

Межгосударственное образовательное учреждение высшего образования
«Белорусско-Российский университет»

УТВЕРЖДАЮ

Ректор Белорусско-Российского
университета

М.Е. Лустенков

22. 06 2026

Регистрационный № УД- Б.Р.О.28.В.17.1/р

Документ подписан
простой электронной подписью

Информация о владельце:

ФИО: Лустенков Михаил Евгеньевич

Должность: Ректор

Дата: 09.09.2026

ОСНОВЫ WEB-ПРОГРАММИРОВАНИЯ

(наименование дисциплины)

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Направление подготовки 09.03.01 – Информатика и вычислительная техника
09.03.04 – Программная инженерия

Направленность (профиль) Программно-аппаратные средства информационных систем;
Разработка программно-информационных систем

Квалификация Бакалавр

	Форма обучения	
	Очная	
	09 03 01	09 03 04
Курс	2	2
Семестр	3, 4	3, 4
Лекции, часы	32	32
Лабораторные занятия, часы	50	68
Курсовая работа, семестр	4	4
Экзамен, семестр	3, 4	3, 4
Контактная работа по учебным занятиям, часы	82	100
Самостоятельная работа, часы	170	152
Всего часов / зачетных единиц	252/7	252/7

Кафедра-разработчик программы: Программное обеспечение информационных технологий

(название кафедры)

Составитель: Ю. В. Вайнилович, канд. техн. наук

Могилев, 2026

Рабочая программа составлена в соответствии с федеральным государственным образовательными стандартами высшего образования – бакалавриат по направлению подготовки 09.03.01 Информатика и вычислительная техника № 929 от 19.09.2017, учебным планом рег. № 090301-2.2, от 18.02.2026, в соответствии с федеральными государственными образовательными стандартами высшего образования - бакалавриат по направлению подготовки 09.03.04 Программная инженерия № 920 от 19.09.2017, учебным планом рег. № 090304-2.2 от 18.02.2026.

Рассмотрена и рекомендована к утверждению кафедрой «Программное обеспечение информационных технологий»

«02» марта 2026 г., протокол № 8.

Зав. Кафедрой


В. В. Кутузов

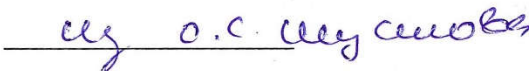
Одобрена и рекомендована к утверждению Научно-методическим советом
Белорусско-Российского университета
(протокол №6 от «15» апреля 2026 г.)

Рецензент:

Заведующий кафедрой программного обеспечения информационных технологий
УО «Могилевский государственный университет им. А.А. Кулешова»,
к.т.н., доцент И.В. Акиншева

Рабочая программа согласована:

Ведущий библиотекарь



Начальник учебно-методического
отдела


_____ О. Е. Печковская

1 ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

1.1. Цель дисциплины

Цель учебной дисциплины состоит в формировании у студентов глубоких теоретических знаний и практических навыков в области web-программирования, полном представлении об основных технологиях и инструментах, используемых при разработке веб-сайтов и приложений как на стороне клиента так и на стороне сервера.

1.2. Планируемые результаты изучения дисциплины

В результате освоения учебной дисциплины студент должен:

знать:

- архитектуру клиент-серверных веб-приложений и принципы их функционирования;
- структуру и семантику языка разметки гипертекста;
- принципы стилизации веб-документов и построения адаптивных интерфейсов;
- основы программирования на языке сценариев веб-страниц;
- принципы организации интерактивного взаимодействия пользователя с веб-интерфейсом;
- методы организации сетевых запросов и обмена данными между клиентской и серверной частями приложения;

- принципы компонентного подхода при разработке пользовательских интерфейсов;
- паттерны управления состоянием веб-приложения;
- основы организации серверной части веб-приложений;
- принципы работы с реляционными системами хранения данных;
- методы контроля версий, отладки и тестирования программного обеспечения.

уметь:

- использовать современные программные средства для разработки веб-приложений;
- проектировать и реализовывать пользовательские интерфейсы по готовым макетам;
- создавать адаптивные и интерактивные веб-страницы;
- реализовывать программную логику клиентской части приложения;
- организовывать обмен данными с внешними и внутренними источниками;
- разрабатывать компоненты интерфейса и обеспечивать их взаимодействие;
- создавать серверную часть веб-приложения и реализовывать взаимодействие с базой данных;

- применять системы контроля версий в процессе разработки;
- выполнять отладку и тестирование веб-приложений.

владеть:

- навыками работы с современными средствами разработки программного обеспечения;
- навыками верстки веб-страниц и построения адаптивных интерфейсов;
- навыками разработки клиентской логики веб-приложений;
- навыками реализации компонентного подхода в разработке интерфейсов;
- навыками организации взаимодействия клиентской и серверной частей веб-приложения;
- навыками работы с системами контроля версий;
- навыками отладки и тестирования программного обеспечения;
- навыками разработки полнофункциональных веб-приложений.

