

УТВЕРЖДЕНА

Решением Ученого совета
АНО ВО «Центральный университет»
«24» июня 2025 г.
Протокол №2

**Рабочая программа дисциплины (модуля)
«Web-разработка»**

Направление подготовки: 02.03.01 Математика и компьютерные науки

Направленность (профиль) подготовки: Математика и компьютерные науки

Квалификация (степень) выпускника: бакалавр

Форма обучения: очная

Срок освоения программы: 4 года

Год набора: 2025

**Москва
2025**

Содержание

1. Краткая характеристика дисциплины (модуля)	3
2. Перечень планируемых результатов обучения	4
3. Тематический план	6
4. Содержание дисциплины (модуля)	6
5. Учебно-методическое обеспечение	7
6. Материально-техническое обеспечение	8
7. Методические и оценочные материалы	9

1. Краткая характеристика дисциплины (модуля)

Рабочая программа дисциплины (модуля) «Web-разработка» составлена в соответствии с федеральным государственным образовательным стандартом высшего образования – бакалавриат по специальности 02.03.01 Математика и компьютерные науки, профиль Математика и компьютерные науки, утвержденный приказом Министерства науки и высшего образования Российской Федерации № 807 от 23.08.2017 года.

Изучение дисциплины (модуля) «Web-разработка» позволяет освоить навыки создания современных, интерактивных и адаптивных веб-приложений, востребованных в цифровой экономике. Это обеспечивает фундамент для понимания клиент-серверных технологий, что критично для разработки эффективных и масштабируемых интернет-решений.

Место дисциплины (модуля) в структуре образовательной программы

Настоящая дисциплина (модуль) включена в учебный план по программе подготовки бакалавриата по направлению 02.03.01 Математика и компьютерные науки, профиль Математика и компьютерные науки и входит в вариативную часть Блока 1, формируемую участниками образовательных отношений.

Дисциплина (модуль) является выборной и доступна для изучения на 3 или 4 курсе в 5, 6 или 7 семестре на выбор.

Цель изучения дисциплины (модуля): формирование практических навыков проектирования, создания и поддержки современных веб-приложений с использованием актуальных технологий и стандартов.

Задачи изучения дисциплины (модуля):

— изучить основы языков разметки и программирования, таких как HTML, CSS и JavaScript, для построения структуры, стиля и интерактивности веб-страниц;

— освоить современные фреймворки и инструменты разработки, включая React, Node.js и системы управления версиями, для эффективного создания масштабируемых веб-приложений;

— понять принципы пользовательского интерфейса, безопасности, оптимизации производительности и адаптивного дизайна, чтобы разрабатывать надежные и пользователь-friendly веб-решения

В результате освоения дисциплины (модуля) обучающийся должен:

знать:

— используемые для веб-разработки языки программирования JavaScript и TypeScript;

— внутренние представления страниц в браузере и взаимодействия браузера с сервером;

— концепции асинхронного программирования;

— механизмы авторизации и сеансов;

уметь:

— проектировать и разрабатывать пользовательские интерфейсы для сервисов;

— реализовывать полноценные приложения;

— тестировать пользовательский интерфейс и автоматизировать этот процесс;

владеть:

— навыком реализации сложного интерфейса Web-приложения.

2. Перечень планируемых результатов обучения

Компетенции, формируемые в результате освоения дисциплины (модуля) при проведении учебных занятий в форме контактной работы обучающихся с педагогическими работниками Университета и в форме самостоятельной работы обучающихся:

