МИНОБРНАУКИ РОССИИ
ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
Колледж информационных и креативных технологий

РАБОЧАЯ ПРОГРАММА

МДК.03.04 Разработка веб-приложений (React)

ПМ.03 Проектирование и разработка веб-приложений

программы подготовки специалистов среднего звена
специальность
09.02.11 Разработка и управление программным обеспечением

Форма обучения - очная

Владивосток 2026

Рабочая программа МДК.03.04 Разработка веб-приложений (React) разработана на основе федерального государственного образовательного стандарта среднего профессионального образования по специальности 09.02.11 Разработка и управление программным обеспечением, квалификация программист, утвержденного приказом Министерства просвещения Российской Федерации от 24 февраля 2025 г. № 138

Разработчик(и): Д.А. Трофимов, преподаватель

Рассмотрено и одобрено на заседании цикловой методической комиссии
Протокол № 9 от «20» мая 2026 г.

Председатель ПЦК _____ П.В. Калашников

СОДЕРЖАНИЕ

1 ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

2 СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

3 УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ

4 КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

1. ОБЩАЯ ХАРАКТЕРИСТИКА ПРОГРАММЫ МЕЖДИСЦИПЛИНАРНОГО КУРСА

1.1. Цель и место междисциплинарного курса в структуре образовательной программы

Междисциплинарный курс МДК.03.04 Разработка веб-приложений (React) входит в профессиональный цикл учебного плана образовательной программы в структуру профессионального модуля ПМ.03 «Проектирование и разработка веб-приложений».

Цель: освоение принципов компонентной архитектуры, методов и современных инструментов библиотеки React для проектирования и разработки надежных, масштабируемых и интерактивных веб-приложений.

1.2. Планируемые результаты освоения междисциплинарного курса

По итогам освоения курса, обучающиеся должны продемонстрировать результаты обучения, соотнесённые с результатами освоения ООП СПО, приведенные в таблице.

Код	Наименование видов деятельности и профессиональных компетенций
ВД 3	Проектирование и разработка веб-приложений
ПК 3.1	Разрабатывать техническое задание на веб-приложение в соответствии с требованием заказчика
ПК 3.2	Разрабатывать веб-приложение в соответствии с техническим заданием
ПК 3.4	Производить тестирование веб-приложений
ПК 3.5	Осуществлять аудит безопасности веб-приложения в соответствии с регламентом по безопасности

В результате освоения междисциплинарного курса обучающийся должен:

Код ПК	Уметь	Знать	Владеть навыками
ПК 3.1	Проводить анкетирование и интервьюирование для выявления требований заказчика. Оформлять техническую документацию в соответствии с нормами и стандартами. Осуществление выбора одного из типовых решений по разработке веб-приложений. Работы со специализированным программным обеспечением для планирования времени и организации работы с клиентами.	Инструменты и методы выявления требований заказчика к веб-приложению. Типовые решения по разработке веб-приложений. Нормы и стандарты оформления технической документации. Принципы проектирования и разработки информационных систем.	Сбора предварительных данных для выявления требований к веб-приложению. Определения первоначальных требований заказчика к веб-приложению и возможности их реализации. Подбора оптимальных вариантов реализации задач и согласование их с заказчиком. Разработки технического задания на веб-приложение в соответствии с требованиями заказчика.
ПК 3.2	Разрабатывать программный код клиентской и серверной части веб-приложений. Использовать язык разметки страниц веб-приложения. Оформлять код программы в соответствии со	Языки программирования и разметки для разработки клиентской и серверной части веб-приложений. Принципы работы объектной модели веб-приложений и браузера. Основы технологии клиент-	Выполнения верстки страниц веб приложений. Кодирования на языках веб программирования. Разработки базы данных. Умения использовать специальные готовые технические решения при разработке веб приложений.

	стандартом кодирования. Использовать открытые библиотеки и фреймворки. Использовать выбранную среду программирования и средства системы. Управлять базами данных. Осуществлять взаимодействие клиентской и серверной частей веб приложений. Разрабатывать код информационных систем. Разрабатывать программный код клиентской части веб-приложений. Оформлять код программы в соответствии со стандартом кодирования. Использовать объектные модели веб приложений и браузера. Разрабатывать анимацию для веб приложений для повышения его доступности и визуальной привлекательности. Использовать основные принципы дизайна интерфейса пользователя и управления стилями, предоставляемыми наборами. Использовать готовые компоненты и стили для эффективной и быстрой разработки интерфейса. Способность адаптировать и настраивать стили и элементы управления для достижения желаемого визуального эффекта и соответствия дизайну	сервер. Технологии разработки серверной части. Особенности отображения веб приложений в размерах рабочего пространства устройств. Особенности отображения элементов ИР в различных браузерах. Особенности выбранной среды программирования и системы управления базами данных. Языки программирования и разметки для разработки клиентской и серверной части веб приложений. Принципы работы объектной модели веб-приложений и браузера. Технологии для разработки анимации. Способы манипуляции элементами страницы веб-приложения. Виды анимации и способы ее применения. Знакомство с существующими наборами стилей, такими как Bootstrap, Foundation, Material UI и другие. Понимание основных концепций и возможностей предоставляемых наборами стилей и элементов управления. Знание CSS и JavaScript для настройки и расширения стилей и элементов управления в выбранных наборах	Выполнения разработки информационных систем. Разработки интерфейса пользователя. Разработки анимационных эффектов. Разработки интерфейсов пользователя, используя существующие наборы стилей, такие как Bootstrap или Foundation, для создания привлекательного и согласованного визуального оформления. Применения предустановленных элементов управления, таких как кнопки, формы, меню и т.д., предоставляемых в выбранных наборах стилей. Адаптации и настройки стилей и элементов управления с использованием CSS и JavaScript
ПК 3.4	Выполнять отладку и тестирование программного кода (в	Сетевые протоколы и основы web-технологий. Современные методики	Использования инструментальных средств контроля версий и баз