1.3 Место учебной дисциплины в системе подготовки студента

Дисциплина относится:

- для 09.03.01 к блоку 1 «Дисциплины (модули)» (Обязательная часть блока 1, Элективные дисциплины), дисциплина по выбору;

– для 09.03.04 к блоку 1 «Дисциплины (модули)» (Часть блока 1, формируемая участниками образовательных отношений), дисциплина по выбору.

Перечень учебных дисциплин, изучаемых ранее, усвоение которых необходимо для изучения данной дисциплины:

- Информатика;
- Программирование;
- Теория алгоритмов.

Перечень учебных дисциплин (циклов дисциплин), которые будут опираться на данную дисциплину:

- Графический дизайн и прототипирование;
- Программирование мобильных приложений;
- Средства взаимодействия человека с вычислительными системами;
- Тестирование и отладка программного обеспечения;
- Командная разработка программно-аппаратных комплексов (только для 09.03.01);
- Технологии командной разработки приложений (только для 09.03.04);
- Архитектура программных систем (только для 09.03.04).

Кроме того результаты, полученные при изучении дисциплины на лекционных занятиях будут применимы при прохождении первой технологической (проектно-технологической) практики и второй технологической (проектно-технологической) практики, а также при подготовке выпускной квалификационной работы и дальнейшей профессиональной деятельности.

1.4 Требования к освоению дисциплины

Освоение данной дисциплины должно обеспечивать формирование следующих компетенций

Коды формируемых компетенций	Наименование формируемых компетенций
для специальности 09.03.01	
ПК-3	Способен разрабатывать и проектировать программное обеспечение
ОПК-8	Способен разрабатывать алгоритмы и программы, пригодные для практического применения
для специальности 09.03.04	
ПК-8	Способность создавать программные интерфейсы
ПК-10	Владение навыками использования различных технологий разработки программного обеспечения

2 СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Вклад дисциплины в формирование результатов обучения выпускника (компетенций) и достижение обобщенных результатов обучения происходит путём освоения содержания обучения и достижения частных результатов обучения, описанных в данном разделе.

2.1 Содержание дисциплины

Номер тем	Наименование тем	Содержание	Коды формируемых компетенций	
			09.03.01	09.03.04
1	2	3	4	5
1	Введение в web-программирование	Архитектура клиент-сервер. Жизненный цикл запроса (DNS → TCP → HTTP → рендеринг). Роли HTML, CSS, JavaScript. SPA vs MPA. VS Code: расширения, Emmet, Live Server. DevTools	ПК-3, ОПК-8	ПК 10 ПК-8

		(Elements, Console, Network, Application). Стандарты W3C/WHATWG, кроссбраузерность, Can I Use		
2	Система контроля версий Git и GitHub	Понятие системы контроля версий, варианты СКВ. Терминология Git, настройка конфигурации, создание репозитория. Основные команды (add, commit, log, diff, checkout, reset). Ветвление, слияние, разрешение конфликтов. Работа с удалённым сервером: push, pull, clone. Pull request и code review. .gitignore, README.md, Markdown. GitHub Pages	ПК-3, ОПК-8	ПК 10 ПК-8
3	Язык разметки гипертекста HTML	HTTP, URL. Структура документа (DOCTYPE, head, body, meta). Семантические теги. Ссылки, якоря, списки, таблицы. Формы: элементы, типы input, валидация (required, pattern). Мультимедиа. Форматы графики. SVG. Доступность: ARIA, alt, роли. SEO-основы.	ПК-3, ОПК-8	ПК 10 ПК-8
4	Каскадные таблицы стилей CSS	Подключение стилей. Селекторы. Специфичность. Каскад и наследование. Блочная модель. Единицы измерения. Позиционирование. Z-index. Overflow. Flexbox. Grid. Медиазапросы, Mobile First. Адаптивные изображения. CSS-переменные. Анимации. Псевдоклассы и псевдоэлементы. Шрифты. Оптимизация изображений.	ПК-3, ОПК-8	ПК 10 ПК-8
5	Профессиональная вёрстка	Чтение макета в Figma (Inspect / Dev Mode, экспорт ассетов). Методология BEM. Препроцессоры: Sass/SCSS. CSS-фреймворк Bootstrap. Tailwind CSS.	ПК-3, ОПК-8	ПК 10 ПК-8
6	JavaScript	Подключение скриптов. Переменные. Типы данных. Операторы. Условные конструкции. Циклы. Функции: объявление, выражение, стрелочные, параметры, return. Область видимости. Регулярные выражения. Массивы: CRUD-методы, методы перебора. Объекты: создание, перебор, вложенность. Деструктуризация. Spread/rest. JSON. Замыкания. Контекст (this), call, apply, bind. Колбэки, функции высшего порядка. ES6-классы: constructor, методы, наследование (extends, super). Модули ES6. DOM-дерево. Поиск, создание, удаление элементов. DocumentFragment. textContent, innerHTML, classList, setAttribute. addEventListener, объект event. Всплытие и погружение. Делегирование событий. Работа с формами. Синхронный vs асинхронный код. Event Loop. Таймеры. requestAnimationFrame. Promises. Fetch API. Async/await. Try/catch/finally. cookie, localStorage, sessionStorage.	ПК-3, ОПК-8	ПК 10 ПК-8
7	Инструменты разработки и сборки	Node.js – установка и назначение. Пакетный менеджер NPM. Понятие о бандлерах: Vite, Webpack. Минификация, Babel – транспиляция. Линтинг и форматирование: ESLint, Prettier. Подключение сторонних библиотек через NPM. ESLint, Prettier	ПК-3, ОПК-8	ПК 10 ПК-8
8	Работа с сетью и фейковым API	Протокол HTTP/HTTPS: методы, заголовки, коды ответов. REST API: принципы, ресурсы,	ПК-3, ОПК-8	ПК 10 ПК-8