Компетенция	Содержание компетенции	Индикатор компетенции	Перечень планируемых результатов обучения по дисциплине (модулю)
УК-1.	Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.1.	Знает методы поиска и анализа информации в области разработки, основные принципы критической оценки источников информации и их релевантности
		УК-1.2.	Умеет критически оценивать источники информации и синтезировать данные из различных источников для решения задач, применять системный подход к анализу и решению комплексных проблем
		УК-1.3.	Имеет практический опыт работы с современными инструментами и технологиями для обработки информации, формулировании и структурировании задач на основе полученной информации
ОПК-1.	Способен консультировать и использовать фундаментальные знания в области математического анализа, комплексного и функционального анализа алгебры, аналитической геометрии, дифференциальной геометрии и топологии, дифференциальных уравнений, дискретной математики и математической логики, теории вероятностей, математической статистики и случайных процессов, численных методов, теоретической механики в профессиональной деятельности	ОПК-1.1.	Знает основные концепции и теории в области математического анализа и смежных дисциплин; методы и подходы, используемые в различных областях математики
		ОПК-1.2.	Умеет применять математические методы для решения профессиональных задач
		ОПК-1.3.	Имеет практический опыт разработки и реализации математических моделей в профессиональной деятельности
ОПК-4.	Способен находить, анализировать, реализовывать программно и использовать на практике математические алгоритмы, в том числе с применением современных вычислительных систем	ОПК-4.1.	Знает базовые основы современного математического аппарата, связанного с проектированием, разработкой, реализацией и оценкой качества программных продуктов и программных комплексов в различных областях человеческой деятельности

		ОПК-4.2.	Умеет использовать этот математический аппарат в профессиональной деятельности
		ОПК-4.3.	Имеет практический опыт применения современного математического аппарата, связанного с проектированием, разработкой, реализацией и оценкой качества программных продуктов и программных комплексов в различных областях человеческой деятельности
ОПК-6.	Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения	ОПК-6.1.	Знает алгоритмы разработки, компьютерные программы, а также алгоритмы вычислительной математики
		ОПК-6.2.	Умеет разрабатывать математические программные продукты и комплексы с использованием современных технологий программирования
		ОПК-6.3.	Имеет практический опыт разработки интеллектуальных информационных систем для визуализации результатов исследований
ПК-3.	Способен использовать методы математического и алгоритмического моделирования при решении теоретических и прикладных задач	ПК-3.1.	Знает основные методы математического и алгоритмического моделирования, а также их применение для решения теоретических и прикладных задач
		ПК-3.2.	Умеет разрабатывать и применять математические модели и алгоритмы для решения различных задач, анализируя полученные результаты
		ПК-3.3.	Имеет практический опыт использования методов математического и алгоритмического моделирования в реальных проектах или исследованиях

3. Тематический план

№п/п	Наименование раздела дисциплины (модуля)	Трудоемкость, академические часы					ТКУ (текущий контроль успеваемости)
		<i>Очная форма</i>					
		Контактная работа			Контроль	Самостоятельная работа	
Лекции	Семинары	Консультации					
1	Технологии веб-разработки	8	8	6		38	Домашние задания
2	Архитектура и фреймворки	8	8	7		38	Домашние задания
3	Тестирование, безопасность	8	8	7	2	40	Домашние задания, Проект
	<i>Зачет с оценкой</i>				4		
	Итого:	24	24	20	6	116	
	<i>Объем дисциплины (модуля) (в ак. ч.)</i>	190					
	<i>Объем дисциплины (модуля) (в зач. ед.)</i>	5					

4. Содержание дисциплины (модуля)

№п/п	Наименование раздела дисциплины (модуля)	Содержание дисциплины (модуля) по темам
1	Технологии веб-разработки	Введение в веб-разработку и основы семантической верстки. Погружение в JavaScript. Инструменты разработки и сборки. TypeScript. Проектная неделя
2	Архитектура и фреймворки	Архитектура веб-приложений и компонентный подход. Построение SPA и основы Web Components. Введение в React Продвинутый React и управление состоянием. Клиент-серверное взаимодействие и API-интеграция. Проектная неделя
3	Тестирование, безопасность	Тестирование веб-приложений. Безопасность веб-приложений и отладка. Проектная неделя

5. Учебно-методическое обеспечение

Университет располагает полным набором лицензионного и свободно распространяемого программного обеспечения, включая продукты отечественного производства.

Каждый студент в течение всего периода обучения получает индивидуальный неограниченный доступ к электронно-библиотечной системе и электронной информационно-образовательной среде университета. Эти системы предоставляют возможность доступа к ресурсам из любой точки, где есть подключение к сети Интернет, как на территории университета, так и за его пределами.

Студентам обеспечен удаленный доступ к современным профессиональным базам данных и информационным справочным системам.

