Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Артеменков Михаил Николаевич
Должность: Ректор
Дата подписания: 09.11.2026 18:40:57
Уникальный программный ключ:
a63b74166e8f4fc17bd1e0462725f79e393bfdf9

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Смоленский государственный университет»

Кафедра прикладной математики и информатики

«Утверждаю»
Проректор по учебно-методической
деятельности
_____ Ю.А. Устименко
«25» декабря 2025 г.

Рабочая программа дисциплины
Б1.В.ДВ.01 Web-программирование

Направление подготовки **01.04.02 Прикладная математика и информатика**
Направленность (профиль) **Математическое и программное обеспечение цифровых радиотехнических систем и комплексов**

Форма обучения: очная
Курс – 1
Семестр – 1
Всего зачетных единиц – 2, часов–72
Форма отчетности: зачет – 1 семестр

Программу разработала: кандидат технических наук, доцент Т.А. Самойлова

Одобрена на заседании кафедры
«20» ноября 2025 г., протокол № 5

Заведующий кафедрой _____ С.В. Козлов

Смоленск
2025

1. Место дисциплины в структуре ОП

Дисциплина «Web - программирование» относится к дисциплинам по выбору. Она изучается в 1 семестре и является вспомогательной для изучения таких дисциплин, как «Проектирование сетевых информационных систем», «Параллельное программирование прикладных задач»и др.

При изучении данной дисциплины необходимы компетенции студентов, сформированные при изучении таких дисциплин, как «Информационные технологии», «Информационные системы»,«Базы данных» уровня бакалавриата/специалитета.

В ходе изучения дисциплины «Web - программирование» рассматриваются вопросы выбора основных архитектур современных веб -приложений, использование баз данных при разработке веб - приложений, применение технологий взаимодействия клиентских и серверных веб - приложений. Дисциплина способствует формированию у магистра целостного представления о возможностях использования веб – технологий в ходе решения задач обработки данных и прикладных математических задач. Приобретенные знания помогут магистру выбрать направление будущих научных исследований.

2. Планируемые результаты обучения по дисциплине

Компетенция	Индикаторы достижения
ПК-3. Способен разрабатывать программное обеспечение цифровых радиотехнических систем и комплексов	**Знает:** архитектуру аппаратной платформы, синтаксис, принципы и особенности программирования (в том числе кроссплатформенного, распределенного и параллельного программирования), стандартные библиотеки выбранного языка программирования, стандарты реализации интерфейсов устройств, технологии разработки и отладки программных продуктов, принципы информационного построения сетевого взаимодействия, методики тестирования программного обеспечения. **Умеет:** использовать методы и приемы формализации и алгоритмизации поставленных задач, применять выбранный язык программирования для написания программного кода, осуществлять отладку программного обеспечения. **Владеет:** основными методами и приемами разработки и отладки программных продуктов.

3. Содержание дисциплины

Основные понятия веб разработки. Одностраничные и многостраничные приложения- Single Page Application (SPA), Multi Page Application (MPA), архитектура, преимущества в разработке. Язык разметки HTML5. Основы таблиц стилей CSS. Основные теги, псевдоклассы и псевдоэлементы. Работа с технологией CSS Flexbox. Работа с макетом в графическом редакторе Figma. Основы создания адаптивного сайта. Применение Bootstrap.. Верстка Grid layout, возможности CSS3.

Frontend разработка. Основы языка JavaScript. Основные операторы JavaScript. Циклы, массивы и структуры данных. Объекты в JavaScript. Введение в DOM объектную модель документа. Обработка событий в JavaScript. Современный JavaScript. Асинхронные запросы. Фреймворк Vue.js. Компоненты Vue.js. JavaScript на сервере.

Backend разработка. Платформа Node.js. Инструменты разработки. Работа с сетевыми запросами. Фреймворк Express. Библиотека ReactJS. Настройка среды разработки и React приложения. Работа с API. Прогрессивные веб приложение (Progressive Web App).

Доступ веб – приложения к СУБД (системе управления базами данных). Работа с архитектурным подходом REST API. Создание веб-API с помощью Node.js и Express. Использование WEbAPI для доступа к СУБД.

4. Тематический план

№ п/п	Разделы и темы	Всего часов	Формы занятий	
			лабораторные занятия	самостоятельная работа
1	Современные подходы к разработке веб-приложений	18	8	10
2	Frontend разработка	18	8	10
3	Backend разработка	18	8	10
4	Доступ веб – приложения к СУБД	18	8	10
	ИТОГО	72	32	40

5. Виды образовательной деятельности

Занятия семинарского типа - лабораторные работы

Задания для лабораторных работ, размещены в ЭИОС СмолГУ (https://cdo.smolgu.ru/local/crw/course.php?id=5356).