		эндпоинты. Формат JSON. Фейковые серверы: json-server, MockAPI.io, JSONPlaceholder. CORS. Практика CRUD через Fetch/async-await. Библиотека Axios: установка, синтаксис, сравнение с Fetch, интерсепторы, обработка ошибок. Индикация загрузки и обработка ошибок		
9	Библиотека React	Понятие фреймворка. Компонентный подход. Создание проекта. JSX-синтаксис. Функциональные компоненты. Props. Состояние. Обработка событий. Условный рендеринг. Хуки. Управляемые формы. Маршрутизация. Библиотеки UI-компонентов	ПК-3, ОПК-8	ПК 10 ПК-8
10	Архитектурный паттерн Redux	Проблема prop drilling, зачем нужен глобальный стейт. React Redux, Redux Toolkit, Redux thunk. Действия и редьюсеры. Middleware. Асинхронные операции.	ПК-3, ОПК-8	ПК 10 ПК-8
11	Менеджер состояния Zustand	Сравнение подходов к управлению состоянием: Redux vs Zustand. Установка и настройка Zustand. Создание store: create, set, get. Использование состояния в компонентах: useStore, селекторы. Асинхронные действия (async/await). Middleware: devtools, persist. Практика: рефакторинг фрагмента приложения с Redux на Zustand	ПК-3, ОПК-8	ПК 10 ПК-8
12	Тестирование React-компонентов	Зачем нужно тестирование. Виды тестов (unit, integration, e2e — обзорно). Jest, Vitest: установка, конфигурация, структура тестов. React Testing Library: render, screen, fireEvent, userEvent, waitFor. Тестирование компонентов: рендеринг, props, пользовательские события, асинхронные операции. Моки (jest.mock). Покрывание кода	ПК-3, ОПК-8	ПК 10 ПК-8
13	Безопасность, производительность и деплой	Основы веб-безопасности: XSS, CSRF – понимание угроз, санитизация ввода, HTTPS. Производительность: Lighthouse-аудит, оптимизация изображений, ленивая загрузка (loading="lazy", Intersection Observer), минификация, кеширование. Деплой: GitHub Pages, Netlify, Vercel. Понятие CI/CD. PWA: Service Worker и манифест	ПК-3, ОПК-8	ПК 10 ПК-8

2.2 Учебно-методическая карта дисциплины

3 семестр

№ недели	Лекции (наименование тем)	Часы	Лабораторные работы	Часы	Самостоятельная работа, часы	Форма контроля знаний	Баллы (max)
Модуль 1							
1	Тема 1. Введение в web-программирование	2	ЛР №1. Среда разработки, DevTools, основы HTML	2	1	ЗЛР	4
2			ЛР №2. Система контроля версий Git и GitHub	2	1	ЗЛР	4

3	Тема 2. Система контроля версий Git и GitHub	2	ЛР №3. Фиксированная вёрстка проекта пользовательского интерфейса	2	1	ЗЛР	4
4			ЛР №3. Фиксированная вёрстка проекта пользовательского интерфейса	2	1	ЗЛР	4
5	Тема 3. Язык разметки гипертекста HTML	2	ЛР №4. Адаптивная вёрстка проекта пользовательского интерфейса	2	1	ЗЛР	4
6			ЛР №4. Адаптивная вёрстка проекта пользовательского интерфейса	2	1	ЗЛР	5
7	Тема 4. Каскадные таблицы стилей CSS	2	ЛР №5. Анимации и визуальные эффекты в CSS	2	1	ЗЛР	5
8			ЛР №6. Динамическое управление DOM-деревом	2	1	ТКУ	30
Модуль 2							
9	Тема 5. Профессиональная вёрстка	2	ЛР №7. Обработка событий и валидация форм	2	1	ЗЛР	3
10			ЛР №8. Интерактивные компоненты web-интерфейса	2	1	ЗЛР	3
11	Тема 6. JavaScript	2	ЛР №8. Интерактивные компоненты web-интерфейса	2	1	ЗЛР	4
12			ЛР №9. Клиентское хранилище данных	2	1	ЗЛР	4
13	Тема 7. Инструменты разработки и сборки	2	ЛР №9. Клиентское хранилище данных	2	2	ЗЛР	4
14			ЛР №10. Сборка и автоматизация проекта	2	2	ЗЛР	4
15	Тема 8. Работа с сетью и фейковым API	2	ЛР №11. Взаимодействие с REST API	2	2	ЗЛР	4
16			ЛР №11. Взаимодействие с REST API	2	2	ЗЛР	4
17			ЛР №12. Интеграция сторонних веб-сервисов	2	2	ТКУ	30
18-20					36	ПА (экзамен)	40
	Итого	16		34	58		100

