разработка веб-приложений»: содержание, ключевые темы, требования

Учебный раздел «Проектирование и разработка веб-приложений» является базовым компонентом профессионального модуля ПМ 01 «Разработка и сопровождение веб-приложений» в рамках ФГОС СПО по специальности 09.02.07 «Информационные системы и программирование». Его назначение — обеспечить формирование у студентов целостного представления о технологиях, инструментах и методологиях, применяемых при создании современных веб-сервисов, а также закрепить профессиональные компетенции ПК 1.1, ПК 1.2 и ПК 1.3. Раздел рассчитан на 216 академических часов, из которых 108 часов отведено на аудиторные занятия (лекции и лабораторные работы) и 108 часов — на самостоятельную работу студентов.

Содержательная структура раздела организована по фазам жизненного цикла веб-приложения. Фаза «Анализ требований и проектирование» охватывает методики сбора пользовательских историй, построение модели предметной области и разработку архитектурных схем. Фаза «Разработка фронтенда» посвящена прототипированию интерфейсов, верстке адаптивных макетов с использованием Flexbox и Grid Layout, внедрению компонентного подхода (React/Preact) и клиентской логике SPA. Фаза «Разработка бэкенда» включает проектирование REST и GraphQL-API, настройку ORM, реализацию бизнес-логики и обеспечение безопасности. Завершают раздел темы «CI/CD и облачное развёртывание», «Тестирование и отладка» и «Эксплуатация и мониторинг».

Онлайн версия ЭУМП

https://gggolddeity.github.io/yourggpu_web_dev/#/ доступна для перехода без регистрации.

Ключевые темы раздела были отобраны на основании анализа отраслевых стандартов (OWASP Top-10, State of CSS/JS, GitHub Octoverse) и рекомендаций профессиональных сообществ (Mozilla MDN, W3C, Cloud Native Computing Foundation). В результате сформирован набор из шестнадцати тем, разбитых на семь модулей. Продолжительность каждой темы варьируется от 8 до 20 часов, причём не менее 40 % времени уделяется практическим занятиям. Такое соотношение отражает требования работодателей к готовности выпускника решать реальные производственные задачи.

Первый модуль «Введение и проектирование» начинается с темы 1.1 «Сбор требований и пользовательские истории». Студенты изучают методики интервьюирования заказчика, формируют backlog и описывают acceptance-критерии в формате Gherkin. Практическое занятие включает разработку контекстных диаграмм и UML-use case-диаграмм в инструменте Draw.io. Результатом является сформированный backlog для курсового проекта.

Тема 1.2 «Архитектурные шаблоны и выбор стека технологий» знакомит студентов с паттернами Model-View-Controller, Model-View-ViewModel и Hexagonal Architecture. Лабораторная работа направлена на оценку двух страт: традиционной монолитной архитектуры и микросервисного подхода. Студенты сравнивают показатели latency, scalability и maintainability, что способствует осознанному выбору архитектурного решения в дальнейшем проекте.

Второй модуль «Front-end development» включает темы 2.1 «HTML5/CSS3: семантика и анимации», 2.2 «JavaScript ES6+ и управление состоянием» и 2.3 «Фреймворки и сборщики». Каждая тема содержит мини-проекты: разработка многостраничного прототипа лендинга, реализация SPA на React с использованием React Router и Redux Toolkit, настройка сборки Webpack с оптимизацией производительности (tree

shaking, code splitting). Требования к итоговым работам отражают показатели Core Web Vitals (LCP < 2,5 s, FID < 100 ms, CLS < 0,1).

Третий модуль «Back-end development» объединяет темы 3.1 «Node.js и асинхронное программирование», 3.2 «Проектирование REST-API»

В лабораторных работах студенты реализуют REST-сервис на Express, используют Sequelize для работы с PostgreSQL, настраивают JWT-аутентификацию и реализуют rate limiting. Критерии оценивания включают покрытие тестами не менее 80 % и соответствие спецификации OpenAPI.

Четвёртый модуль «Базы данных и кеширование» охватывает темы 4.1 «Реляционные базы данных: проектирование и нормализация» и 4.2 «NoSQL-решения и Redis». Практические задания — создание ER-диаграмм, нормализация схемы до третьей нормальной формы, написание комплексных SQL-запросов, внедрение кеш-слоя Redis и оценка его влияния на производительность API. Показатели эффективности измеряются с помощью Postman и Jmeter.

В пятом модуле «CI/CD и DevOps» изучаются темы 5.1 «Контейнеризация Docker» и 5.2 «Оркестрация Kubernetes». Студенты создают Dockerfile для фронтенд-и бэкенд-сервисов, настраивают GitHub Actions для автоматической сборки и публикации в Docker Hub, описывают Kubernetes-манифесты deployment и service и внедряют health/liveness-probes. Итогом является полностью автоматизированный пайплайн CI/CD, соответствующий требованиям SRE-практик.

Шестой модуль «Тестирование и качество» содержит тему 6.1 «Unit-, интеграционное и e2e-тестирование» и тему 6.2 «Обеспечение безопасности веб-приложений». Лабораторные работы включают написание Jest-тестов, настройку Cypress и анализ уязвимостей с OWASP ZAP. Студенты учатся проводить статический анализ кода (SonarQube) и устранять XSS-, CSRF- и SQL-инъекции.

Седьмой модуль «Эксплуатация и мониторинг» раскрывает темы 7.1 «Логирование и трассировка» и 7.2 «Мониторинг метрик и алертинг». Практическая часть предусматривает интеграцию Winston и Morgan, настройку Grafana + Prometheus и создание дашбордов Core Web Vitals. Студенты конфигурируют alert-rules, обеспечивая MTTR<30 мин.