На занятиях для каждой работы задание предоставляется студентам в электронном виде.

Лабораторные работы №1-4. ВерсткаHTML_JavaScript. CSSGridLayout, Flexbox
Лабораторные работы №5-8. Использование JavaScript - контента в WEB-приложениях.
Лабораторные работы № 9-12. Модель DOM на платформе Node.js
Лабораторные работы №13-16. Разработка SPA-приложения средствами React.

Самостоятельная работа

Самостоятельная работа направлена на углубление и закрепление знаний студентов и развитии практических умений. Она заключается в работе с лекционными материалами, поиске и обзоре литературы и электронных источников, информации по заданным темам курса, в изучении тем, вынесенных на самостоятельную проработку и самостоятельном выполнении заданий.

Темы для самостоятельного изучения

1. Приемы обработки дизайн - макета сайта в Figma.
2. Синтаксис и семантика HTML-элементов.
3. Операторы Java Script (JS).
4. Функции JS.
5. Массивы и объекты JS.
6. JavaScript в браузере. DOM.
7. Хранение кода проекта в Git-репозитории.
8. Встроенные модули Node.js. Менеджер пакетов npm.

Задания для самостоятельной работы

Задание 1. Разработать адаптивную визитку с использованием HTML/CSS.
Задание 2. Реализовать клиентскую валидацию с помощью JavaScript.

Задание 3. Разработать PHP-скрипт, принимающий текстовые и числовые значения через GET-параметры.

Задание 4. Разработать форму регистрации с полями: логин, email, пароль.

Задание 5. Разработать PHP-API для выполнения CRUD-операций.

6. Критерии оценивания результатов освоения дисциплины (модуля)

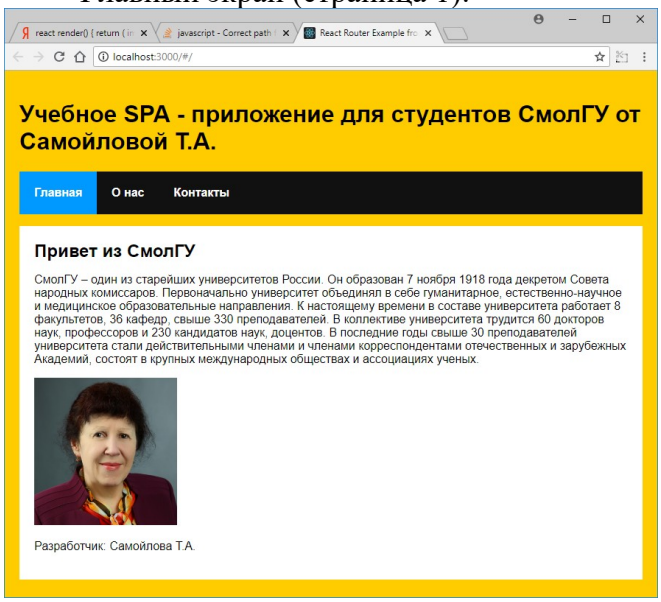
6.1. Оценочные средства и критерии оценивания для текущей аттестации

1. Задания для лабораторных занятий

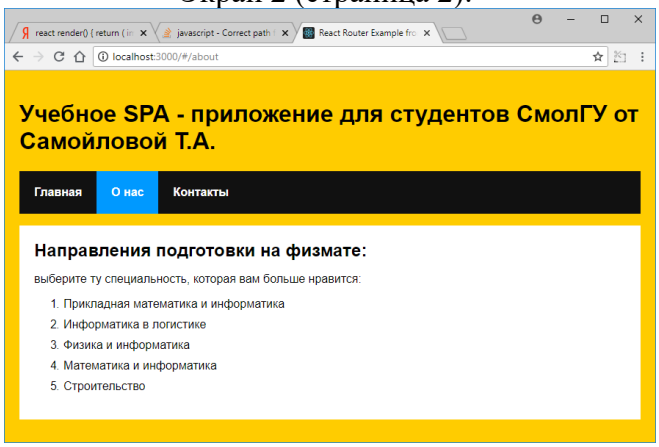
Образец задания

Создайте веб-приложение SPA - архитектуры средствами фреймворка React.js и React Router, содержащее меню и вывод соответствующих его пунктам 3-4 экранов произвольного содержания. Ваша фамилия и фото на одной из страниц обязательны. Для пользователя экраны должны полностью соответствовать страницам. Ваши страницы (обычно называемые представлениями) должны загружаться средствами одной и той же страницы Пример работы SPA-приложения приведен ниже.