	том числе с использованием инструментальных средств). Выполнять оптимизацию и рефакторинг программного кода. Кодировать на скриптовых языках программирования. Тестировать веб-приложения с использованием тест-планов. Применять инструменты подготовки тестовых данных. Выбирать и комбинировать техники тестирования веб-приложений. Работать с системами контроля версий в соответствии с регламентом использования системы контроля версий. Выполнять проверку веб-приложения по техническому заданию	тестирования. Эргономику пользовательских интерфейсов. Основные принципы отладки и тестирования программных продуктов. Методы организации работы при проведении процедур тестирования. Возможности используемой системы. Контроль версий и вспомогательных инструментальных программных средств для обработки исходного текста программного кода. Регламент использования системы контроля версий. Предметную область проекта для составления тест-планов	данных, учета дефектов. Тестирования веб-приложений с точки зрения логической целостности. Тестирования интеграции веб-приложения с внешними сервисами и учетными системами
ПК 3.5	Осуществлять аудит безопасности веб приложений. Модифицировать веб-приложение с целью внедрения программного кода по обеспечению безопасности его работы. Способность проводить аудит безопасности веб-приложений, используя различные инструменты и методы, такие как сканирование уязвимостей, тестирование на проникновение и анализ кода. Анализировать полученные результаты аудита и тестирования на проникновение для определения уязвимостей и рисков безопасности. Предоставлять отчеты и рекомендации по улучшению	Источники угроз информационной безопасности и меры по их предотвращению. Регламенты и методы разработки безопасных веб-приложений. Различные инструменты и методы для проведения аудита безопасности веб-приложений, такие как сканеры уязвимостей (Nessus, OpenVAS), инструменты тестирования на проникновение (Metasploit, Burp Suite) и анализ кода (SonarQube). Основные уязвимости и риски безопасности веб-приложений, такие как инъекции, межсайтовый скриптинг, подделка запросов между сайтами и т.д. Знание методов и рекомендаций по устранению уязвимостей и повышению безопасности веб-	Обеспечения безопасной и бесперебойной работы. Осуществления аудита безопасности веб-приложения в соответствии с регламентом по безопасности. Идентификации потенциальных уязвимостей и рисков безопасности веб-приложения. Проведения тестирования на проникновение для проверки уровня защиты веб-приложения. Анализ полученных результатов аудита и тестирования на проникновение для определения слабых мест и рекомендаций по их устранению

	безопасности веб-приложений на основе проведенного аудита.	приложений на основе результатов аудита.	

2. СТРУКТУРА И СОДЕРЖАНИЕ МЕЖДИСЦИПЛИНАРНОГО КУРСА

2.1 Объем МДК и виды учебной работы

Вид учебной работы	Объем в часах	В т.ч. в форме практ. подготовки
Учебные занятия	42	28
Самостоятельная работа	20	
Консультации	2	
Промежуточная аттестация	4	
Всего	**68**	**28**

2.2. Тематический план и содержание МДК

Наименование разделов и тем	Содержание учебного материала и формы организации деятельности обучающихся	Объем в часах	Коды компетенций, формированию которых способствует элемент программы
Раздел 1. Проектирование веб-приложений и разработка технического задания		**20**	**ПК 3.1, ПК 3.2**
Тема 1.1. Анализ требований заказчика и разработка технического задания	**Содержание учебных материалов**	**2**	
	Жизненный цикл веб-приложений и методологии разработки (Waterfall, Agile, Scrum, Kanban).	1	ПК 3.1
	Сбор, анализ и классификация требований заказчика: бизнес-требования, пользовательские, функциональные и нефункциональные требования.	1	ПК 3.1
	В том числе практических занятий	**4**	
	Сбор и формализация функциональных и нефункциональных требований по брифу заказчика.	2	ПК 3.1
	Разработка и оформление технического задания на создание веб-приложения.	2	ПК 3.1
Тема 1.2. Моделирование предметной области и проектирование архитектуры веб-приложения	**Содержание учебных материалов**	**3**	
	Основы проектирования клиент-серверных веб-приложений и принципы компонентного подхода (React).	1	ПК 3.1, ПК 3.2
	Применение унифицированного языка моделирования UML (диаграммы прецедентов Use Case, последовательности, компонентов и развертывания) на основе ТЗ.	1	ПК 3.1, ПК 3.2

	Проектирование информационной архитектуры веб-приложения, пользовательских сценариев (User Flow) и иерархии компонентов интерфейса.	1	ПК 3.1, ПК 3.2
	В том числе практических занятий	**6**	
	Построение диаграмм вариантов использования (Use Case) и последовательности (Sequence Diagram) на основе технического задания.	2	ПК 3.1, ПК 3.2
	Разработка схемы информационной архитектуры, структурных прототипов (Wireframes) и дерева компонентов веб-приложения.	4	ПК 3.1, ПК 3.2
	В том числе самостоятельная работа обучающихся Построение диаграммы компонентов и диаграммы развертывания проектируемого веб-приложения по техническому заданию.	**5**	**ПК 3.1, ПК 3.2**
Раздел 2. Разработка, тестирование и аудит безопасности веб-приложений		**42**	**ПК 3.2, ПК 3.4, ПК 3.5**
Тема 2.1. Архитектура и разработка веб-приложений в соответствии с техническим заданием	**Содержание учебных материалов**	**5**	
	Клиент-серверная архитектура современных веб-приложений (SPA, SSR) и организация структуры проекта (React).	1	ПК 3.2
	Разработка пользовательских интерфейсов: функциональные компоненты, передача данных (Props), локальное состояние и хуки (State, Hooks).	1	ПК 3.2
	Маршрутизация в веб-приложениях (React Router), реализация форм и валидация пользовательского ввода в соответствии с ТЗ.	1	ПК 3.2
	Архитектурные паттерны (MVC, MVVM, Flux) и глобальное управление состоянием веб-приложения (Context API, менеджеров состояния).	1	ПК 3.2
	Проектирование и реализация взаимодействия с серверным окружением (REST API, GraphQL, Fetch/Axios, WebSockets).	1	ПК 3.2
	В том числе практических занятий	**6**	
	Разработка компонентной архитектуры, маршрутизации и управления состоянием веб-приложения в соответствии с ТЗ.	4	ПК 3.2
	Реализация клиент-серверного взаимодействия с REST API (отправка запросов, обработка и отображение JSON-данных).	2	ПК 3.2