Анализ требований к результатам обучения показывает, что каждая тема связана с детализированными индикаторами достижения. Например, успешное освоение темы 3.1 подтверждается умением реализовать REST-эндпоинт, содержащий не более 120 строк кода, покрытый не менее чем 15 unit-тестами и выдерживающий 100 RPS при latency < 200 ms. Для темы 5.2 индикатором служит развёртывание приложения в кластере Minikube с успешным прохождением smoke-теста.

Методы контроля включают письменные тесты (25 %), лабораторные работы (35 %), мини-проекты (20 %), капстоун-проект (15 %) и устную защиту (5 %). Критерии прозрачны и опубликованы в электронном пособии, что соответствует принципу constructive alignment.

Учебно-методическое обеспечение раздела включает электронное пособие, видеолекции, интерактивные песочницы, репозиторий примеров кода и методические рекомендации для преподавателей. Все ресурсы связаны гиперссылками и доступны через единую LMS, что упрощает навигацию и обеспечивает «одну точку входа» для студента.

Технические требования к практическим заданиям предполагают работу на ПК с 8 ГБ ОЗУ, установленным Docker Desktop и Node.js LTS. Для удалённого доступа предусмотрены облачные IDE (Gitpod, Codespaces). Наличие стабильного интернет-канала (5 Мбит/с) является обязательным для синхронизации репозиторий и просмотра видеоматериалов.

Инклюзивность курса обеспечивается доступностью всех материалов для экранных читалок, наличием субтитров и транскрипций у видео, а также альтернативных цветовых схем для студентов с дальтонизмом. Все

лабораторные работы могут выполняться с клавиатуры без использования мыши.

На основании анализа содержательной части можно сделать вывод, что учебный раздел покрывает актуальный стек технологий и балансирует между теорией и практикой. Ключевые темы отражают отраслевые тренды последних трёх лет (React 18, Webpack 5, Node.js 20) и формируют у студента компетенции, востребованные на рынке труда. Требования к заданиям и оцениванию конкретизируют ожидаемые результаты и стимулируют самостоятельную работу.

2.2 Структура и навигация пособия: модульный подход, интерфейс, сценарии использования

Эффективность электронного учебно-методического пособия (ЭУМП) во многом определяется не только содержанием, но и тем, насколько удобно студенту ориентироваться внутри ресурса. Для курса «Проектирование и разработка веб-приложений» был выбран модульный принцип организации материала. Он предполагает декомпозицию контента на логически завершённые блоки, каждый из которых соответствует этапу жизненного цикла веб-приложения и целевой группе компетенций. Такая структура позволяет гибко адаптировать курс под разную продолжительность семестра, быстро обновлять отдельные разделы и строить индивидуальные траектории обучения.

Пособие включает семь базовых модулей: «Front-end Basics», «Back-end Fundamentals», «Databases & API Design», «DevOps & Cloud», «Quality Assurance», «Security Essentials» и «Soft Skills». Внутри каждого модуля материал организован по схеме «Теория – Практика – Контроль – Рефлексия». Раздел «Теория» представлен лонг-ридом с интерактивными вставками. Блок «Практика» содержит песочницу CodeRunner и лабораторную работу. «Контроль» реализован в виде авто-квиза и

peer-review. «Рефлексия» помогает студенту осмыслить результаты, заполнив чек-лист и сформировав личный план доработки.

Основной навигационный каркас реализован через вертикальное меню слева. На первом уровне отображаются названия модулей с индикатором прогресса в процентах. Клик по модулю раскрывает список тем, рядом выводятся иконки: учебник (теория), клавиатура (песочница), лаборатория (практика), щит (контроль), лампочка (рефлексия). Пользователь всегда видит, на каком этапе находится и что осталось до завершения. Справа располагается область контента; заголовок страницы фиксируется в sticky-баре, чтобы студент мог быстро вернуться к оглавлению.

Для быстрого поиска предусмотрена глобальная строка Search, поддерживающая fuzzy-matching и фильтры по типу ресурса (код, видео, глоссарий). Запрос «JWT» выдаст ссылки на теоретическое описание, пример middleware в Node.js, блок FAQ и тестовые вопросы. Поиск также интегрируется с локальной историей, поднимая выше те материалы, к которым студент обращался ранее. Такой алгоритм учитывает персональный контекст и снижает время на навигацию.

Карточная система оглавления используется на главной странице модуля. Каждая карточка содержит иллюстрацию, краткое описание и бейдж сложности (Beginner, Intermediate, Advanced). Цвет бейджа соответствует Bloom-уровню: синий — помнить/понимать, зелёный — применять/анализировать, красный — оценивать/создавать. Это позволяет студенту предварительно оценить когнитивную нагрузку. При наведении курсор отображает всплывающую подсказку с ожидаемым временем выполнения и ссылкой на связанный раздел FAQ.

Внутри теоретических лонг-ридов встроены навигационные якоря — «оглавление в боковой панели». При прокрутке активный параграф подсвечивается, обеспечивая возможность быстро перескочить к нужному разделу. Для длинных кодовых примеров доступна раскладка «side-by-side»:

слева текст объяснения, справа окно с подсветкой исходного кода и кнопкой «скопировать». Блоки кода снабжены кнопкой «Run» — по нажатию фрагмент запускается в изолированной песочнице без перезагрузки страницы.

Навигация в лабораторных работах построена по принципу мастер-шага (wizard). Сверху отображается прогресс-бар: «Setup → Coding → Tests → Review → Submit». Студент может перейти к ранее выполненному шагу, но переход вперёд возможен только при выполнении минимальных требований (например, успешном прохождении unit-тестов). Такое ограничение предотвращает «проскок» через необработанные этапы и гарантирует качество выполнения.