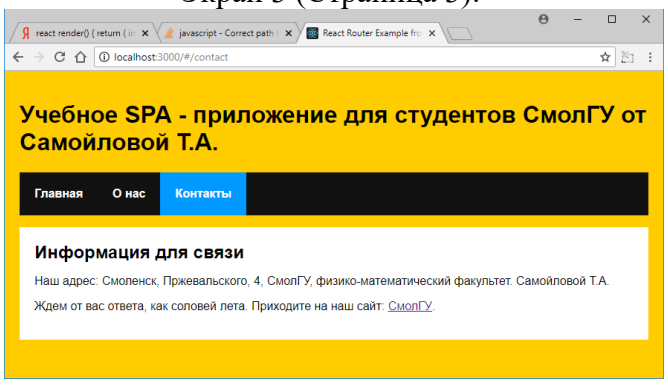
Главный экран (страница 1):



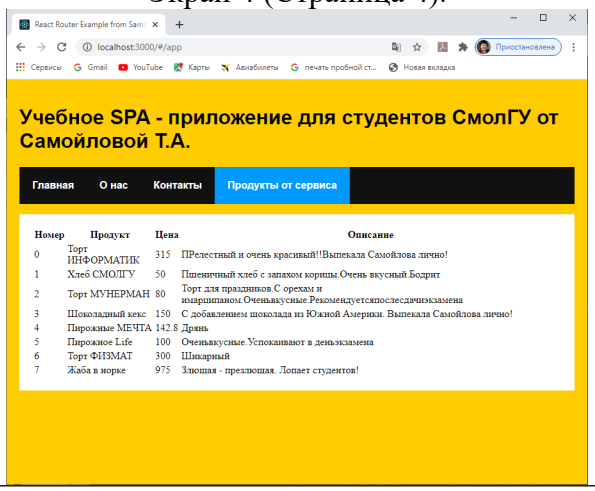
Экран 2 (страница 2):



Экран 3 (Страница 3):



Экран 4 (Страница 4):



Критерии оценивания задания для лабораторной работы:

№ п/п	Критерий	Балл

1.	Задание выполнено в срок (на соответствующем занятии)	1
2.	Задание выполнено студентом самостоятельно (без помощи и подсказок преподавателя)	1
3.	Результат строго соответствует заданию, служит прямым ответом на соответствующий вопрос задания	1
4.	Представленный результат проходит проверку (тестирование) на разных наборах входных данных	1
5.	Студент дает правильные пояснения о том, как выполнялось задание и каковы его результаты	1
6.	Работа оформлена в соответствии с требованиями (структура, форматирование, подписи, комментарии в коде и т. п.)	1
7.	Алгоритм/решение выстроено последовательно, блоки кода или расчётов логически разделены и обоснованы	1
8.	Использованы оптимальные методы и алгоритмы; решение не содержит избыточных или неоправданно сложных шагов	1
9.	В работе (особенно в коде или расчётах) присутствуют понятные пояснения, облегчающие восприятие	1
10.	Студент уверенно отвечает на дополнительные вопросы по заданию, демонстрирует понимание смежных аспектов задачи	1
	Итоговая (суммарная) оценка	Max - 10

Оценка «зачтено» - 5 баллов и более; оценка «не зачтено» - менее 5 баллов.

2. Теоретические вопросы

Теоретические вопросы предлагаются студенту во время сдачи лабораторного задания и направлены на проверку знания студентом основных теоретических положений дисциплины.

Примеры теоретических вопросов:

1. Как создать веб-приложение SPA - архитектуры средствами фреймворка React.js?
2. Как создать веб-приложение SPA - архитектуры средствами фреймворка React Router?
3. В чему существенное отличие создания веб-приложения средствами фреймворка React.js и React Router?

Критерии оценки ответов на теоретические вопросы:

Показатель	Количество баллов
1) Полнота раскрытия темы (ответ охватывает все ключевые аспекты вопроса, не упускает существенные детали)	1
2) Точность терминологии (корректно использованы понятия и термины дисциплины; отсутствуют смысловые искажения)	1
3) Логичность и структура (ответ выстроен последовательно: есть введение, основная часть, выводы; мысли связаны между собой)	1
4) Глубина понимания (показано осмысление материала: не просто пересказ, а объяснение причинно-следственных связей, принципов, механизмов)	1
5) Аргументированность (положения подкреплены примерами, ссылками на источники, формулы, теоремы или экспериментальные данные)	1
6) Самостоятельность изложения (ответ не сводится к дословному воспроизведению учебника/лекции; присутствует индивидуальная интерпретация)	1
7) Актуальность и современность (упомянуты современные исследования, технологии или дискуссии по теме (если применимо))	1
8) Корректность примеров (приведённые иллюстрации точно соответствуют	1