4 семестр
Для специальности 09.03.01

№ недели	Лекции (наименование тем)	Часы	Лабораторные работы	Часы	Самостоятельная работа, часы	Форма контроля знаний	Баллы (max)
Модуль 1							
1	Тема 9. Библиотека React	2	ЛР №13. Архитектура React-приложения	2	5	ЗЛР	8

2							
3	Тема 9. Библиотека React	2	ЛР №14. Маршрутизация в React-приложениях	2	5	ЗЛР	8
4							
5	Тема 9. Библиотека React	2	ЛР №15. Управление состоянием компонентов React	2	5		
6							
7	Тема 10. Архитектурный паттерн Redux	2	ЛР №15. Управление состоянием компонентов React	2	5	ЗЛР ТЗ	8 6
8						ТКУ	30
Модуль 2							
9	Тема 10. Архитектурный паттерн Redux	2	ЛР №16. Нуки React. Работа с API	2	5	ЗЛР	7
10							
11	Тема 11. Менеджер состояния Zustand	2	ЛР №17. Библиотека MUI	2	5	ЗЛР	7
12							
13	Тема 12. Тестирование React-компонентов	2	ЛР №18. Менеджер состояний Redux Toolkit	2	5	ЗЛР	8
14							
15	Тема 13. Безопасность, производительность и деплой	2	ЛР №21. Интеграция сторонних web-сервисов	2	5	ЗЛР	8
16							
17						ТКУ	30
1-17	Выполнение курсовой работы				36		
18-20					36	ПА (экзамен)	40
	Итого	16		16	112		100

4 семестр

Для специальности 09.03.04

№ недели	Лекции (наименование тем)	Часы	Лабораторные работы	Часы	Самостоятельная работа, часы	Форма контроля знаний	Баллы (max)
Модуль 1							
1	Тема 9. Библиотека React	2	ЛР №13. Архитектура React-приложения	2			
2			ЛР №13. Архитектура React-приложения	2		ЗЛР	5
3	Тема 9. Библиотека React	2	ЛР №14. Маршрутизация в React-приложениях	2	3	ЗЛР	5
4			ЛР №15. Управление состоянием компонентов React	2			

5	Тема 9. Библиотека React	2	ЛР №15. Управление состоянием компонентов React	2	3	ЗЛР	5
6			ЛР №16. Нуки React. Работа с API	2			
7	Тема 10. Архитектурный паттерн Redux	2	ЛР №16. Нуки React. Работа с API	2		ЗЛР	5
8			ЛР №17. Библиотека MUI	2		ЗЛР ТЗ ТКУ	5 5 30
Модуль 2							
9	Тема 10. Архитектурный паттерн Redux	2	ЛР №18. Менеджер состояний Redux Toolkit	2	3		
10			ЛР №18. Менеджер состояний Redux Toolkit	2		ЗЛР	5
11	Тема 11. Менеджер состояния Zustand	2	ЛР №19. Менеджер состояния Zustand	2	3	ЗЛР	5
12			ЛР №20. Кэширование запросов TanStack Query	2		ЗЛР	5
13	Тема 12. Тестирование React-компонентов	2	ЛР №21. Интеграция сторонних web-сервисов	2	3		
14			ЛР №21. Интеграция сторонних web-сервисов	2		ЗЛР	5
15	Тема 13. Безопасность, производительность и деплой	2	ЛР №22. Тестирование React-компонентов	2	3		
16			ЛР №22. Тестирование React-компонентов	2		ЗЛР	5
17			ЛР №23. Деплой веб-приложения	2	4	ЗЛР ТКУ	5 30
1-17	Выполнение курсовой работы				36		
18-20					36	ПА (экзамен)	40
	Итого	16		34	94		100

Принятые обозначения:

ТКУ – текущий контроль успеваемости;

ПА – промежуточная аттестации;

ТЗ – тестовое задание;

ЗЛР – защита лабораторной работы.

Итоговая оценка определяется как сумма текущего контроля и промежуточной аттестации и соответствует баллам:

Экзамен

Оценка	Отлично	Хорошо	Удовлетворительно	Неудовлетворительно
Баллы	87-100	65-86	51-64	0-50

2.3. Требования к курсовой работе

Целью курсового проектирования является приобретение практических навыков разработки web-приложений.