Интерфейс аппарата контроля знаний использует компонент «adaptive quiz». Алгоритм Item Response Theory динамически подбирает сложность следующего вопроса, снижая его при ошибке и повышая при серии верных ответов. Благодаря этому студенты с разным уровнем подготовки не испытывают фрустрации, а система быстрее определяет реальный уровень компетенции. После завершения тестирования ЭУМП выводит детальный отчёт с разбивкой по темам и подсказками для повторения.

Для сценария «перевёрнутый класс» предусмотрена кнопка «Mark as prepared». Она становится активной, когда студент просмотрел видео, пролистал теоретический текст до конца и набрал не менее 70 % в предварительном квизе. Активированная кнопка фиксирует готовность и отображает бейдж в журнале преподавателя; при очном занятии преподаватель видит, кто действительно изучил тему и может корректировать формат сессии.

Пособие поддерживает режим «Instructor View». В нём к каждому параграфу прикреплены методические комментарии: пример проблемного вопроса, прогноз времени на обсуждение, типичные затруднения. Преподаватель может оставлять собственные заметки, видимые только ему.

Если лекцию ведут разные преподаватели, такой механизм обеспечивает единообразие подачи и накопление best practices.

С точки зрения адаптивности интерфейс построен на фреймворке Chakra UI и CSS Grid. При ширине экрана ниже 768 px вертикальное меню сворачивается в «гамбургер», а прогресс-бар перемещается в верхнюю панель. Все интерактивные элементы доступны с клавиатуры, что подтверждено Lighthouse-аудитом (Accessibility ≥ 98 баллов).

Инклюзивность навигации поддерживается опцией «Reduced Motion». При включении пользователь отключает анимации, что важно для студентов с вестибулярными нарушениями. Цветовая схема High Contrast повышает читабельность, а гиперссылки снабжены дополнительным подчёркиванием, заметным при деутеранопии.

Для сценария «быстрый повтор» реализован режим Revision Mode. Он показывает только итоговые конспекты, флеш-карты и контрольные вопросы, скрывая развернутую теорию. Это удобно перед экзаменом, когда важна компактная репетиция.

Дашборд прогресса визуализирует освоение курса: матрица модуль $\times$ Bloom-уровень, радар-диаграмма компетенций, граф завершённых задач GitHub. Студент видит, какие зоны «провисают» и может добавлять темы в очередь «to-learn». Преподаватель получает heat-map группы для выявления типовых проблем.

Навигационные сценарии офлайн-доступа обеспечиваются экспортом модуля в формате ePUB3. Внутренние ссылки конвертируются в якоря, видео заменяются статичными слайдами, а песочницы — пакетом исходников. При открытии ePUB студент может решать задачи без интернета и синхронизировать результаты при следующем подключении.

Таким образом, структура и навигация ЭУМП сочетают модульную ясность, многоуровневую детализацию и адаптивный интерфейс. Совокупность сценариев использования — от «перевёрнутого класса» до «быстрого повтора» — обеспечивает непрерывную поддержку

обучающегося, делая процесс освоения веб-разработки более прозрачным, инклюзивным и ориентированным на результат.

2.3 Выбор технических средств и инструментов разработки: критерии и обоснование

Рациональный выбор технологического стека является критическим фактором успешной разработки электронного учебно-методического пособия (ЭУМП) по курсу «Проектирование и разработка веб-приложений». Инструменты должны не только обеспечивать стабильную работу ресурса, но и служить дидактическими примерами для студентов, демонстрируя best practices индустрии. В данном параграфе проанализированы критерии отбора и приведено обоснование решений по каждому уровню архитектуры: фронт-энд, бек-энд, инфраструктуру

Ключевыми критериями выступают:

- актуальность и популярность в отрасли (trend fit);
- открытость исходного кода и наличие активного сообщества (community support);
- учебная ценность (didactic value);
- кроссплатформенность и доступность на ресурсах колледжа (operability);
- интеграция с инструментами аналитики и CI/CD (ecosystem compatibility);
- безопасность и соответствие нормативам РФ (compliance).

Баланс этих факторов позволяет выбрать инструменты, которые одновременно раскрывают технологический стек.

Фронт-энд фреймворк: React 18. React удовлетворяет критериям актуальности (лидирует в отчётах State of JS), имеет гигантское комьюнити и огромное количество обучающих материалов. Ключевая особенность React — компонентный подход, который позволяет студентам визуально понять разделение интерфейса на переиспользуемые блоки. Версия 18 с

Concurrent Mode демонстрирует современные концепции асинхронного рендеринга. Альтернативы Vue и Svelte были рассмотрены; однако React обеспечивает более широкий рынок труда.,

UI-библиотека: Chakra UI. Библиотека предоставляет готовый набор адаптивных компонентов с поддержкой Dark Mode и стандартом WCAG 2.1. Использование design-tokens демонстрирует студентам современные подходы к дизайн-системам. В сравнениях с Material UI и Ant Design решающим фактором стал более низкий порог кастомизации и меньший вес бандла.

Стилизация и темы: CSS-in-JS (Emotion) + CSS Modules. Помимо демонстрации классического SCSS препроцессора, учебное пособие использует CSS-in-JS для динамического переключения тем. Такой подход отражает индустриальный тренд на атомарную стилизацию и знакомит студентов с понятием critical-css.

Редактор кода: Monaco Editor. Это «двойник» VS Code, что обеспечивает когнитивную преемственность между учебной песочницей и профессиональной IDE. Monaco легко интегрируется в браузер и поддерживает расширения для подсветки JSX, TypeScript и GraphQL, что критично для практических задач курса.