теории, не содержат ошибок и не вводят в заблуждение)	
9) Ясность и доступность (текст понятен без дополнительных уточнений; сложные моменты разъяснены; нет избыточной перегруженности)	1
10) Грамотность и оформление (отсутствуют орфографические, пунктуационные и стилистические ошибки; формулы, схемы, цитаты оформлены по правилам)	1
Итоговая (суммарная) оценка	Max - 10

Оценка «зачтено» - 5 баллов и более; оценка «не зачтено» - менее 5 баллов.

3. Выполнение заданий для самостоятельной работы (перечень заданий приведен в разделе 5 «Самостоятельная работа»).

Критерии оценивания выполнения заданий:

№ п/п	Критерий	Балл
1.	Задание выполнено в срок (на соответствующем занятии)	1
2.	Задание выполнено студентом самостоятельно (без помощи и подсказок преподавателя)	1
3.	Результат строго соответствует заданию, служит прямым ответом на соответствующий вопрос задания	1
4.	Представленный результат проходит проверку (тестирование) на разных наборах входных данных	1
5.	Студент дает правильные пояснения о том, как выполнялось задание и каковы его результаты	1
6.	Работа оформлена в соответствии с требованиями (структура, форматирование, подписи, комментарии в коде и т. п.)	1
7.	Алгоритм/решение выстроено последовательно, блоки кода или расчётов логически разделены и обоснованы	1
8.	Использованы оптимальные методы и алгоритмы; решение не содержит избыточных или неоправданно сложных шагов	1
9.	В работе (особенно в коде или расчётах) присутствуют понятные пояснения, облегчающие восприятие	1
10.	Студент уверенно отвечает на дополнительные вопросы по заданию, демонстрирует понимание смежных аспектов задачи	1
	Итоговая (суммарная) оценка	Max - 10

Оценка «зачтено» - 5 баллов и более; оценка «не зачтено» - менее 5 баллов.

4. Проверочная работа

Пример задания для проверочной работы

1. Создайте веб - приложение Node.js. Средствами виртуального DOM фреймворка React.js организуйте вывод списка на страницу сайта. Загрузку React-библиотек выполните из интернет - источников.

2. Создайте веб - приложение Node.js. Средствами модели DOMJS организуйте динамический вывод списка из 4-8 элементов на страницу. Ваша фамилия на странице обязательна.

Критерии оценивания проверочной работы:

№ п/п	Критерий	Балл
1.	Задание выполнено в срок (на соответствующем занятии)	1
2.	Задание выполнено студентом самостоятельно (без помощи и подсказок преподавателя)	1

3.	Результат строго соответствует заданию, служит прямым ответом на соответствующий вопрос задания	1
4.	Представленный результат проходит проверку (тестирование) на разных наборах входных данных	1
5.	Студент дает правильные пояснения о том, как выполнялось задание и каковы его результаты	1
6.	Работа оформлена в соответствии с требованиями (структура, форматирование, подписи, комментарии в коде и т. п.)	1
7.	Алгоритм/решение выстроено последовательно, блоки кода или расчётов логически разделены и обоснованы	1
8.	Использованы оптимальные методы и алгоритмы; решение не содержит избыточных или неоправданно сложных шагов	1
9.	В работе (особенно в коде или расчётах) присутствуют понятные пояснения, облегчающие восприятие	1
10.	Студент уверенно отвечает на дополнительные вопросы по заданию, демонстрирует понимание смежных аспектов задачи	1
	Итоговая (суммарная) оценка	Max - 10

Оценка «зачтено» - 5 баллов и более; оценка «не зачтено» - менее 5 баллов.

6.2. Оценочные средства и критерии оценивания для промежуточной аттестации

Форма промежуточной аттестации – **зачет**

Для определения уровня сформированности компетенций применяется процедура независимой оценки. Независимая оценка уровня сформированности компетенций осуществляется комиссией (с возможностью привлечения внешних экспертов со стороны работодателей). В состав комиссии входят 3 преподавателя: заведующий кафедрой – председатель комиссии и независимый эксперт, преподаватель, ведущий лабораторные занятия по дисциплине «Web-программирование», преподаватель кафедры – независимый эксперт. Зачет проходит в форме просмотра портфолио.