	В том числе самостоятельная работа обучающихся Разработка интерактивных компонентов, форм и модулей веб-приложения под заданное техническое задание.	**5**	**ПК 3.2**
Тема 2.2. **Тестирование и контроль качества веб-приложений**	**Содержание учебных материалов**	**2**	
	Виды, уровни и методы тестирования веб-приложений: модульное (Unit), интеграционное, сквозное (E2E) и регрессионное тестирование.	1	ПК 3.4
	Функциональное и нефункциональное тестирование веб-приложения на соответствие ТЗ (тестирование UI/UX, кроссбраузерности и производительности).	1	ПК 3.4
	В том числе практических занятий	**6**	
	Разработка тест-кейсов и проведение модульного и компонентного тестирования веб-приложения.	4	ПК 3.4
	Проведение функционального и интеграционного тестирования веб-приложения, оформление отчетов о дефектах (баг-репортов).	2	ПК 3.4
	В том числе самостоятельная работа обучающихся Составление плана тестирования (Test Plan) и чек-листов проверки веб-приложения на соответствие техническому заданию.	**5**	**ПК 3.4**
Тема 2.3. **Аудит безопасности и защита веб-приложений**	**Содержание учебных материалов**	**2**	
	Регламенты безопасности и классификация основных уязвимостей веб-приложений (OWASP Top 10: XSS, CSRF, инъекции, небезопасная конфигурация).	1	ПК 3.5
	Архитектурные механизмы защиты веб-приложений: безопасная аутентификация и авторизация (JWT, OAuth), настройка политик CORS, CSP и шифрования (HTTPS).	1	ПК 3.5
	В том числе практических занятий	**6**	
	Проведение аудита безопасности веб-приложения и его зависимостей на наличие типовых уязвимостей (XSS, CSRF, уязвимости пакетов) по регламенту.	4	ПК 3.5
	Настройка механизмов защиты веб-приложения (экранирование данных, безопасное хранение токенов, настройка заголовков безопасности) по результатам аудита.	2	ПК 3.5
	В том числе самостоятельная работа обучающихся Составление отчета по результатам аудита безопасности веб-приложения и разработка плана устранения	**5**	**ПК 3.5**

	уязвимостей в соответствии с регламентом.		
Консультации		**2**	**ПК 3.1, ПК 3.2, ПК 3.4, ПК 3.5**
Промежуточные аттестации		**4**	**ПК 3.1, ПК 3.2, ПК 3.4, ПК 3.5**
Всего		**68**	

3. УСЛОВИЯ РЕАЛИЗАЦИИ МЕЖДИСЦИПЛИНАРНОГО КУРСА

3.1. Материально-техническое обеспечение

Для реализации программы МДК.03.04 «Разработка веб-приложений (React)» предусмотрено наличие следующих помещений:

Лаборатория организации и принципов построения информационных систем.

Основное оборудование: Компьютеризированные рабочие места обучающихся с выходом в сеть "Интернет" - 20 шт., стулья - 20 шт., комплект мебели (стол и стул) для преподавателя - 1 шт., графические станции ПК №2 WorkStation WS 917 20 шт., автоматизированное рабочее место преподавателя ПК №2 WorkStation WS 917, графический планшет Wacom - 20 шт., мультимедийное оборудование - 1 шт., телевизор, экран, многофункциональное устройство (МФУ) формат А3 -1 шт., доска маркерная - 1 шт.

Программное обеспечение: PyCharm, Visual Studio Code, IntelliJ IDEA, JetBrains Rider, Python (Anaconda3) с TensorFlow/PyTorch, Microsoft Visio, Draw.io, Figma, Figma, Adobe Illustrator, MySQL Workbench, HeidiSQL, MySQL, PostgreSQL, Git, Docker Desktop, Microsoft Project, Draw.io, Microsoft SQL Server Management Studio, Microsoft Office.

Основное оборудование: Рабочие места на базе компьютерной техники с возможностью подключения к информационно-телекоммуникационной сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду ВВГУ: комплекты учебной мебели (столы и стулья) – 19 шт., персональные компьютеры (облачные мониторы) - 19 шт; доска маркерная - 1шт., лазерный копир-принтер-сканер с 2 лотками Xerox WorkCentre 3345 DNI.

Программное обеспечение: Microsoft Windows 7 Professional; Microsoft Office ProPlus 2010 Russian Acdmc; СПС КонсультантЮрист: Версия Проф; Adobe Acrobat Reader; Google Chrome; Adobe Flash Player; 7-Zip 18.01 (x64).

3.2. Учебно-методическое обеспечение

Для реализации программы учебной дисциплины библиотечный фонд ВВГУ укомплектован печатными и электронными изданиями.

Обучающиеся из числа инвалидов и лиц с ограниченными возможностями здоровья обеспечены печатными и (или) электронными образовательными ресурсами в формах, адаптированных к ограничениям их здоровья.

Основные источники

1. Проектирование информационных систем : учебник и практикум для среднего профессионального образования / Д. В. Чистов, П. П. Мельников, А. В. Золотарюк, Н. Б. Ничепорук. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 273 с. — (Профессиональное образование). — ISBN 978-5-534-20362-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/562355

2. Грекул, В. И. Проектирование информационных систем : учебник и практикум для среднего профессионального образования / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — 2-е изд. — Москва : Издательство Юрайт, 2025. — 404 с. — (Профессиональное образование).
— ISBN 978-5-534-19506-4. — Текст : электронный // Образовательная платформа Юрайт

[сайт]. — URL: https://urait.ru/bcode/566739

3. Соколова, В. В. Разработка мобильных приложений : учебное пособие для среднего профессионального образования / В. В. Соколова. — Москва : Издательство Юрайт, 2024. —160 с. — (Профессиональное образование). — ISBN 978-5-534-16868-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/542342

Дополнительные источники

1. Черпаков, И. В. Основы программирования : учебник и практикум для среднего профессионального образования / И. В. Черпаков. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 196 с. — (Профессиональное образование). — ISBN 978-5-534- 18760-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/561922

2. Зыков, С. В. Объектно-ориентированное программирование : учебник и практикум для вузов / С. В. Зыков. — 2-е изд. — Москва : Издательство Юрайт, 2024. — 151 с. — (Высшее образование). — ISBN 978-5-534-16941-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/537385