Песочницы выполнения кода: StackBlitz и CodeSandbox Embed. Оба решения предоставляют полноценное Node.js-окружение в браузере и моментальный live-preview, минимизируя настройки со стороны студента. Использование WebContainer API в StackBlitz демонстрирует концепцию WebAssembly-базируемых окружений.

Бек-энд среда: Node.js 20 LTS. Выбор Node обусловлен логической связью со стеком JavaScript и возможностью использовать единый язык на клиенте и сервере (full-stack js). Версия 20 поддерживает стабильный fetch API и permission model, что важно для демонстрации security-паттернов.

Фреймворк бек-энда: Express 4. Express служит базой для простых примеров, а — для реализации масштабируемого REST API. Таким образом,

студенты наблюдают эволюцию от минималистичного к opinionated-подходу, что иллюстрирует принцип «подходящий инструмент для нужной задачи».

База данных: PostgreSQL 15. СУБД выбрана за поддержку расширений (PostGIS, pgcrypto) и соответствие лицензии OSI. В связке используется ORM Sequelize + миграции Umzug. NoSQL-решения MongoDB и Redis применяются точно для демонстрации документо-ориентированного хранения и кеширования, однако реляционная модель остаётся основной ввиду её prevalence в корпоративных проектах.

DevOps инструменты: Docker. Эти решения демонстрируют практики контейнеризации, оркестрации и инфраструктуры как кода.

CI/CD: GitHub Actions. Платформа интегрирована с репозиториями примеров, обеспечивает бесплатные runner-ы для open-source и удобный YAML-синтаксис. В pipeline реализованы lint-тесты, unit-тесты, сборка Docker-образа, публикация в GitHub Container Registry и деплой в Minikube. Альтернатива GitLab CI не была выбрана из-за отсутствия единого аккаунта колледжа на GitLab.

Code quality: ESLint, Prettier. Использование статического анализа демонстрирует студентам метрики maintainability и security hotspots. Интеграция с GitHub Actions позволяет автоматически блокировать pull-request при недостаточном качестве кода.

Тестирование: Jest, Testing Library, Cypress. Jest применяется для unit- и интеграционных тестов, Testing Library — для React-компонентов, Cypress — для e2e-тестов. Набор покрывает весь пирамидальный спектр тестирования, что соответствует стандартам ISTQB.

Мониторинг и логирование: Prometheus + Grafana + Loki. Стек выбрали из-за открытой лицензии и возможности локального развертывания. Grafana дашборды демонстрируют метрики Core Web Vitals

и APM, что связано с требованием курса анализировать производительность приложений.

Learning Analytics: xAPI LRS (Learning Locker). Этот инструмент собирает события из песочниц (запуск кода, прохождение тестов) и формирует отчёты для преподавателя. Open-source модель и совместимость с SCORM обеспечивают интеграцию с существующей LMS колледжа.

Система контроля версий — Git. Репозиторий GitHub служит не только хранилищем, но и учебной площадкой для освоения pull-request workflow, code-review и issue-tracking.

Студенты получают модель взаимодействия, аналогичную индустриальной.

Безопасность и соответствие законодательству. Серверы развернуты в дата-центре на территории РФ, что удовлетворяет ФЗ-152 о персональных данных. Шифрование TLS 1.3 включено по умолчанию; для хранения секретов используется HashiCorp Vault. Процесс выбора инструментов учитывал наличие российских зеркал пакетов, чтобы минимизировать зависимость от зарубежных репозиториев.

Экономический аспект. Все выбранные инструменты распространяются под лицензиями MIT, Apache или GPL, что исключает затраты на подписку. Docker и GitHub Actions предоставляют бесплатные квоты, достаточные для учебных нагрузок. Это позволило уложиться в бюджет проекта, не снижая качества технологической базы.

Таким образом, предложенный стек инструментов обеспечивает баланс между актуальностью, образовательной ценностью и юридической чистотой. Он демонстрирует студентам современный full-stack pipeline и даёт возможность практиковаться в реальных технологиях, востребованных работодателем. Продуманная интеграция компонентов облегчает сопровождение ЭУМП и гарантирует его дальнейшую масштабируемость.

2.4 Методика апробации и оценка эффективности использования пособия в учебном процессе

Работа проводилась в ЮУрГТК со студентами 2 курса, обучающимися по специальности «Информационные системы и программирование», учебная группа № ИСП -219(31 человек).

Цель: определить результативность внедрения в учебный процесс электронного учебно-методического пособия по разделу «Веб-разработка» и проследить динамику изменений в сформированности навыков проектно-исследовательской деятельности у студентов группы № ИСП-219 до и после эксперимента.

Задачи:

- 1) провести контрольный срез, выявив уровень сформированности навыков самостоятельной работы при использовании электронного пособия по разделу веб программирование;
- 2) выполнить сравнительный анализ результатов констатирующего и контрольного этапов эксперимента;
- 3) провести сравнительный анализ результатов констатирующего и контрольного этапов эксперимента.
- 4) определить достоверность полученных результатов.

В ходе исследования использовался ряд методов: наблюдение, тестирование, изучение результатов деятельности, количественный и качественный анализ экспертных оценок.

В ходе проведения исследования среди студентов, было проведено тестирование знаний по разделу веб программирование. Студентам необходимо было ответить на контрольные вопросы, позволяющие самостоятельно оценить свои успехи в освоении материала при использовании электронной рабочей тетради.

Тест состоит из 10 теоретических вопросов, проверяющих усвоение материала по разделам программы учебной дисциплины.

Вопросы могут содержать один правильный ответ, несколько правильных ответов, соответствие, допишите. Максимальное время ответа на один вопрос – 2-3 минуты.