Примерная тематика курсовых работ хранится на кафедре.

Содержание курсовой работы включает три части:

- 1) техническое задание – анализ предметной области, сбор требований к web-приложению;
- 2) проектирование дизайна web-приложения – разработка дизайн-макетов всех страниц приложения в соответствии с требованиями;
- 3) реализация web-приложения – разработка клиентской компоненты web-приложения.

Разбивка этапов курсовой работы, определение количества минимальных и максимальных баллов за каждый из них производится преподавателем. Примерный перечень этапов выполнения курсовой работы и количество баллов за каждый из них представлен в таблице.

№	Этап выполнения	Минимум	Максимум
1	Техническое задание	9	15
2	Проектирование дизайна web-приложения	12	20
3	Реализация web-приложения	12	20
4	Оформление пояснительной записки и графического материала	3	5
	Итого за выполнение курсового проекта	36	60
	Защита курсовой работы	15	40

3 ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

При изучении дисциплины используется модульно-рейтинговая система оценки знаний студентов. Применение форм и методов проведения занятий при изучении различных тем курса представлено в таблице.

№ п/п	Форма проведения занятия	Вид аудиторных занятий		Всего часов
		Лекции	Лабораторные занятия	
для специальности 09.03.01				
1	С использованием ЭВМ	1-13	1-18, 21	82
	ИТОГО	32	50	82
для специальности 09.03.04				
1	С использованием ЭВМ	1-13	1-23	100
	ИТОГО	32	68	100

4. ОЦЕНОЧНЫЕ СРЕДСТВА

Используемые оценочные средства по учебной дисциплине представлены в таблице и хранятся на кафедре.

№ п/п	Вид оценочных средств*	Количество комплектов
1	Вопросы к экзамену/лабораторным работам	1/22
2	Экзаменационные билеты	1
3	Перечень тем курсовых работ	1
4	Тестовые задания	1

5 МЕТОДИКА И КРИТЕРИИ ОЦЕНКИ КОМПЕТЕНЦИЙ СТУДЕНТОВ

5.1 Формируемые компетенции и индикаторы их достижения

Для специальности 09.03.01

<i>ПК-3. Способен разрабатывать и проектировать программное обеспечение</i>
<i>ИПК-3.1. Разрабатывает программное обеспечение с использованием современных средств и технологий</i>
Знает: Современные средства и технологии разработки программного обеспечения; архитектурные паттерны построения веб-приложений; принципы организации и

структурирования программного кода; методы обеспечения качества программного обеспечения
Умеет: Применять современные средства разработки для создания программного обеспечения; проектировать архитектуру веб-приложений; организовывать структуру проекта в соответствии с принятыми стандартами; использовать системы контроля версий для управления разработкой
Владеет: Навыками разработки программного обеспечения с использованием современных технологий и инструментов; навыками проектирования и реализации компонентов веб-приложений; навыками организации командной разработки
<i>ОПК-8. Способен разрабатывать алгоритмы и программы, пригодные для практического применения</i>
<i>ИОПК-8.4 Разрабатывает серверную и клиентскую логику веб-приложений с использованием современных языков программирования, технологий и фреймворков.</i>
Знает: Принципы разработки клиентской и серверной частей веб-приложений; современные языки программирования, технологии и фреймворки для веб-разработки; методы организации взаимодействия между компонентами приложения; принципы работы с базами данных
Умеет: Разрабатывать клиентскую логику веб-приложений с использованием современных фреймворков; создавать серверную часть приложения и реализовывать программный интерфейс; организовывать взаимодействие с базами данных; обеспечивать обмен данными между клиентом и сервером
Владеет: Навыками разработки клиентской части веб-приложений; навыками создания серверной логики и программных интерфейсов; навыками интеграции клиентской и серверной частей в единое приложение; навыками отладки и тестирования веб-приложений

Для специальности 09.03.04

<i>ПК-8. Способен создавать программные интерфейсы</i>
<i>ИПК-8.3. Способен создавать программные интерфейсы Webприложений</i>
Знает: Принципы проектирования программных интерфейсов веб-приложений; архитектурные стили построения программных интерфейсов; стандарты и протоколы обмена данными в веб-приложениях; методы обеспечения безопасности программных интерфейсов
Умеет: Проектировать программные интерфейсы веб-приложений в соответствии с требованиями; реализовывать программные интерфейсы с использованием современных технологий; организовывать обмен данными между компонентами приложения; документировать программные интерфейсы
Владеет: Навыками проектирования и реализации программных интерфейсов веб-приложений; навыками работы с протоколами обмена данными; навыками обеспечения взаимодействия клиентской и серверной частей приложения; навыками тестирования программных интерфейсов
<i>ПК-10. Владеет навыками использования различных технологий разработки программного обеспечения</i>
<i>ИПК-10.2. Умеет использовать современные технологии WEBпрограммирования и Интернетпрограммирования в решении практических задач разработки ПО</i>
Знает: Современные технологии веб-программирования и интернет-программирования; принципы построения пользовательских интерфейсов веб-приложений; технологии разработки клиентской и серверной частей приложений; инструменты и средства автоматизации разработки
Умеет: Применять современные технологии веб-программирования для решения практических задач; разрабатывать пользовательские интерфейсы по готовым макетам; создавать интерактивные веб-приложения; интегрировать различные компоненты и сервисы в единое приложение
Владеет: Навыками применения технологий веб-программирования в практической деятельности; навыками разработки клиентской и серверной частей веб-приложений; навыками