Требования к структуре и содержанию портфолио

Портфолио студента включает в себя следующие виды деятельности и их оценки:

№ п/п	Вид деятельности	Количество заданий	Пороговая оценка	Максимальная оценка
1.	Выполнение и защита заданий лабораторных работ	16 лабораторных работ	80	160
2.	Ответы на теоретические вопросы	4 блока вопросов	20	40
3.	Выполнение заданий для самостоятельной работы	5 заданий	25	50
4.	Проверочная работа	1 проверочная работа	5	10

Критерии и показатели оценивания портфолио по уровням

Уровень	Критерии и показатели оценивания
10 баллов	По результатам оценки портфолио набрано 93-100% от максимально возможного количества баллов, при этом по всем видам деятельности достигнуты пороговые значения
9 баллов	По результатам оценки портфолио набрано 86-92% от максимально возможного количества баллов, при этом по всем видам деятельности

	достигнуты пороговые значения
8 баллов	По результатам оценки портфолио набрано 79-85% от максимально возможного количества баллов, при этом по всем видам деятельности достигнуты пороговые значения
7 баллов	По результатам оценки портфолио набрано 71-78% от максимально возможного количества баллов, при этом по всем видам деятельности достигнуты пороговые значения
6 баллов	По результатам оценки портфолио набрано 64-70% от максимально возможного количества баллов, при этом по всем видам деятельности достигнуты пороговые значения
5 баллов	По результатам оценки портфолио набрано 57-63% от максимально возможного количества баллов, при этом по всем видам деятельности достигнуты пороговые значения
4 балла	По результатам оценки портфолио набрано 50-56% от максимально возможного количества баллов, при этом по всем видам деятельности достигнуты пороговые значения
3 балла	По результатам оценки портфолио набрано 36-49% от максимально возможного количества баллов и/или хотя бы по одному из видов деятельности не достигнуто пороговое значение
2 балла	По результатам оценки портфолио набрано 21-45% от максимально возможного количества баллов и/или хотя бы по одному из видов деятельности не достигнуто пороговое значение
1 балл	По результатам оценки портфолио набрано менее 21% от максимально возможного количества баллов и/или хотя бы по одному из видов деятельности не достигнуто пороговое значение

Таблица перевода оценок из 10-ти бальной системы в 5-ти бальную систему

Оценка по 10-ти бальной системе	Оценка по 5-ти бальной системе
10, 9, 8, 7,6,5,4	зачтено
3, 2, 1	не зачтено

7. Перечень основной и дополнительной учебной литературы

7.1. Основная литература

1. Сысолетин Е.Г., Ростунцев С.Д. Разработка интернет-приложений: учебник для вузов / под научной редакцией Л. Г. Доросинского. – М.: Издательство Юрайт, 2025. – 80 с. – URL: https://urait.ru/bcode/562916.

2. Лаврищева Е.М. Программная инженерия и технологии программирования сложных систем: учебник для вузов – М.: Издательство Юрайт, 2025. – 432 с. – URL: https://urait.ru/bcode/470923.

7.2. Дополнительная литература

1. Итан Браун. Веб-разработка с применением Node и Express. Полноценное использование стека JavaScript = Web Development with Node and Express / Итан Браун. — Санкт-Петербург: Питер, 2019. – 336 с.

2. Томас Марк Тиленс. React в действии. –Санкт-Петербург: Питер, 2019. – 368 с.

3. Брэд Дейли, Брендан Дейли, Калеб Дейли. Разработка веб-приложений с помощью Node.js, MongoDB и Angular: исчерпывающее руководство по использованию стека MEAN = Web Development with Node and Express. – Санкт-Петербург: «Диалектика-Вильямс», 2020. – 656 с.

7.3. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

1. Система дистанционного обучения СмолГУ (moodle.smolgu.ru).

2. Интернет-Университет Информационных Технологий. Web-технологии, учебный курс, https://intuit.ru/studies/courses/3523/765/info

8. Материально-техническое обеспечение

Учебная аудитория для проведения занятий семинарского типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации - компьютерная аудитория с выходом в Интернет.

Помещение для самостоятельной работы – компьютерный класс с доступом к сети «Интернет» и ЭИОС СмолГУ.

9. Программное обеспечение

KasperskyEndpointSecurity для бизнеса Стандартный АО «Лаборатория Касперского», лицензия 1FB6-161215-133553-1-6231.

Microsoft Open License, лицензия 49463448 всоставе: Microsoft Windows Professional 7 Russian; Microsoft Office 2010 Russian.

Node.js

РедакторAtom

MICROSOFT VISUAL STUDIO COMMUNITY 2019

СУБДSQLServer EXPRESS 2019