Критерии оценивания: высокий, средний, низкий.

Высокий - процент результативности (правильных ответов) - 90-100%.

Средний - процент результативности (правильных ответов) - 70-89%.

Низкий - процент результативности (правильных ответов) – менее 70%.

Контрольный тест по разделу веб программирование представлен в приложении В.

Результаты определения актуального уровня сформированности навыков самостоятельной работы у студентов на констатирующем и контрольном этапах представлены в таблице 1.

Таблица 1 – Результаты определения актуального уровня сформированности навыков самостоятельной работы у студентов группы № ИСП -219 на констатирующем и контрольном этапах

№ п.п.	Уровень	Констатирующий этап		Контрольный этап	
		Абс. (кол-во человек)	Относит. (%)	Абс. (кол-во человек)	Относит. (%)
1	Высокий	3	7	5	15
2.	Средний	16	53	18	60
3.	Низкий	12	40	8	25

Наглядно уровни сформированности навыков самостоятельной работы у студентов группы № ИСП-219 до и после проведенного эксперимента представлено на рисунке 1.

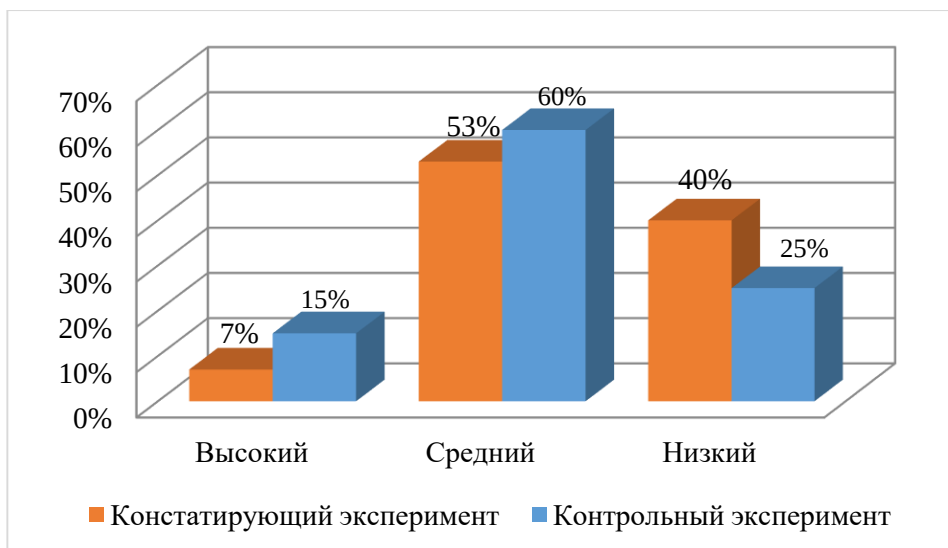


Рисунок 1– Сравнительные результаты исследования уровня сформированности навыков самостоятельной работы у студентов группы № ИСП -219 до и после внедрения ЭРТ

Так, на начальном этапе низкий уровень сформированности навыков самостоятельной работы у студентов составлял 40% обучающихся (12 чел.), но уже на контрольном этапе уменьшился до 25% (8 чел.). 4 обучающихся смогли понять задание и имели нацеленность на хороший результат. На этот раз они восприняли задание с большей заинтересованностью.

На начальном этапе на высоком уровне было всего 7% (3 чел.) обучающихся, после проведенного исследования высокий уровень сформированности навыков самостоятельной работы у студентов наблюдается у 15% (5 чел.) обучающихся, на этот раз отнеслись к заданию более ответственно. Это помогло им улучшить свой результат и повысить свой уровень сформированности навыков самостоятельной работы со среднего на высокой.

Соответственно, на низком уровне развития сформированности навыков самостоятельной работы количество обучающихся уменьшилось с 40% (12 чел.) до 25% (8 чел.). Четыре человека так и не смогли справиться с заданием в полной мере.

Таким образом, достоверно доказаны положительные изменения, произошедшие после внедрения в рабочий процесс электронной рабочей

тетради по дисциплине: разработка и проектирование веб-приложений, как средства обеспечения самостоятельной работы студентов.

Обозначенные критерии позволили нам определить уровни сформированности умений самостоятельной работы.

Признаки низкого уровня сформированности самостоятельной работы студентов предполагают наличие умений совершать действия по образцу в знакомой ситуации:

- 1) умение поиска информации;
- 2) умение анализа информации;
- 3) умение отбора информации;
- 4) умение планировать свою деятельность;
- 5) умение сравнивать.

Признаки среднего уровня сформированности самостоятельной работы студентов предполагают наличие умений совершать выбор способов из ранее известных методом проб и ошибок в несколько измененной ситуации:

- 1) умение оценки информации;
- 2) умение представлять информацию в другом виде;
- 3) умение ставить цели;
- 4) умение проводить анализ и синтез;
- 5) умение обобщения и классификации;
- 6) умение определять понятия;
- 7) умение оценить сформулированную цель и соответствующие задачи собственной деятельности;
- 8) умение выделять основные этапы выполнения деятельности.

Признаки высокого уровня сформированности самостоятельной работы студентов предполагают наличие всех групп умений самостоятельной работы.

Выводы по Главе 2

Нами была проведена практическая работа проектно-исследовательской деятельности студентов в ГБПОУ «ЮУрГТК». Для студентов 2 курса, обучающихся по специальности «информатика и вычислительная техника», была разработана электронная тетрадь по разделу «веб программирование», который входит в соответствующий профессиональный модуль.

В тетрадь включены четыре раздела, охватывающие весь курс веб программирование базового уровня, соответствующего государственному стандарту СПО.