3. Станкевич, Л. А. Интеллектуальные системы и технологии : учебник и практикум для среднего профессионального образования / Л. А. Станкевич. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 478 с. — (Профессиональное образование). — ISBN 978-5-534-20364-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/566524

Электронные ресурсы

1. Информационно-справочная система «Консультант Плюс» – http://www.consultant.ru/
2. Профессиональная база данных: "Открытая база ГОСТов"/ Режим доступа: http://standartgost.ru/, доступ свободный
3. Электронная библиотечная система «РУКОНТ» - Режим доступа: https://lib.rucont.ru/
4. Электронная библиотечная система ZNANIUM.COM - Режим доступа: https://znanium.com/
5. Электронно-библиотечная система издательства "Юрайт" - Режим доступа: https://urait.ru/
6. Электронно-библиотечная система издательства "Лань" - Режим доступа: https://e.lanbook.com/
7. Президентская библиотека им. Б.Н. Ельцина (база данных различных профессиональных областей) Режим доступа: https://www.prlib.ru/, доступ свободный

4 МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ИЗУЧЕНИЮ И РЕАЛИЗАЦИИ МЕЖДИСЦИПЛИНАРНОГО КУРСА

Начиная изучение дисциплины, студенту необходимо:

- ознакомиться с программой, изучить список рекомендуемой литературы;
- внимательно разобраться в структуре курса, в системе распределения учебного материала по видам занятий, формам контроля, чтобы иметь представление о курсе в целом.

Самостоятельная работа студентов (СРС) — это деятельность учащихся, которую они совершают без непосредственной помощи и указаний преподавателя, руководствуясь сформировавшимися ранее представлениями о порядке и правильности выполнения операций. Цель СРС в процессе обучения заключается, как в усвоении знаний, так и в формировании умений и навыков по их использованию в новых условиях на новом учебном материале. Самостоятельная работа призвана обеспечивать возможность осуществления студентами самостоятельной познавательной деятельности в обучении, и является видом учебного труда, способствующего формированию у студентов самостоятельности. Кроме

того, для расширения и углубления знаний по дисциплине целесообразно использовать: публикации в тематических журналах; полнотекстовые базы данных библиотеки; имеющиеся в библиотеках вуза. Успешное освоение дисциплины предполагает активную работу студентов на всех занятиях аудиторной формы: лекций и практических занятий, выполнение аттестационный мероприятий, эффективную самостоятельную работу.

4.1 Методические рекомендации обучающимся по обеспечению самостоятельной работы

Самостоятельная работа проводится с целью: систематизации, закрепления, углубления и расширения теоретических знаний и практических умений, приобретаемых студентами в ходе аудиторных занятий; формирования умений использовать специальную литературу; развития познавательных способностей и активности обучающихся; формирования самостоятельности мышления, способности к саморазвитию, самосовершенствованию и самореализации; развития исследовательских умений.

Для лучшего освоения материала и систематизации знаний по дисциплине необходимо отвечать на вопросы для самоконтроля. Во время самостоятельной проработки лекционного материала особое внимание следует уделять возникшим вопросам, непонятным терминам, спорным точкам зрения. В случае необходимости обращаться к преподавателю за консультацией. Студент должен четко уяснить, что именно с лекции начинается его подготовка к практическому занятию. Вместе с тем, лекция лишь организует мыслительную деятельность, но не обеспечивает глубину усвоения программного материала.

При подготовке к практическому занятию особое внимание необходимо обратить на содержание основных положений и выводов, объяснение явлений и фактов, уяснение практического приложения рассматриваемых теоретических вопросов. В процессе этой работы студент должен стремиться понять и запомнить основные положения рассматриваемого материала, примеры, поясняющие его, а также разобраться в иллюстративном материале. В процессе подготовки рекомендуется взаимное обсуждение материала, во время которого закрепляются знания, а также приобретается практика в изложении и разъяснении полученных знаний, развивается речь.

При необходимости следует обращаться за консультацией к преподавателю. Перед консультацией, необходимо хорошо продумать вопросы, которые требуют разъяснения.

В процессе изучения дисциплины студенту необходимо ориентироваться на самостоятельную проработку лекционного материала, подготовку к практическим занятиям, выполнение тестов, кейсовых заданий, самостоятельное изучение некоторых разделов курса. Для проведения занятий лекционного типа используются учебно-наглядные пособия в форме презентационных материалов, обеспечивающих тематические иллюстрации, соответствующие темам лекций.

4.2 Особенности организации обучения для лиц с ограниченными возможностями здоровья и инвалидов

При необходимости обучающимся из числа лиц с ограниченными возможностями здоровья и инвалидов (по заявлению обучающегося) предоставляется учебная информация в доступных формах с учетом их индивидуальных психофизических особенностей:

для лиц с нарушениями зрения: в печатной форме увеличенным шрифтом; в форме электронного документа; индивидуальные консультации с привлечением тифлосурдопереводчика; индивидуальные задания, консультации и др.

для лиц с нарушениями слуха: в печатной форме; в форме электронного документа; индивидуальные консультации с привлечением сурдопереводчика; индивидуальные задания, консультации и др.

для лиц с нарушениями опорно-двигательного аппарата: в печатной форме; в форме электронного документа; индивидуальные задания, консультации и др.