использования современных инструментов и фреймворков; навыками решения типовых задач веб-разработки

5.2 Критерии оценки лабораторных работ

1) вопросы к лабораторным работам

Балл	Критерии
3-8	Студент демонстрирует полное понимание темы, грамотно использует терминологию, даёт исчерпывающие ответы на вопросы, способен обосновать принятые решения и связать их с теоретическими положениями
2	Студент демонстрирует базовое понимание темы, допускает ошибки в терминологии, отвечает на вопросы частично, затрудняется в обосновании принятых решений
1	Студент не понимает сути выполненной работы, не владеет терминологией, не способен ответить на основные вопросы по теме

2) тестовые задания

Балл	Критерии
5-6	85–100% правильных ответов
4	70–84% правильных ответов
3	55–69% правильных ответов
2	Менее 55% правильных ответов

5.3 Критерии оценки курсовой работы

Баллы	Критерии оценивания
36–40	Все задачи проекта решены в полном объёме, содержание полностью соответствует заданию. Приложение полностью работоспособно, все заявленные функции реализованы и корректно работают. Студент демонстрирует полную самостоятельность в решении задач, уверенно ориентируется в коде и архитектуре проекта. Код чистый, структурированный и читаемый, соблюдены принципы организации проекта. Проект содержит оригинальные решения, выходящие за рамки базовых требований. Пояснительная записка оформлена в соответствии с требованиями, графическая часть качественная и информативная. Доклад чёткий, логически выстроенный, студент свободно владеет профессиональной терминологией. Студент уверенно и аргументированно отвечает на все заданные вопросы.
28–35	Основные задачи проекта решены, содержание в целом соответствует заданию. Приложение работоспособно, основные функции реализованы, имеются незначительные ошибки. Студент выполнил работу в основном самостоятельно, ориентируется в коде, но затрудняется с объяснением отдельных решений. Код в целом читаемый, структура проекта логична, имеются незначительные нарушения стиля. Проект содержит отдельные элементы, демонстрирующие инициативность студента. Оформление в целом соответствует требованиям, имеются незначительные отклонения. Доклад понятный, терминология используется с незначительными неточностями. Студент отвечает на большинство вопросов, но отдельные ответы недостаточно полные.
20–27	Задачи проекта решены частично, содержание недостаточно полно раскрывает тему. Приложение частично работоспособно, часть заявленных функций не реализована или работает некорректно. Самостоятельность выполнения работы вызывает сомнения, студент слабо ориентируется в отдельных частях проекта. Код плохо структурирован, имеется значительное дублирование, отсутствует единый стиль. Проект выполнен строго по заданию без дополнительной инициативы. Оформление имеет существенные нарушения требований, графическая часть неполная. Доклад сбивчивый, нарушена логика изложения, допускаются ошибки в терминологии. Студент затрудняется с ответами на вопросы, демонстрирует поверхностное понимание материала.
15–19	Задачи проекта решены фрагментарно, содержание не соответствует заданию. Приложение неработоспособно или имеет критические ошибки. Работа выполнена несамостоятельно, студент не ориентируется в собственном проекте. Код нечитаемый, структура проекта отсутствует. Элементы творчества и инициативы отсутствуют. Пояснительная записка

Баллы	Критерии оценивания
	оформлена с грубыми нарушениями. Доклад не раскрывает содержание проекта. Студент не может ответить на вопросы.

5.4 Критерии оценки экзамена

Для оценки экзамена используется следующее оценочное средство:

1) Экзаменационные билеты

Экзаменационный билет включает 3 теоретических вопроса. Ответы на вопросы оцениваются по следующим критериям.

Баллы	Критерии оценивания
36–40	Ответ полный, логически выстроенный, демонстрирует глубокое понимание материала. Студент свободно владеет терминологией, корректно раскрывает содержание всех вопросов билета, приводит примеры и обоснования. Допускаются незначительные неточности, не влияющие на общее качество ответа.
28–35	Ответ в целом правильный и достаточно полный. Студент демонстрирует хорошее понимание материала, владеет терминологией, раскрывает основное содержание вопросов. Допускаются отдельные неточности или недостаточная аргументация по некоторым аспектам.
20–27	Ответ неполный, но содержит основные положения по вопросам билета. Студент демонстрирует общее понимание материала, однако допускает ошибки в использовании терминологии или раскрытии отдельных понятий. Ответ требует уточняющих вопросов со стороны преподавателя.
10–19	Ответ фрагментарный, раскрывает лишь отдельные аспекты вопросов билета. Студент допускает существенные ошибки, демонстрирует слабое владение терминологией и недостаточное понимание взаимосвязей между понятиями.
1–9	Ответ неудовлетворительный. Студент демонстрирует отрывочные знания, не раскрывает содержание вопросов билета, допускает грубые ошибки. Ответ не соответствует минимальным требованиям к освоению дисциплины.
0	Ответ отсутствует или полностью не соответствует содержанию вопросов билета.