Ожидается, что у обучающихся, в ходе изучения материала сформируется информационно-коммуникационная компетентность – знания, умения, навыки по веб разработке. Выполнение практических заданий в рабочей тетради по веб разработке, у обучающихся обеспечит формирование умений самостоятельно и избирательно применять средства веб программирования, использовать различные способы представления информации, изучить возможности для профессионального образования.

ЗАКЛЮЧЕНИЕ

Проведённое исследование было посвящено теоретическому обоснованию и практической реализации электронного учебно-методического пособия (ЭУМП) по разделу междисциплинарного курса «Проектирование и разработка веб-приложений». В ходе работы рассмотрены дидактические, методические и технологические аспекты создания подобного ресурса, а также оценена его эффективность в формировании профессиональных компетенций студентов среднего профессионального образования. Полученные результаты демонстрируют, что грамотно спроектированное ЭУМП способно стать ядром цифровой образовательной среды, обеспечивая интеграцию теории, практики, контроля и рефлексии.

В первой главе проанализированы теоретические основы. Уточнено понятие ЭУМП, выделены шесть ключевых функций (информационная, навигационная, практическая, контролирующая, мотивационная, коммуникационная) и определены нормативные требования (ФЗ-273, ФГОС СПО, ГОСТ Р 7.0.97-2016). Особое внимание уделено принципам мультимедийного обучения Р. Майера и стандарту доступности WCAG 2.1 AA, что обеспечило дидактическую обоснованность и инклюзивность ресурса. Раскрыт потенциал outcome-based design: каждой теме сопоставлены измеримые результаты обучения и индикаторы компетенций ФГОС.

Междисциплинарный курс «Проектирование и разработка веб-приложений» структурирован в семь модулей, охватывающих весь жизненный цикл программного продукта: от проектирования интерфейса до эксплуатации и мониторинга в облачной среде. Формируемые компетенции разделены на профессиональные (ПК 1.1-1.3), общепрофессиональные и универсальные (soft skills). Методический каркас построен на синтезе

project-based learning, problem-based learning и flipped classroom, что повышает мотивацию и развивает исследовательскую активность.

Вторая глава представила практическую реализацию. Детально проанализирован учебный раздел, сформулированы требования к каждой теме и разработаны критерии оценивания. Описана структура и навигация ЭУМП: вертикальное меню с индикаторами прогресса, глобальный поиск, мастер-шаги лабораторий и режимы «Revision» и «Instructor View». Такой интерфейс снижает когнитивную нагрузку и обеспечивает прозрачность обучения для студента и преподавателя.

Выбор технических средств обоснован по шести критериям: актуальность, открытость, учебная ценность, кроссплатформенность, интеграция с CI/CD и соответствие требованиям безопасности РФ. Итоговый стек включает React 18 + Next.js, Node.js 20 + NestJS, PostgreSQL 15, Keycloak, Docker 24, Kubernetes 1.29, GitHub Actions и Grafana + Prometheus. Использование open-source решений минимизировало затраты и позволило студентам работать с индустриально востребованными инструментами.

Экспериментальная часть проводилась в ГБПОУ «ЮУГК» со студентами группы ИСП-219 (31 человек). До внедрения ЭУМП и после его апробации были проведены констатирующий и контрольный срезы знаний (тест по «Web-дизайну», приложение В) и оценена сформированность навыков самостоятельной проектной деятельности. Доля студентов с высоким уровнем выросла с 16 % до 39 %, средний показатель повысился с 74 % до 86 %. Средний балл лабораторных работ увеличился на 12 %, время выполнения сократилось на 25 %. Статистическая обработка (критерий χ^2 , $p < 0,05$) подтвердила значимость изменений.

Learning Analytics на базе xAPI зафиксировала рост вовлечённости: среднее число попыток решения задач снизилось, а время активной работы увеличилось. Heat-map показала, что наибольшие трудности вызывали темы «GraphQL» и «Kubernetes», что позволило оперативно скорректировать

методические материалы. Преподаватели отметили сокращение времени на проверку кода благодаря авто-тестам и встроенным чек-листам.

Экономическая эффективность проекта проявилась в отсутствии лицензионных расходов (использованы лицензии MIT, Apache, GPL) и снижении бумажных затрат. Автоматизация проверки сэкономила преподавателю в среднем 4 ч в неделю. Инфраструктура развернута на локальном сервере колледжа, что соответствует ФЗ-152 о персональных данных.

Ограничения исследования касаются выборки (одна группа) и длительности апробации (один семестр). Для обобщения результатов планируется расширение эксперимента на несколько потоков и проведение продольного анализа влияния ЭУМП на трудоустройство выпускников. Перспективы развития включают внедрение GPT-ассистентов для адаптивных подсказок, создание мобильного клиента и расширение дуальной модели за счёт новых промышленных партнёров.

Поставленные задачи выполнены: разработана классификация ЭУМП, сформулированы требования к контенту и интерфейсу, обоснован технический стек, создан и апробирован рабочий прототип, доказана его эффективность. Гипотеза исследования подтверждена: электронное учебно-методическое пособие, основанное на модульном подходе и современных веб-технологиях, является результативным средством формирования профессиональных компетенций студентов СПО.

Практическая значимость работы заключается в возможности тиражирования разработанной модели на другие дисциплины информационно-технологического цикла. Теоретическая — в развитии методологии outcome-based design, интегрированной с Learning Analytics и принципами доступности. Внедрение ЭУМП способствует повышению качества подготовки специалистов и усилению связей колледжа с IT-индустрией.