5 КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ КУРСА

Код ПК	Критерии оценки результата (показатели освоенности компетенций)	Формы контроля и методы оценки
ПК.3.1	Оценка «отлично»: техническое задание и архитектурные модели разработаны в полном соответствии с требованиями заказчика и стандартами оформления. Оценка «хорошо»: техническое задание разработано в соответствии с требованиями заказчика, но содержит незначительные неточности в формулировках требований или схемах моделирования. Оценка «удовлетворительно»: техническое задание разработано с частичным соответствием требованиям заказчика, имеются пробелы в описании структуры.	Экспертная оценка разработанного технического задания, защита отчётов по практическим работам и устное собеседование.
ПК.3.2	Оценка «отлично»: веб-приложение разработано в полном соответствии с ТЗ, функционал работает. Оценка «хорошо»: веб-приложение и его компоненты разработаны в соответствии с ТЗ, но имеются минимальные недочёты в реализации логики или взаимодействия модулей. Оценка «удовлетворительно»: веб-приложение разработано с существенными отступлениями от ТЗ, реализован только базовый функционал.	Защита отчётов по практическим работам, демонстрация разработанного веб-приложения и проверка соответствия программных компонентов требованиям ТЗ.
ПК.3.6	Оценка «отлично»: тестирование веб-приложения выполнено в полном объёме, тест-кейсы, автотесты и баг-репорты оформлены корректно. Оценка «хорошо»: тестирование веб-приложения выполнено в соответствии с методикой, допускаются мелкие недочёты в покрытии тестами или оформлении тестовой документации. Оценка «удовлетворительно»: тестирование веб-приложения выполнено частично, проверены только базовые пользовательские сценарии.	Проверка выполнения практических работ по тестированию веб-приложения, оценка разработанных тест-кейсов и баг-репортов, контроль на зачёте / экзамене.
ПК.3.7	Оценка «отлично»: аудит безопасности веб-приложения проведён в полном соответствии с регламентом по безопасности, уязвимости выявлены и устранены. Оценка «хорошо»: аудит безопасности и настройка механизмов защиты выполнены в соответствии с регламентом, но имеются мелкие неточности в отчёте или конфигурации политик безопасности. Оценка «удовлетворительно»: аудит безопасности веб-приложения и настройка средств защиты реализованы частично или с ошибками.	Проверка практических заданий по проведению аудита безопасности, защита отчёта о выявленных уязвимостях и плана повышения защищённости веб-приложения.

МИНОБРНАУКИ РОССИИ
ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
Колледж информационных и креативных технологий

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

для проведения текущего контроля и промежуточной аттестации

по **МДК 03.04 Разработка веб-приложений (React)**

ПМ.03 Проектирование и разработка веб-приложений

программы подготовки специалистов среднего звена
09.02.11 Разработка и управление программным обеспечением

Форма обучения: очная

Владивосток 2026

1. Общие сведения

Фонд оценочных средств (далее – ФОС) предназначены для контроля и оценки образовательных достижений обучающихся, освоивших программу МДК.03.04 «Разработка веб-приложений (React)» в соответствии с техническим заданием и оценка их качества

ФОС включают в себя контрольные материалы для проведения текущего контроля успеваемости и промежуточной аттестации по дисциплине, которая проводится в форме дифференцированного зачёта (с использованием оценочного средства - устный опрос в форме ответов на вопросы билетов, устный опрос в форме собеседования, выполнение письменных заданий, тестирование и т.д.).

2. Планируемые результаты обучения, обеспечивающие результаты освоения образовательной программы

Код ПК	Код результата обучения	Наименование результата обучения
ПК 3.1	З1	Методологии и этапы жизненного цикла разработки веб-приложений, методы сбора, анализа и классификации требований заказчика.
	З2	Стандарты, структуру и правила оформления технического задания (ТЗ) и спецификаций требований (SRS) на создание веб-приложений.
	З3	Нотации и инструменты визуального моделирования архитектуры и бизнес-процессов (UML-диаграммы, User Flow, Wireframes).
	У1	Проводить предпроектное обследование, собирать и формализовывать функциональные и нефункциональные требования заказчика.
	У2	Строить диаграммы прецедентов (Use Case), последовательности и пользовательские сценарии для проектируемой системы.
	У3	Разрабатывать и оформлять техническое задание на веб-приложение в соответствии с требованиями заказчика и отраслевыми стандартами.
	П1	Сбора, анализа и документирования требований заказчика к проектируемому веб-приложению.
	П2	Разработки технического задания и построения концептуальных моделей информационной архитектуры системы.
ПК 3.2	З1	Основные архитектурные подходы к построению веб-приложений (SPA, SSR, клиент-серверная и компонентная архитектура React).
	З2	Принципы разработки пользовательских интерфейсов (функциональные компоненты, состояние State, хуки Hooks, маршрутизация) и архитектурные паттерны (MVC, MVVM, Flux).
	З3	Принципы проектирования и реализации клиент-серверного взаимодействия и программных интерфейсов (REST API, GraphQL, Fetch/Axios, WebSockets).
	У1	Проектировать компонентную структуру, маршрутизацию и схему управления состоянием веб-приложения в соответствии с требованиями технического задания.
	У2	Разрабатывать интерактивные пользовательские интерфейсы, формы с валидацией и функциональные модули веб-приложения (React).
	У3	Реализовывать клиент-серверное взаимодействие веб-приложения с внешними сервисами и API согласно техническому заданию.
	П1	Проектирования и разработки компонентной архитектуры и клиентской части веб-приложений по техническому заданию.
	П2	Интеграции веб-приложения с серверным окружением по API и реализации управления состоянием приложения.
ПК 3.4	З1	Виды, уровни и методы тестирования веб-приложений (модульное Unit, интеграционное, сквозное E2E и регрессионное тестирование).

Код ПК	Код результата обучения	Наименование результата обучения
	З2	Методики функционального и нефункционального тестирования веб-приложений (проверка UI/UX, кроссбраузерности, производительности) на соответствие ТЗ.
	З3	Инструменты автоматизированного тестирования веб-приложений и компонентов (Jest, React Testing Library, Cypress), правила оформления тест-кейсов и баг-репортов.
	У1	Составлять планы тестирования (Test Plan), чек-листы и тест-кейсы для проверки функциональности веб-приложения по техническому заданию.
	У2	Разрабатывать и выполнять модульные, компонентные и интеграционные тесты для проверки корректности работы веб-приложения.
	У3	Выявлять дефекты в работе веб-приложения, проводить отладку кода и оформлять отчёты об ошибках (баг-репорты).
	П1	Проведения ручного и автоматизированного (модульного, компонентного, интеграционного) тестирования веб-приложений.
	П2	Разработки тестовой документации (тест-кейсов, чек-листов, баг-репортов) и контроля качества веб-приложения на соответствие ТЗ.
ПК 3.5	З1	Регламенты обеспечения безопасности веб-приложений и классификацию основных уязвимостей (OWASP Top 10: XSS, CSRF, инъекции, уязвимости зависимостей).
	З2	Архитектурные механизмы защиты веб-приложений (безопасная аутентификация JWT/OAuth, политики CORS и CSP, HTTPS) и методики проведения аудита безопасности.
	У1	Проводить аудит безопасности исходного кода, клиент-серверного взаимодействия и сторонних зависимостей (npm audit) веб-приложения в соответствии с регламентом.
	У2	Настраивать механизмы защиты веб-приложения (экранирование данных, безопасное хранение токенов, заголовки безопасности) и оформлять отчёт по результатам аудита.
	П1	Проведения аудита безопасности веб-приложений и выявления уязвимостей в соответствии с регламентом по безопасности.
	П2	Технической реализации механизмов защиты информации и устранения выявленных уязвимостей веб-приложения.