6 МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ОРГАНИЗАЦИИ И ВЫПОЛНЕНИЮ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ

Самостоятельная работа студентов (СРС) направлена на закрепление и углубление освоения учебного материала, развитие практических умений. СРС включает следующие виды самостоятельной работы студентов:

- самостоятельное изучение материала по учебникам и другим источникам;
- решение задач и упражнений по образцу.

Перечень контрольных вопросов и заданий для самостоятельной работы студентов приведен в приложении и хранится на кафедре.

Для СРС рекомендуется использовать источники, приведенные в п. 7.

7 УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

7.1 Основная литература:

№ п/п	Библиографическое описание	Гриф	Количество экземпляров/URL
1	Полуэктова, Н. Р. Разработка веб-приложений : учебник для вузов / Н. Р. Полуэктова. — 2-е изд. — Москва : Издательство Юрайт, 2026. — 204 с.	Рекомендовано УМО ВО в качестве учебника для студентов ВУЗов	https://urait.ru/bcode/588458

7.2 Дополнительная литература

№ п/п	Библиографическое описание	Гриф	Количество экземпляров/URL
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).	Рекомендовано УМО ВО в качестве учебника для студентов ВУЗов	https://urait.ru/bcode/562916

7.3 Перечень ресурсов сети Интернет по изучаемой дисциплине

<http://moodle.bru.by> – Образовательный портал Белорусско-Российского университета;

<http://e.biblio.bru.by/> – Электронная библиотека Белорусско-Российского университета;

<https://znanium.com/> – Электронно-библиотечная система Znanium;

<https://stepik.org/catalog> – Российская образовательная платформа и конструктор бесплатных открытых онлайн-курсов и уроков;

<https://habr.com/ru/> – Хабр. Публикации по ИТ тематикам;

<https://developer.mozilla.org/ru/docs/Web/HTML> – MDN Web Docs. Основы HTML: теги, атрибуты, семантическая верстка, формы, мультимедиа.

<https://developer.mozilla.org/ru/docs/Web/CSS> – MDN Web Docs. Полная документация по CSS, включая флекс- и сеточную (grid) верстку.

<https://css-tricks.com/snippets/css/complete-guide-grid/> – CSS-Tricks. Подробное руководство по CSS Grid с практическими примерами.

<https://sass-lang.com/> – Официальная документация препроцессора Sass/Scss (Less можно использовать по аналогии через официальный сайт Less).

<https://developer.mozilla.org/ru/docs/Web/JavaScript> – MDN Web Docs. Базовая и продвинутая справка по JavaScript, DOM, событиям, таймерам, localStorage.

<https://javascript.info/> – Learn JavaScript. Современный подробный курс по JavaScript, включая промисы, async/await, Fetch API.

https://developer.mozilla.org/ru/docs/Web/API/Fetch_API – MDN Web Docs. Официальная документация по Fetch API для асинхронных сетевых запросов.

<https://www.typescriptlang.org/docs/handbook/> – TypeScript Handbook. Официальное руководство по типизации объектов, массивов, функций, дженерикам, интерфейсам и кортежам.

<https://basarat.gitbook.io/typescript/> – TypeScript Deep Dive. Детальный бесплатный справочник по продвинутым возможностям TypeScript.

<https://react.dev/> – React Documentation. Официальная документация React: функциональные компоненты, hooks, роутинг, условный рендеринг.

<https://reactrouter.com/> – React Router. Документация библиотеки для реализации SPA-приложений в React.

<https://mui.com/> – Material-UI. Документация библиотеки компонентов Material UI для React.

<https://react-bootstrap.github.io/> – React Bootstrap. Справка по React-компонентам на базе Bootstrap.

<https://redux.js.org/tutorials/essentials/part-1-overview-concepts> – Redux Essentials. Официальный tutorial по Redux: actions, reducers, middleware, Redux Toolkit.

<https://redux-toolkit.js.org/rtk-query/overview> – RTK Query. Документация по работе с API через Redux Toolkit.

<https://github.com/axios/axios> – Axios GitHub. Официальная справка по библиотеке Axios для HTTP-запросов.

<https://nodejs.org/en/docs/> – Node.js Documentation. Основы Node.js на сервере: модули, файловая система, стримы, асинхронные методы.

<https://expressjs.com/en/guide/routing.html> – Express.js Guides. Документация по Express: роутинг, middleware, REST API, раздача статики.

<https://www.postgresql.org/docs/> – PostgreSQL Manual. Официальная документация по PostgreSQL и SQL-операциям.