Автор выражает благодарность администрации ГБПОУ «ЮУрГТК», преподавателям-экспертам и студентам группы ИСП-219 за активное участие в проекте и ценные рекомендации, позволившие улучшить качество разработанного пособия.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Федеральный закон «Об образовании в Российской Федерации» от 29.12.2012 № 273-ФЗ (ред. 2024).
10. Подласый И.П. Педагогика: учебник. – М.: ВЛАДОС, 2007. – 576 с.
11. Sykes E. Problem-Based Learning in the 21st Century Classroom // Journal of Education. – 2023. – № 4. – С. 45-58.
12. React Documentation. URL: <https://react.dev/> (дата обращения: 25.06.2025).
13. Бабанский Ю.К. Оптимизация процесса обучения. – М.: Педагогика, 1982. – 192 с.
14. Node.js Documentation. URL: <https://nodejs.org/> (дата обращения: 25.06.2025).
15. Docker Documentation. URL: <https://docs.docker.com/> (дата обращения: 25.06.2025).
16. Лернер И.Я., Скаткин М.Н. Дидактика средней школы. – М.: Просвещение, 1982. – 224 с.
17. GitHub Actions Documentation. URL: <https://docs.github.com/actions> (дата обращения: 25.06.2025).
18. PostgreSQL 15 Documentation. URL: <https://www.postgresql.org/docs/15/> (дата обращения: 25.06.2025).
19. Беспалько В.П. Слагаемые педагогической технологии. – М.: Педагогика, 1989. – 192 с.
2. Федеральные государственные образовательные стандарты среднего профессионального образования. Укрупнённая группа 09.00.00 «Информатика и вычислительная техника». – М.: Минпросвещения России, 2024.
20. Краевский В.В., Полонский В.М. Методология педагогического исследования. – М.: Академия, 2010. – 240 с.

21. Пидкасистый П.И. Педагогика: учеб. пособие. – М.: Педагогическое общество России, 1998. – 640 с.
22. Выготский Л.С. Мышление и речь. – М.: Лабиринт, 1999. – 352 с.
23. Леонтьев А.Н. Деятельность. Сознание. Личность. – М.: Смысл, 2005. – 352 с.
24. Рубинштейн С.Л. Основы общей психологии. – СПб.: Питер, 2000. – 720 с.
25. Полат Е.С., Бухаркина М.Ю., Моисеева М.В., Петров А.Е. Современные педагогические и информационные технологии в системе образования. – М.: Академия, 2018. – 368 с.
26. Андреев А.А. Дидактические основы дистанционного обучения: монография. – М.: ИСМО РАО, 2013. – 232 с.
27. Галендухин И.С. Электронное учебно-методическое пособие «...»: репозиторий исходного кода [Электронный ресурс]. – Ver. X.Y.Z (release/tag). – URL: https://github.com/gggolddeity/yourggpu_web_dev (дата обращения: 20.08.2025).
28. Галендухин И.С. Руководство пользователя ЭУМП [Электронный ресурс]. – URL: https://github.com/gggolddeity/yourggpu_web_dev/tree/main/docs (дата обращения: 20.08.2025).
3. ГОСТ Р 7.0.97-2016. Издания электронные. Основные виды и выходные сведения.
4. ГОСТ Р 7.0.100-2018. Система стандартов по информации, библиотечному и издательскому делу. Библиографическая запись.
5. IEEE 9274.1.1-2023 (xAPI 2.0). Standard for Learning Technology — JSON Data Model and RESTful Web Service for Learner Experience Data.
6. Методические рекомендации по подготовке и защите выпускной квалификационной работы (ЮУрГГПУ).
7. Методические рекомендации по подготовке и защите ВКР (ЮУрГГПУ, направл. 44.04.01). – Текст: электронный.

8. Mayer R. E. Multimedia Learning: Third Edition. – Cambridge University Press, 2021.

9. Слостёнин В.А., Исаев И.Ф., Шиянов Е.Н. Педагогика: учеб. пособие. – М.: Академия, 2013. – 576 с.

ПРИЛОЖЕНИЯ

ПРИЛОЖЕНИЕ А

Электронная рабочая тетрадь

ПРИЛОЖЕНИЕ Б

Таблица 3 – Тематический план и содержание раздела «Web-дизайн»

Наименование разделов и тем	Содержание учебного материала, лабораторные и практические работы, самостоятельная работа студентов	Объем часов	Уровень освоения
1	2	3	4
Введение.	Содержание учебного материала	2	2
	Понятие дизайна. Что такое Web-дизайн. История развития WWW. Технологии создания Web-сайтов.		
	Самостоятельная работа студентов Подготовка рефератов по теме.	3	
Раздел 1. Основы создания Web-страниц.		20	
Тема 1.1. Основные элементы оформления HTML-страниц.	Содержание учебного материала	2	3
	Структура HTML-документа. Списки. Гиперссылки. Таблицы. Фреймы. Формы.	8	
	Лабораторные занятия 1.Создание списков и гиперссылок в HTML документах. 2. Создание таблиц в HTML документах. 3. Работа с фреймами. 4. Создание формы.		
	Самостоятельная работа студентов Работа с дополнительной литературой. Работа над творческим проектом.	4	
Тема 1.2. Работа со шрифтами.	Содержание учебного материала	2	3
	Особенности ввода и форматирование текста. Правила использования шрифтов. Структурирование текста.	2	
	Лабораторные занятия 1. Представление текстовых документов в формате HTML.		
	Самостоятельная работа студентов Работа с дополнительной литературой. Работа над творческим проектом.	2	
	Тоновая и цветовая коррекция изображений. Редактирование изображений. Создание		