3. Описание процедуры оценивания

Результаты обучения по модулю, уровень сформированности компетенций оцениваются по четырёх бальной шкале оценками: «отлично», «хорошо», «удовлетворительно», «неудовлетворительно» (по бальной системе. Максимальная сумма баллов по дисциплине равна 100 баллов.)

Текущая аттестация по модулю проводится с целью систематической проверки достижений обучающихся. Объектами оценивания являются: степень усвоения теоретических знаний, уровень овладения практическими умениями и навыками по всем видам учебной работы, качество выполнения самостоятельной работы, учебная дисциплина (активность на занятиях, своевременность выполнения различных видов заданий, посещаемость всех видов занятий по аттестуемой дисциплине).

При проведении промежуточной аттестации оценивается достижение студентом запланированных по модулю результатов обучения, обеспечивающих результаты освоения образовательной программы в целом. Оценка на зачете / экзамене выставляется с учетом оценок, полученных при прохождении текущей аттестации.

Критерии оценивания устного ответа

(оценочные средства: контрольная работа, собеседование, устное сообщение.)

5 баллов - ответ показывает прочные знания основных процессов изучаемой предметной области, отличается глубиной и полнотой раскрытия темы; владение терминологическим аппаратом; умение объяснять сущность, явлений, процессов, событий, делать выводы и обобщения, давать аргументированные ответы, приводить примеры; свободное владение монологической речью, логичность и последовательность ответа; умение приводить примеры современных проблем изучаемой области.

4 балла - ответ, обнаруживающий прочные знания основных процессов изучаемой предметной области, отличается глубиной и полнотой раскрытия темы; владение терминологическим аппаратом; умение объяснять сущность, явлений, процессов, событий, делать выводы и обобщения, давать аргументированные ответы, приводить примеры; свободное владение монологической речью, логичность и последовательность ответа. Однако допускается одна - две неточности в ответе.

3 балла – ответ, свидетельствующий в основном о знании процессов изучаемой предметной области, отличающийся недостаточной глубиной и полнотой раскрытия темы; знанием основных вопросов теории; слабо сформированными навыками анализа явлений, процессов, недостаточным умением давать аргументированные ответы и приводить примеры; недостаточно свободным владением монологической речью, логичностью и последовательностью ответа. Допускается несколько ошибок в содержании ответа; неумение привести пример развития ситуации, провести связь с другими аспектами изучаемой области.

2 балла – ответ, обнаруживающий незнание процессов изучаемой предметной области, отличающийся неглубоким раскрытием темы; незнанием основных вопросов теории, несформированными навыками анализа явлений, процессов; неумением давать аргументированные ответы, слабым владением монологической речью, отсутствием логичности и последовательности. Допускаются серьезные ошибки в содержании ответа; незнание современной проблематики изучаемой области.

Критерии оценивания письменной работы

(оценочные средства: реферат, конспект, контрольная работа, письменный отчет по практической работе, портфолио, доклад (сообщение), в том числе выполненный в форме презентации, творческое задание, курсовая работа).

5 баллов - студент выразил своё мнение по сформулированной проблеме, аргументировал его, точно определив ее содержание и составляющие. Проблема раскрыта полностью, выводы обоснованы. Приведены данные отечественной и зарубежной литературы, статистические сведения, информация нормативно-правового характера. Студент владеет навыком самостоятельной работы по заданной теме; методами и приемами анализа теоретических и/или практических аспектов изучаемой области. Фактических ошибок, связанных с пониманием проблемы, нет; графически работа оформлена правильно.

4 балла - работа характеризуется смысловой цельностью, связностью и последовательностью изложения; допущено не более 1 ошибки при объяснении смысла или содержания проблемы. Проблема раскрыта. Не все выводы сделаны и/или обоснованы. Для аргументации приводятся данные отечественных и зарубежных авторов. Продемонстрированы исследовательские умения и навыки. Фактических ошибок, связанных с пониманием проблемы, нет. Допущены одна-две ошибки в оформлении работы.

3 балла – студент проводит достаточно самостоятельный анализ основных этапов и смысловых составляющих проблемы; понимает базовые основы и теоретическое обоснование выбранной темы. Проблема раскрыта не полностью. Выводы не сделаны и/или выводы не обоснованы Проведен анализ проблемы без привлечения дополнительной литературы. Допущено не более 2 ошибок в смысле или содержании проблемы, оформлении работы.

2 балла - работа представляет собой пересказанный или полностью переписанный исходный текст без каких бы то ни было комментариев, анализа. Не раскрыта структура и теоретическая составляющая темы. Проблема не раскрыта. Выводы отсутствуют. Допущено три или более трех ошибок в смысловом содержании раскрываемой проблемы, в оформлении работы.

Критерии оценивания тестового задания

Оценка	Отлично	Хорошо	Удовлетворительно	Неудовлетворительно
Количество правильных ответов	91 % и ≥	от 81% до 90,9 %	не менее 70%	менее 70%

Критерии выставления оценки студенту на зачете/ экзамене

(оценочные средства: устный опрос в форме ответов на вопросы билетов, устный опрос в форме собеседования, выполнение письменных разноуровневых задач и заданий, комплексная расчетно-графическая работа, творческое задание, кейс-задача, портфолио, проект и т.п.)