Основная литература:

1. Полуэктова Н. Р. Разработка веб-приложений : учебник для вузов / Н. Р. Полуэктова. — 2-е изд. — Москва : Издательство Юрайт, 2025. — 204 с. — (Высшее образование). — ISBN 978-5-534-18645-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/567610>.

2. Тузовский А. Ф. Проектирование и разработка web-приложений : учебник для вузов / А. Ф. Тузовский. — Москва : Издательство Юрайт, 2025. — 219 с. — (Высшее образование). — ISBN 978-5-534-16300-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/561176>.

3. Черный, Б. Профессиональный TypeScript. Разработка масштабируемых JavaScript-приложений / Б. Черный. - Санкт-Петербург : Питер, 2021. - 352 с. - ISBN 978-5-4461-1651-5. - Текст : электронный. - URL: <https://znanium.ru/catalog/product/2142477>.

4. Бэнкс, А. React: современные шаблоны для разработки приложений : практическое руководство / А. Бэнкс, Е. Порселло. - 2-е изд. - Санкт-Петербург : Питер, 2022. - 320 с. - (Бестселлеры O'Reilly). - ISBN 978-5-4461-1492-4. - Текст : электронный. - URL: <https://znanium.com/catalog/product/2123877>.

5. Вандеркам, Д. Эффективный TypeScript: 62 способа улучшить код : практическое руководство / Д. Вандеркам. - Санкт-Петербург : Питер, 2021. - 288 с. - (Серия «Бестселлеры O'Reilly»). - ISBN 978-5-4461-1623-2. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1733517>.

6. Фримен, Э. Изучаем программирование на JavaScript : практическое руководство / Э. Фримен, Э. Робсон. - Санкт-Петербург : Питер, 2021. - 640 с. - (Серия «Head First O'Reilly»). - ISBN 978-5-4461-0893-0. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1739643>.

Дополнительная литература:

1. Сысолетин Е. Г. Разработка интернет-приложений : учебник для вузов / Е. Г. Сысолетин, С. Д. Ростунцев ; под научной редакцией Л. Г. Доросинского. — Москва : Издательство Юрайт, 2025. — 80 с. — (Высшее образование). — ISBN 978-5-534-17124-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/562916>.

2. Web-разработки в asp. Net web forms : учебник для вузов / С. Т. Гуляева, В. В. Миронов, Н. О. Котелина, И. И. Лавреш. — Москва : Издательство Юрайт, 2025. — 134 с. — (Высшее образование). — ISBN 978-5-534-19885-0. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/569218>.

6. Материально-техническое обеспечение

Университет располагает материально-технической базой, соответствующей действующим противопожарным правилам и нормам и обеспечивающей проведение всех видов дисциплинарной и междисциплинарной подготовки, практической и научно-исследовательской работ обучающихся, предусмотренных учебным планом.

Помещения, которые представляют собой учебные аудитории для проведения занятий лекционного типа, занятий семинарского (практического) типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы и помещения для хранения и профилактического обслуживания учебного оборудования. Помещения укомплектованы специализированной мебелью и техническими средствами обучения, служащими для представления учебной информации большой аудитории.

Изучение дисциплины (модуля) обеспечивается в учебных аудиториях, оснащенных:

- столами и стульями;
- компьютерной техникой;
- механическими калькуляторами;
- специализированным оборудованием, включая демонстрационное оборудование.

Помещения для самостоятельной работы обучающихся, в том числе приспособленные для использования инвалидами и лицами с ограниченными возможностями здоровья, оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду Университета.

Обучающимся предоставляется доступ (в том числе удаленный) к ресурсам информационно-телекоммуникационной сети «Интернет», электронным ресурсам (в том числе электронным библиотечным системам, современным профессиональным базам данных и информационным справочным системам):

№	Наименование портала (издания, курса, документа)	Ссылка
1.	Научная электронная библиотека elibrary.ru библиотека	https://elibrary.ru/defaultx.asp
2.	База данных для IT-специалистов	https://habr.com
3.	База данных ScienceDirect	https://www.sciencedirect.com
4.	Официальный сайт Министерства науки и высшего образования Российской Федерации	https://minobrnauki.gov.ru/
5.	Федеральный портал «Российское образование»	https://www.edu.ru/
6.	Информационная система "Единое окно доступа к образовательным ресурсам"	http://window.edu.ru/
7.	Единая коллекция цифровых образовательных ресурсов	http://school-collection.edu.ru/
8.	Федеральный центр информационно - образовательных ресурсов	http://fcior.edu.ru/

Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), в том числе комплект лицензионного программного обеспечения, современные профессиональные базы данных и информационные справочные системы:

Наименование ПО	Производство	Лицензионное / свободно распространяемое
Операционные системы:		
Microsoft Imagine (Windows Client, Server)	зарубежное	лицензионное

Браузеры:		
Яндекс.Браузер	отечественное	свободно распространяемое
Google Chrome	зарубежное	свободно распространяемое
Офисные приложения:		
Microsoft Imagine (Visio, OneNote)	зарубежное	лицензионное
TeXstudio	зарубежное	свободно распространяемое
Adobe Acrobat Reader	зарубежное	свободно распространяемое
Программное обеспечение для планирования и учета времени:		
Toggle app	зарубежное	свободно распространяемое
Системы управления проектами:		
Microsoft Imagine (Project)	зарубежное	лицензионное
Системы управления базами данных:		
Microsoft Imagine (SQL Server)	зарубежное	лицензионное
Системы резервного копирования (backup):		
Acronis Backup Advanced for HyperV	зарубежное	лицензионное
Справочно-правовые системы:		
КонсультантПлюс: справочно-правовая система	отечественное	лицензионное
Средства антивирусной защиты:		
Kaspersky Endpoint Security для бизнеса Стандартный Russian Edition	отечественное	лицензионное
Среды разработки:		
Visual Studio Code	зарубежное	свободно распространяемое
Bash (Unix shell)	зарубежное	свободно распространяемое
Anaconda	зарубежное	свободно распространяемое
Robotic Operating System	зарубежное	свободно распространяемое
CopelliaSim	зарубежное	свободно распространяемое
Google Colaboratory	зарубежное	свободно распространяемое
Пакеты программных средств и библиотек:		
AutoPsy	зарубежное	свободно распространяемое
Interactive Disassembler (IDA)	зарубежное	свободно распространяемое
Системы управления библиографической информацией:		
Zotero	зарубежное	свободно распространяемое
Сервисы и службы:		
Bind	зарубежное	свободно распространяемое
Docker	зарубежное	свободно распространяемое

7. Методические и оценочные материалы

Методические указания для обучающихся по освоению дисциплины (модуля)

В процессе изучения дисциплины (модуля) «Web-разработка» в рамках текущего контроля успеваемости используются такие виды учебной работы, как лекции, семинары, консультации, домашние задания, проект, а также различные виды самостоятельной работы обучающихся по заданию преподавателя, направленные на развитие навыков профессиональной лексики, закрепление практических профессиональных компетенций, поощрение инициатив.

Лекция – систематическое, последовательное, монологическое изложение преподавателем учебного материала, как правило, теоретического характера.

В процессе лекций рекомендуется вести конспект лекций: кратко и схематично фиксировать основные идеи, выводы и обобщения лекции; выделять важные мысли, ключевые слова и термины. Необходимо отметить вопросы или материалы, которые вызывают затруднения, и попытаться найти ответы в рекомендованной литературе. Если разобраться в материале не удастся, следует сформулировать вопрос и задать его

преподавателю на консультации или во время семинарского (практического) занятия.

Участие в семинаре (аудиторная работа) – активная работа студента на семинаре, его ответы на вопросы преподавателя и участие в дискуссии.

Для успешного участия в семинаре студентам рекомендуется заранее ознакомиться с темой обсуждения, прочитать необходимые материалы и подготовить вопросы. Важно активно слушать и вовлекаться в дискуссию, высказывая свои мнения и аргументируя их. При ответах на вопросы преподавателя стоит быть уверенным, четким и логичным, опираясь на изученный материал. Также полезно поддерживать диалог с однокурсниками, чтобы обогатить обсуждение и расширить свои знания.

Консультации – структурированные встречи, на которых преподаватели предоставляют индивидуальную или групповую помощь в освоении учебного материала, обсуждении вопросов и решении проблем, возникающих в процессе обучения.