<https://www.mongodb.com/docs/manual/> – MongoDB Manual. Руководство по MongoDB и CRUD-операциям.

<https://sequelize.org/> – Sequelize. ORM для PostgreSQL/MySQL/SQLite в Node.js.

<https://mongoosejs.com/> – Mongoose. ORM/ODM для MongoDB в Node.js.

<https://jestjs.io/> – Jest. Официальный сайт популярного фреймворка для unit-тестирования JavaScript и React.

<https://testing-library.com/docs/react-testing-library/intro/> – React Testing Library. Подход к тестированию отображения и поведения React-компонентов.

<https://owasp.org/> – OWASP. Материалы по тестированию и обеспечению безопасности веб-приложений.

<https://business-online.su/blog/klyuchevye-principy-raboty-poiskovyh-sistem/> – Статья о работе поисковых систем. Объяснение, как роботы индексируют контент, что важно для SPA-приложений.

7.4 Перечень наглядных и других пособий, методических рекомендаций по проведению учебных занятий, а также методических материалов к используемым в образовательном процессе техническим средствам

Методические рекомендации

Основы WEB-программирования [Электронный ресурс] : метод. рек. к лаб. работам для студентов / сост. Ю. В. Вайнилович. - Могилев : Белорус.-Рос. ун-т.

Основы web-программирования [Электронный ресурс] : метод. рек. к курсов. проектир. для студентов / сост. Ю. В. Вайнилович. - Могилев : Белорус.-Рос. ун-т, 2023. - 36с.

Перечень программного обеспечения, используемого в учебном процессе

№	Наименование	Назначение	Лицензия
1	Visual Studio Code	Редактор исходного кода	Бесплатное (MIT)
2	Git	Распределённая система контроля версий	Бесплатное (GPL-2.0)
3	Node.js	Среда выполнения JavaScript	Бесплатное (MIT)
4	npm	Пакетный менеджер для Node.js	Бесплатное
5	PostgreSQL	Реляционная СУБД	Бесплатное (PostgreSQL License)
6	pgAdmin	Графический клиент для PostgreSQL	Бесплатное (PostgreSQL License)
7	Google Chrome	Веб-браузер с инструментами разработчика	Бесплатное
8	Mozilla Firefox	Веб-браузер с инструментами разработчика	Бесплатное (MPL-2.0)
9	Microsoft Edge	Веб-браузер с инструментами разработчика	Бесплатное
10	Figma	Инструмент проектирования UI/UX	Бесплатное (Free Tier)
11	Postman	Платформа для тестирования API	Бесплатное (Free Tier)
12	Thunder Client	Расширение VS Code для тестирования API	Бесплатное
13	Webpack	Сборщик модулей JavaScript	Бесплатное (MIT)
14	Babel	Транспилатор JavaScript	Бесплатное (MIT)
15	ESLint	Статический анализатор кода	Бесплатное (MIT)
16	Prettier	Инструмент форматирования кода	Бесплатное (MIT)
17	Live Server	Локальный сервер с автообновлением	Бесплатное (MIT)
18	GitHub Pages	Хостинг статических сайтов	Бесплатное

№	Наименование	Назначение	Лицензия
19	GitHub	Платформа хостинга репозитория	Бесплатное
20	React	Библиотека для создания интерфейсов	Бесплатное (MIT)
21	React Router	Маршрутизация в React	Бесплатное (MIT)
22	Redux Toolkit	Управление состоянием	Бесплатное (MIT)
23	Axios	HTTP-клиент	Бесплатное (MIT)
24	MUI (Material UI)	UI-компоненты для React	Бесплатное (MIT)
25	Express.js	Веб-фреймворк для Node.js	Бесплатное (MIT)
26	Sequelize	ORM для реляционных баз данных	Бесплатное (MIT)
27	Framer Motion	Библиотека анимаций	Бесплатное (MIT)
28	Recharts	Визуализация данных	Бесплатное (MIT)
29	react-hook-form	Работа с формами	Бесплатное (MIT)
30	react-i18next	Интернационализация	Бесплатное (MIT)
31	react-toastify	Уведомления	Бесплатное (MIT)
32	bcrypt	Хеширование паролей	Бесплатное (MIT)
33	jsonwebtoken	JWT-токены	Бесплатное (MIT)
34	multer	Загрузка файлов	Бесплатное (MIT)
35	cors	Middleware для CORS	Бесплатное (MIT)
36	dotenv	Переменные окружения	Бесплатное (BSD-2-Clause)

8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

Материально-техническое обеспечение дисциплины содержится в паспорте лаборатории а. 517/2, рег. № паспорта лаборатории № ПУЛ - 4 517/2- 25; в паспорте лаборатории а. 518/2, рег. № паспорта лаборатории № ПУЛ - 4 518/2- 25; в паспорте лаборатории а. 502/2, рег. № паспорта лаборатории № ПУЛ - 4 502/2- 25.