Оценка по промежуточной аттестации	Характеристика качества сформированности компетенций
«зачтено» / «отлично»	Студент демонстрирует сформированность дисциплинарных компетенций на продвинутом уровне: обнаруживает всестороннее, систематическое и глубокое знание учебного материала, умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний, причем не затрудняется с ответом при видоизменении заданий, правильно обосновывает принятое решение, владеет разносторонними навыками и приемами выполнения практических задач.
«зачтено» / «хорошо»	Студент демонстрирует сформированность дисциплинарных компетенций на базовом уровне: основные знания, умения освоены, но допускаются незначительные ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации.
«зачтено» / «удовлетворительно»	Студент демонстрирует сформированность дисциплинарных компетенций на пороговом уровне: имеет знания только основного материала, но не усвоил его деталей, в ходе контрольных мероприятий допускаются значительные ошибки, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала, испытывает затруднения при выполнении практических работ, при оперировании знаниями и умениями при их переносе на новые ситуации.
«не зачтено» / «неудовлетворительно»	Студент демонстрирует сформированность дисциплинарных компетенций на уровне ниже порогового: выявляется полное или практически полное отсутствие знаний значительной части программного материала, студент допускает существенные ошибки, неуверенно, с большими затруднениями выполняет практические работы, умения и навыки не сформированы.

4. Примеры оценочных средств для проведения промежуточной аттестации

Вопросы для дифференцированного зачета

1. Что такое архитектура веб-приложения? Дайте определение и назовите её основные задачи.

2. Какие основные этапы входят в жизненный цикл проектирования и разработки веб-приложения (Waterfall, Agile, Scrum)?
3. В чём разница между функциональными и нефункциональными требованиями к системе? Приведите по три примера каждого вида.
4. Какие методы используются для сбора и выявления требований заказчика на предпроектном этапе?
5. Что такое техническое задание (ТЗ) и зачем оно нужно как заказчику, так и команде разработчиков?
6. Какие обязательные разделы входят в структуру технического задания на разработку веб-приложения по ГОСТ / SRS?
7. Что такое пользовательская история (User Story) и по какому шаблону она формулируется?
8. Зачем при проектировании архитектуры используется язык UML? Назовите основные виды UML-диаграмм.
9. Что отображает диаграмма прецедентов (Use Case) и какие базовые элементы в ней используются?
10. Как с помощью диаграммы последовательности (Sequence Diagram) описать процесс взаимодействия клиента, сервера и базы данных?
11. В чём заключается разница между диаграммой компонентов (Component Diagram) и диаграммой развертывания (Deployment Diagram)?
12. Что такое информационная архитектура веб-приложения и зачем разрабатываются пользовательские сценарии (User Flow)?
13. Зачем нужны вайрфреймы (Wireframes), интерактивные прототипы и дерево компонентов на этапе согласования ТЗ с заказчиком?
14. Как устроена клиент-серверная архитектура? В чём отличие двухуровневой модели от трёхуровневой?
15. В чём разница между одностраничными приложениями (SPA), серверным рендерингом (SSR) и статической генерацией (SSG)?
16. Что представляет собой компонентная архитектура веб-приложения (на примере React) и каковы правила декомпозиции интерфейса?
17. Что такое функциональные компоненты в React, чем отличаются входные свойства (Props) от локального состояния (State)?
18. Как работает виртуальное дерево документа (Virtual DOM) в React и за счёт чего оно повышает производительность интерфейса?
19. Назначение и правила использования базовых хуков React (useState, useEffect, useContext, useRef, useMemo).
20. Как организуется клиентская маршрутизация в одностраничных веб-приложениях с помощью React Router?
21. Как реализуются управляемые и неуправляемые формы в React и какими способами выполняется валидация пользовательского ввода по ТЗ?
22. В чём заключаются ключевые различия между архитектурными паттернами MVC, MVVM и однонаправленным потоком данных (Flux / Redux)?
23. Как решается проблема «прокидывания пропсов» (Props Drilling) с помощью Context API и глобальных менеджеров состояния?
24. Что такое API и какую роль программные интерфейсы играют во взаимодействии клиентской и серверной частей веб-приложения?
25. Каким базовым принципам должен соответствовать веб-сервис в стиле REST? Какие HTTP-методы и коды состояния в нём используются?
26. Как в React-приложении организовать асинхронные запросы к серверу (Fetch API, Axios) и обработку состояний загрузки и ошибок?
27. В чём разница между подходами REST и GraphQL? В каких задачах GraphQL оказывается эффективнее?

28. Зачем в веб-приложениях используется протокол WebSockets и чем он отличается от обычных HTTP-запросов?
29. Как принципы SOLID и модульная организация кода применяются при разработке масштабируемых веб-приложений?
30. Каковы основные цели тестирования веб-приложений? В чём разница между верификацией и валидацией продукта по ТЗ?
31. Что представляет собой «пирамида тестирования»? Опишите назначение модульного (Unit), интеграционного и сквозного (E2E) тестирования.
32. Что такое функциональное и нефункциональное тестирование веб-приложения (тестирование UI/UX, кроссбраузерности, адаптивности, нагрузки)?
33. Из каких обязательных полей состоит тест-кейс (Test Case) и чем он отличается от чек-листа проверки?
34. Как правильно оформить отчёт о дефекте (баг-репорт)?В чём разница между серьёзностью (Severity) и приоритетом (Priority) ошибки?
35. Какие инструменты используются для модульного и компонентного тестирования React-приложений (Jest, React Testing Library)?
36. Зачем при тестировании компонентов используются заглушки и имитации (Mocks, Stubs) сетевых запросов к API?
37. Что такое регрессионное тестирование и как автоматизировать сквозное (E2E) тестирование пользовательских сценариев в браузере (Cypress / Playwright)?
38. Что включает в себя регламент по безопасности веб-приложения и каковы основные этапы проведения аудита безопасности?
39. Что такое стандарт OWASP Top 10? Назовите наиболее критические уязвимости современных веб-приложений.
40. Что такое межсайтовый скриптинг (XSS), какие виды XSS существуют и как защитить от них React-приложение?
41. В чём суть атаки межсайтовой подделки запросов (CSRF) и какими архитектурными методами (CSRF-токены, SameSite Cookies) она предотвращается?
42. Как возникают уязвимости инъекций (SQL-инъекции, NoSQL и командные инъекции) и как обеспечить безопасную валидацию и санитизацию данных?
43. Как организовать безопасную аутентификацию и авторизацию в веб-приложении с использованием JWT-токенов и протокола OAuth 2.0? Где безопасно хранить токены на клиенте?
44. Зачем нужны политики безопасности браузера CORS (Cross-Origin Resource Sharing) и заголовки CSP (Content Security Policy)?
45. Как проводится аудит безопасности исходного кода (SAST) и проверка сторонних пакетов и зависимостей проекта (npm audit) на известные уязвимости (CVE)?