Консультации могут включать разъяснение сложных тем, подготовку к экзаменам и помощь в выполнении курсовых работ, что способствует более глубокому пониманию предмета и улучшению академической успеваемости.

Домашнее задание – набор задач по темам недели.

При работе над домашними заданиями важно внимательно ознакомиться с требованиями и сроками выполнения. Рекомендуется разбивать задания на этапы, чтобы избежать перегрузки и лучше усвоить материал. Использовать различные источники информации, включая учебники и онлайн-ресурсы, для более глубокого понимания темы.

Проект – исследовательская работа по курсу и презентация результатов.

Для успешной подготовки к проекту: четко определите цели и задачи проекта, распределите роли и обязанности между участниками, а также установите сроки выполнения каждой части работы. Регулярно проводите встречи для обсуждения прогресса и решения возникающих вопросов.

Самостоятельная работа – работа студентов, направленная на углубленное изучение отдельных тем и вопросов учебной дисциплины (модуля).

В процессе самостоятельной работы студенты взаимодействуют с рекомендованными материалами при минимальном участии преподавателя. Задачи студента включают работу с конспектами лекций (обработка текста), повторное изучение учебных материалов, планов и тезисов ответов, изучение дополнительных тем, выполнение учебно-исследовательских заданий и другое.

Система оценивания результатов обучения по дисциплине (модулю)

Критерии получения уровня и оценивания сформированности компетенций по дисциплине (модулю) «Web-разработка»

Оценивание уровня учебных достижений, обучающихся по дисциплине (модулю), осуществляется в виде текущего контроля успеваемости и промежуточной аттестации.

Промежуточная аттестация по дисциплине (модулю) осуществляется в форме **зачета с оценкой**, при этом проводится оценка компетенций, сформированных по дисциплине.

Для оценивания текущего контроля успеваемости и промежуточной аттестации используется десятибалльная шкала оценивания, которая соотносится с традиционной пятибалльной шкалой следующим образом:

Десятибалльная оценка	Пятибалльная оценка	Оценка за зачет	Общая характеристика результата обучения по дисциплине (модулю)
10	Отлично	Зачтено	<p>Студент полностью владеет знаниями, изложенными в рабочей программе, и глубоко осмысляет дисциплину. Он самостоятельно и логически последовательно отвечает на все вопросы, акцентируя внимание на наиболее важном. Умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя ключевые моменты и устанавливая причинно-следственные связи. Четко формулирует ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные задачи. Студент хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты дисциплины (модуля) с практическими задачами.</p>
9	Отлично	Зачтено	
8	Отлично	Зачтено	
7	Хорошо	Зачтено	<p>Студент обладает знаниями предмета почти в полном объеме рабочей программы и самостоятельно, логически последовательно и всесторонне отвечает на все вопросы, акцентируя внимание на наиболее значимых моментах. Он умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя его ключевые аспекты и устанавливая причинно-следственные связи. Формулирует свои ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные ситуационные задачи. Студент хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты предмета с практическими задачами.</p>
6	Хорошо	Зачтено	
5	Удовлетворительно	Зачтено	<p>Студент обладает базовыми знаниями по дисциплине, но испытывает трудности при самостоятельных ответах и использует неточные формулировки. В ходе ответов он допускает ошибки,</p>
4	Удовлетворительно	Зачтено	

Десятибалльная оценка	Пятибалльная оценка	Оценка за зачет	Общая характеристика результата обучения по дисциплине (модулю)
			касающиеся сути вопросов. Студент способен решать только самые простые задачи и владеет лишь минимальным набором методов исследования.
3	Не сдан	Не зачтено	Студент не овладел обязательным минимумом знаний по предмету и не может ответить на вопросы, даже если преподаватель задает дополнительные наводящие вопросы.
2	Не сдан	Не зачтено	
1	Не сдан	Не зачтено	

Дисциплина (модуль) «Web-разработка» оценивается следующим образом:

Активность	Вес	Описание
Домашние задания	20%	Набор задач по темам недели
Проект	30%	Презентация результатов исследовательской работы по дисциплине (модулю)
Зачет с оценкой	50%	Письменная или устная работа над заданием, направленным на проверку полученных знаний и навыков по курсу

Формула расчёта итоговой оценки по дисциплине (модулю) «Web-разработка»:
« $0,2 \times$ среднее за домашние задания + $0,3 \times$ за проект + $0,5 \times$ зачет с оценкой».