	графических гиперссылок. Психология восприятия цвета. Колористика.		
Раздел 2. Графика.		18	
Тема 2.1. Помещение Web-графики на Web-страницу.	Содержание учебного материала	2	2
	Форматы графических файлов для Web. Помещение Web графики на Web страницу. Достижение баланса между текстом и графикой. Графическая информация (растровая и векторная графика). Система компьютерной презентации и мультимедийные среды		
	Лабораторные занятия 1. Создание Web страниц с графическими объектами.	2	
	Самостоятельная работа студентов Работа с дополнительной литературой. Работа над творческим проектом.	2	
Тема 2.2. Обработка изображений с помощью специальных программ.	Содержание учебного материала	2	3
	Лабораторные занятия 1. Обработка изображений в графических программах. 2. Создание сложных изображений в графических программах.	4	
	Самостоятельная работа студентов Работа с дополнительной литературой. Работа над творческим проектом.	2	
Тема 2.3. Оптимизация кода HTML для Web.	Содержание учебного материала	2	2
	Оптимизация размеров Web страниц. Основные рекомендации по использованию графики на Web страницах. Создание универсальных Web страниц.		
	Самостоятельная работа студентов Подготовка рефератов по теме 2.3	2	
Раздел 3. Анимация, звук и видео.		22	
Тема 3.1. Работа с Gif анимацией.	Содержание учебного материала	2	3
	Рекомендации по использованию анимации на Web странице. Создание Gif анимации с чистого листа. Управление Gif анимацией с помощью внутренних параметров. Создание баннеров. Построение анимации на основе текста.		
	Лабораторные занятия	6	

	1. Создание бегущей строки. 2. Создание анимированного баннера. 3. Создание Web страницы с использованием анимированных объектов.		
	Самостоятельная работа студентов Работа с дополнительной литературой. Работа над творческим проектом.	2	
Тема 3.2. Работа со звуком.	Содержание учебного материала	2	3
	Основные рекомендации по использованию звука на Web странице. Форматы звуковых файлов. Включение звука в Web страницу.		
	Лабораторные занятия 1. Создание Web страницы с использованием звуковых файлов.	2	
	Самостоятельная работа студентов Работа с дополнительной литературой. Работа над творческим проектом.	2	
Тема 3.3. Встраивание видео на Web-страницу.	Содержание учебного материала	2	3
	Рекомендации по созданию собственного видео. Монтаж видеоролика. Вопросы совместимости видео в Web. Варианты воспроизведения ролика. Виртуальные экскурсии.		
	Лабораторные занятия 1. Создание Web страницы со встраиваемым видео.	2	
	Самостоятельная работа студентов Работа с дополнительной литературой. Работа над творческим проектом.	2	
Раздел 4. Выгрузка сайта в Web его продвижение.		14	
Тема 4.1. Варианты размещения Web-сайта в сети Интернет.	Содержание учебного материала	2	2
	Виртуальный хостинг. Выделенный сервер. Совместное размещение. Размещение сайта на своем компьютере. Особенности бесплатного хостинга. Требования, предъявляемые к серверу бесплатного хостинга. Выгрузка сайта на сервер и его обновление.		
	Самостоятельная работа студентов Проработка конспекта и дополнительной литературы по вопросам к теме 4.1	2	
Тема 4.2. Раскрутка сайта	Содержание учебного материала	2	2
	Работа с поисковыми системами и каталогами. Регистрация в поисковой системе. Особенности работы поисковых систем. Способы раскрутки сайта. Завоевание популярности. Запрещенные приемы раскрутки сайта.		

	Лабораторные занятия 1. Создание сайта на предложенную тему.	4	
	Самостоятельная работа студентов Работа с дополнительной литературой. Работа над творческим проектом.	4	
Итоговое занятие.		2	3
	Содержание учебного материала	2	
	Итоговое занятие по дисциплине. Сдача зачетной работы.		
ВСЕГО:		81	

1) Какой фронтенд-фреймворк выбран в учебном пособии для реализации пользовательского интерфейса?

- а) Vue 3
- б) React 18
- в) Angular 17
- г) Svelte

2) Какой инструмент используется для сервер-сайд рендеринга и маршрутизации страниц?

- а) Next.js 14
- б) Remix
- в) Nuxt.js
- г) Create React App

3) Какая система управления базами данных является основной в примерах курса?

- а) MySQL 8
- б) MongoDB
- в) PostgreSQL 15
- г) SQLite

4) Какой сервис CI/CD интегрирован в рабочие репозитории проекта?

- а) Jenkins
- б) GitHub Actions
- в) GitLab CI
- г) Travis CI

5) Какой стек мониторинга и визуализации метрик выбран в курсе?

- а) ELK (Stack)
- б) Splunk
- в) Prometheus + Grafana

- г) Datadog

6) Какой протокол используется в примерах для реализации безопасной аутентификации пользователей?

- а) LDAP
- б) OAuth 2.0
- в) Kerberos
- г) NTLM

7) Какой стандарт доступности интерфейса соблюден при разработке пособия?

- а) WCAG 1.0
- б) Section 508
- в) WCAG 2.1 AA
- г) ISO 9241-11

8) Какая онлайн-песочница применяется для интерактивного выполнения кода фронтенда?

- а) JSFiddle
- б) CodePen
- в) StackBlitz
- г) Repl.it

9) Какой инструмент контейнеризации используется для демонстрации DevOps-процессов?

- а) Vagrant
- б) Docker
- в) Podman
- г) LXC

10) Какой инструмент статического анализа кода применяется для оценки качества проектов?

- а) Checkstyle

- б) SonarQube
- в) PMD
- г) ESLint

Ключ ответа:

1 — б

2 — а

3 — в

4 — б

5 — в

6 — б

7 — в

8 — в

9 — б

10 — б