Экзаменационные билеты

Билет 1

1. Классификация требований к веб-приложению: бизнес-требования, пользовательские, функциональные и нефункциональные.
2. По предложенному описанию проекта (интернет-магазин электроники) сформулируйте по 5 функциональных и нефункциональных требований для технического задания.

Билет 2

1. Структура и стандарты оформления технического задания (ТЗ) на разработку веб-приложения.

2. Разработайте фрагмент технического задания на модуль «Личный кабинет пользователя» (назначение, входные/выходные данные, описание функциональных сценариев).

Билет 3

1. Применение языка UML при проектировании веб-систем. Элементы и правила построения диаграммы прецедентов (Use Case).
2. Постройте Use Case диаграмму для веб-приложения онлайн-записи к врачу (роли: Пациент, Врач, Администратор) на основе требований ТЗ.

Билет 4

1. Моделирование динамического поведения системы: диаграмма последовательности (Sequence Diagram) и пользовательские сценарии (User Flow).
2. Разработайте схему процесса авторизации и восстановления пароля пользователя в виде Sequence-диаграммы (Клиент — Сервер приложения — База данных — Почтовый сервис).

Билет 5

1. Проектирование информационной архитектуры веб-приложения, создание прототипов (Wireframes) и декомпозиция интерфейса на компоненты.
2. Спроектируйте схему информационной архитектуры и дерево иерархии React-компонентов для главной страницы образовательного портала по техническому заданию.

Билет 6

1. Клиент-серверная и компонентная архитектура современных веб-приложений (SPA, SSR).
2. Разработайте структурную схему клиент-серверного веб-приложения системы бронирования билетов и опишите распределение логики между клиентом (React) и сервером.

Билет 7

1. Основы разработки интерфейсов на React: функциональные компоненты, входные свойства (Props) и управление локальным состоянием (State, хук useState).
2. Разработайте React-компонент интерактивной карточки товара со счётчиком количества и кнопкой добавления в корзину в соответствии с ТЗ.

Билет 8

1. Жизненный цикл компонента и работа с побочными эффектами в React (хук useEffect). Асинхронная загрузка данных с сервера.
2. Реализуйте React-компонент списка пользователей, который при монтировании получает данные по REST API (через fetch/axios), обрабатывая состояния загрузки (Loading) и ошибки (Error).

Билет 9

1. Маршрутизация в одностраничных веб-приложениях (React Router): динамические маршруты, вложенные страницы и защищённые роуты (Private Routes).
2. Спроектируйте и опишите конфигурацию маршрутизации веб-приложения (Главная, Каталог, Страница товара по ID, Профиль с проверкой авторизации, страница 404).

Билет 10

1. Архитектурные паттерны управления состоянием веб-приложения: MVC, MVVM, Flux и использование Context API.
2. Спроектируйте архитектуру глобального состояния веб-приложения (переключение темы оформления и хранение данных текущего пользователя) с использованием React Context API.

Билет 11

1. Архитектурный стиль REST и принципы проектирования RESTful API для взаимодействия клиента и сервера.
2. Разработайте спецификацию REST API (методы HTTP, URL-эндпоинты, примеры тела запроса/ответа в JSON и коды статусов) для CRUD-операций над сущностью «Заказ».

Билет 12

1. Работа с формами в веб-приложениях: управляемые компоненты в React и валидация пользовательского ввода на соответствие ТЗ.
2. Разработайте компонент формы регистрации пользователя с клиентской валидацией полей (Email, пароль, подтверждение пароля) и выводом сообщений об ошибках.

Билет 13

1. Виды и уровни тестирования веб-приложений: модульное (Unit), интеграционное, сквозное (E2E) и регрессионное тестирование.
2. Разработайте набор из 5 позитивных и негативных тест-кейсов (Test Cases) для проверки модуля «Корзина покупок» на соответствие техническому заданию.

Билет 14

1. Функциональное и нефункциональное тестирование веб-приложений. Правила фиксации дефектов и оформления баг-репортов.
2. Проанализируйте описание сбоя в работе веб-формы оформления заказа и составьте полный баг-репорт (шаги воспроизведения, фактический и ожидаемый результат, Severity, Priority).

Билет 15

1. Автоматизированное тестирование компонентов веб-приложения с использованием Jest и React Testing Library.
2. Напишите модульный/компонентный тест для проверки корректного рендеринга и обработки клика по кнопке в компоненте формы авторизации.

Билет 16

1. Методика тестирования клиент-серверного взаимодействия и REST API (проверка статус-кодов, структуры JSON, граничных значений и времени отклика).
2. Составьте чек-лист тестирования API-эндпоинта создания нового товара (POST /api/products) с проверкой валидации данных и прав доступа.

Билет 17

1. Регламент обеспечения безопасности веб-приложений и классификация основных уязвимостей по стандарту OWASP Top 10.
2. Проведите аудит предложенного фрагмента кода веб-приложения, выявите уязвимость межсайтового скриптинга (XSS) и перепишите код с соблюдением правил безопасного экранирования.

Билет 18

1. Безопасность аутентификации и авторизации в веб-приложениях: использование JWT-токенов, защита маршрутов и безопасное хранение данных на клиенте.
2. Проведите аудит схемы авторизации SPA-приложения, хранящего токены в localStorage, опишите риски безопасности и спроектируйте защищённую схему с использованием HttpOnly Cookies.

Билет 19

1. Механизмы защиты клиент-серверного взаимодействия: предотвращение CSRF-атак, настройка политик CORS и заголовков безопасности CSP.
2. Спроектируйте конфигурацию политик CORS и заголовков Content-Security-Policy (CSP) для защиты веб-приложения от несанкционированных кросс-доменных запросов и загрузки внешних скриптов.

Билет 20

1. Методика проведения аудита безопасности веб-приложения: анализ исходного кода, проверка сторонних зависимостей (npm audit) и составление отчёта по регламенту.
2. Проанализируйте отчёт сканера уязвимостей зависимостей веб-проекта, классифицируйте найденные угрозы по уровню критичности и составьте технический план их устранения.