В рамках изучения дисциплины (модуля) возможно получение бонусных баллов.

Текущий контроль успеваемости обучающихся по дисциплине (модулю)

Примерные домашние задания

Домашнее задание «Технологии веб-разработки»

1. **Семантическая верстка:** Создайте простую веб-страницу, используя семантические теги HTML5 (например, <header>, <nav>, <article>, <section>, <footer>). Объясните, почему вы выбрали именно эти теги.

2. **JavaScript:** Напишите скрипт, который изменяет текст в элементе <h1> на вашей веб-странице при нажатии на кнопку. Используйте метод `addEventListener`.

3. **Инструменты разработки:** Ознакомьтесь с инструментами разработчика в вашем браузере (например, Chrome DevTools). Сделайте скриншоты использования вкладок "Elements" и "Console" и напишите краткое описание их функционала.

4. **TypeScript:** Напишите небольшой проект на TypeScript, который включает функцию, принимающую два числа и возвращающую их сумму. Убедитесь, что вы используете строгую типизацию.

5. **Проектная неделя:** Опишите идею для своего проекта на следующую неделю. Какие технологии вы планируете использовать и как будет выглядеть структура вашего проекта?

Домашнее задание «Архитектура и фреймворки»

1. **Компонентный подход:** Создайте простой компонент на React, который отображает список задач. Компонент должен принимать массив задач в качестве свойства (props).

2. **SPA:** Разработайте одностраничное приложение (SPA) с использованием React Router. Создайте как минимум две страницы: "Главная" и "Контакты". Реализуйте навигацию между ними.

3. **Web Components:** Создайте кастомный элемент (Web Component), который отображает кнопку. При нажатии на кнопку должен выводиться алерт с текстом "Кнопка нажата!".

4. **Управление состоянием:** Используя React, создайте приложение для управления счетчиком (increment/decrement). Используйте useState для управления состоянием.

5. **API-интеграция:** Напишите функцию на JavaScript, которая делает запрос к публичному API (например, JSONPlaceholder) и выводит список пользователей на вашей веб-странице.

Домашнее задание «Тестирование и безопасность»

1. **Тестирование веб-приложений:** Напишите юнит-тест для функции, которая проверяет, является ли строка палиндромом. Используйте библиотеку Jest для написания тестов.

2. **Безопасность веб-приложений:** Изучите основные уязвимости веб-приложений (например, XSS, CSRF, SQL Injection) и напишите краткий отчет о том, как защититься от каждой из них.

3. **Отладка:** Создайте простой веб-приложение на JavaScript и намеренно внесите в него ошибку. Используйте инструменты разработчика для отладки и исправления ошибки.

4. **Тестирование интерфейса:** Напишите тест для вашего React-компонента, используя библиотеку Testing Library. Убедитесь, что компонент правильно отображает переданные ему свойства.

5. **Проектная неделя:** Опишите, как вы будете тестировать ваше приложение, которое вы разрабатываете на следующей неделе. Какие подходы к тестированию вы планируете использовать (юнит-тесты, интеграционные тесты и т.д.)?

Примерное описание и критерии оценивания к итоговому проекту

Описание итогового проекта:

В рамках итогового проекта студенты разрабатывают полнофункциональное frontend-приложение в формате Single Page Application (SPA), демонстрирующее применение изученных технологий веб-разработки. Проект включает использование TypeScript для типизации, модульную структуру кода, работу с пакетами, взаимодействие с DOM и браузерными API, создание React-компонентов, управление состояниями и навигацию без перезагрузки страницы. В приложении реализуется динамическая загрузка данных с использованием асинхронных запросов и WebSocket, а также интеграция со сторонними сервисами через API. Для безопасности настраивается HTTPS с SSL-сертификатом. Проект сопровождается автоматизированными UI-тестами, покрывающими ключевые

пользовательские сценарии.

Структура проекта и задачи студентов:

1. Архитектура и настройка проекта

- Создать структуру проекта с разделением на модули и компоненты.
- Настроить сборку и управление пакетами (npm/yarn).
- Подключить TypeScript и настроить конфигурацию.

2. Разработка интерфейса

- Реализовать React-компоненты с использованием типизации.
- Организовать навигацию между страницами без перезагрузки (React Router или аналог).
- Управлять состояниями приложения (React Context, Redux или аналог).

3. Работа с данными

- Реализовать асинхронную загрузку данных с сервера через REST API и WebSocket.
- Обеспечить обработку ошибок и загрузочных состояний.

4. Интеграция с внешними сервисами

- Настроить взаимодействие с внешними API.
- Обеспечить корректное отображение и обновление данных.

5. Безопасность и развертывание

- Настроить HTTPS с использованием SSL-сертификата.
- Подготовить проект к запуску в защищённой среде.

6. Тестирование

- Написать автоматизированные UI-тесты для проверки основных пользовательских сценариев.

Критерии оценивания:

- Реализация основных функций и требований проекта.
- Использование TypeScript с корректной типизацией.
- Чистая и модульная организация кода.
- Рабочая навигация и управление состояниями.
- Корректная загрузка данных и работа с API.
- Настройка HTTPS с SSL.
- Наличие автоматических UI-тестов.
- Удобство интерфейса и качество пользовательского опыта.
- Документация с инструкциями по запуску и описанием проекта.

Задания для промежуточной аттестации по дисциплине (модулю)

№ п/п	Задание	Ответ	Компетенция
1	Укажите основной язык для клиентской логики в веб-разработке.	JavaScript (варианты: JS, Java Script)	УК-1
2	Укажите инструмент для сборки проектов в веб-разработке.	Webpack (варианты: Webpack, сборщик)	УК-1
3	Укажите тип языка для типизации в JavaScript.	TypeScript (варианты: TS, Type Script)	УК-1
4	Укажите подход для семантической верстки HTML.	Семантические теги (варианты: semantic markup, семантическая разметка)	УК-1
5	Укажите математический метод для сортировки массивов в JavaScript.	Сортировка (варианты: sort, алгоритм сортировки)	ОПК-1
6	Укажите структуру данных для хранения пар ключ-значение в JS.	Объект (варианты: object, объект)	ОПК-1

7	Укажите концепцию для работы с асинхронностью в JS.	Promise (варианты: promise, промис)	ОПК-1
8	Укажите алгоритм поиска элементов в массиве.	Линейный поиск (варианты: linear search, линейный)	ОПК-4
9	Укажите метод оптимизации производительности в React.	Мемоизация (варианты: memoization, мемо)	ОПК-4
10	Укажите операцию для объединения данных в API.	Fetch (варианты: fetch, асинхронный запрос)	ОПК-4
11	Укажите алгоритм для рендеринга компонентов в React.	Виртуализация (варианты: virtualization, виртуальный DOM)	ОПК-4
12	Укажите функцию для создания компонента в React.	useState (варианты: useState, хук состояния)	ОПК-6
13	Укажите метод для тестирования компонентов.	Jest (варианты: Jest, фреймворк тестирования)	ОПК-6
14	Укажите протокол для клиент-серверного взаимодействия.	HTTP (варианты: HTTP, протокол)	ОПК-6
15	Укажите инструмент для отладки в браузере.	DevTools (варианты: DevTools, инструменты разработчика)	ОПК-6
16	Укажите механизм для управления состоянием в SPA.	Redux (варианты: Redux, менеджер состояния)	ОПК-6
17	Укажите модель для архитектуры веб-приложений.	MVC (варианты: MVC, модель-представление-контроллер)	ПК-3
18	Укажите подход к моделированию компонентов в React.	Компонентный подход (варианты: component-based, компонентная архитектура)	ПК-3
19	Укажите метод моделирования безопасности в веб.	CORS (варианты: CORS, cross-origin resource sharing)	ПК-3
20	Укажите алгоритмическую модель для тестирования приложений.	Unit-тестирование (варианты: unit testing, модульное тестирование)	ПК-3