Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Максимов Алексей Борисович
Должность: директор департамента по образовательной политике
Дата подписания: 22.05.2024 17:01:23
Уникальный программный ключ:
8db180d1a3f02ac9e60521a5672742735c18b1d6

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное автономное образовательное учреждение
высшего образования
«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
Факультет информационных технологий

УТВЕРЖДАЮ
Декан факультета
«Информационные технологии»
_____ / Д.Г.Демидов /
«15» февраля 2024г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Веб-программирование и дизайн»

Направление подготовки/специальность
09.03.02 Информационные системы и технологии

Профиль/специализация
Автоматизированные системы обработки информации и управления
Информационные системы умных пространств
Информационные технологии в креативных индустриях
Программное обеспечение игровой компьютерной индустрии
Технологии дополненной и виртуальной реальности

Квалификация
Бакалавр

Формы обучения
Очная, заочная

Москва, 2024 г.

Разработчик(и):

ст. преподаватель кафедры
«Информатика и информационные технологии»  / П.С. Новиков /

Согласовано:

Заведующий кафедрой
«Информатика и информационные технологии»,
к.т.н.  / Е.В. Булатников /

Содержание

1. Цели, задачи и планируемые результаты обучения по дисциплине....4
2. Место дисциплины в структуре образовательной программы 4
3. Структура и содержание дисциплины....5
   3.1. Виды учебной работы и трудоемкость(по формам обучения)....5
   3.2. Тематический план изучения дисциплины....5
   3.3. Содержание дисциплины 8
   3.4. Тематика семинарских/практических и лабораторных занятий....9
   3.5. Тематика курсовых проектов (курсовых работ)....10
4. Учебно-методическое и информационное обеспечение....10
   4.1. Нормативные документы и ГОСТы....11
   4.2. Основная литература 11
   4.3. Дополнительная литература....11
   4.4. Электронные образовательные ресурсы....11
   4.5. Лицензионное и свободно распространяемое программноеобеспечение....11
   4.6. Современные профессиональные базы данных и информационные справочные системы 12
5. Материально-техническое обеспечение....12
6. Методические рекомендации 12
   6.1. Методические рекомендации для преподавателя по организацииобучения....12
   6.2. Методические указания для обучающихся по освоениюдисциплины 12
7. Фонд оценочных средств....13
   7.1. Методы контроля и оценивания результатов обучения 13
   7.2. Шкала и критерии оценивания результатов обучения 13
   7.3. Оценочные средства 14

1. Цели, задачи и планируемые результаты обучения по дисциплине

Целью освоения дисциплины «Веб-программирование и дизайн» является формирование понимания идеологии и ключевых аспектов создания одностраничных адаптивных веб-приложений на базе библиотеки React, и выполнения их дизайна на базе CSS-фреймворков, достаточного для практического использования в процессе дальнейшего обучения и в профессиональной сфере.

К основным **задачам** освоения дисциплины следует отнести:

- изучение библиотеки React
- изучение библиотеки Redux
- получение знаний достаточных, для создания одностраничных адаптивных веб-приложений и выполнение их дизайна на базе CSS-фреймворков.

Обучение по дисциплине «Веб-программирование и дизайн» направлено на формирование у обучающихся следующих компетенций:

Код и наименование компетенций	Индикаторы достижения компетенции
ОПК-6. Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий	ИОПК-6.1. знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий ИОПК-6.2. умеет применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ ИОПК-6.3. имеет навыки программирования, отладки и тестирования прототипов программно-технических комплексов задач

2. Место дисциплины в структуре образовательной программы

Дисциплина относится к модулю «Базовое программирование» обязательной части Блока 1. Дисциплины (модули) учебного плана программы бакалавриата.

Дисциплина взаимосвязана логически и содержательно-методически со следующими дисциплинами и практиками ОПОП:

- Введение в программирование
- Технологии прикладного программирования
- Объектно-ориентированное программирование
- Программирование для мобильных устройств
- Цифровые методы обработки информации
- Нейронные сети в креативных индустриях
- Веб-технологии
- Анимационная графика

- UI/UX-дизайн
- Технологии компьютерной верстки
- Видеомэппинг
- Генеративный дизайн
- Разработка игровых приложений
- Архитектура информационных систем
- Производственная практика (проектно-технологическая)
- Производственная практика (преддипломная)
- Выполнение и защита выпускной квалификационной работы

3. Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 4 зачетные единицы (144 часа).

3.1. Виды учебной работы и трудоемкость

3.1.1. Очная форма обучения

№ п/п	Вид учебной работы	Количество часов	Семестры
			4
1	**Аудиторные занятия**	**72**	**72**
	В том числе:		
1.1	Лекции	18	36
1.2	Семинарские/практические занятия		
1.3	Лабораторные занятия	54	54
2	**Самостоятельная работа**	**72**	**72**
	В том числе:		
2.1	Подготовка и выполнение лабораторных работ	72	72
3	**Промежуточная аттестация**		
	Зачет/диф.зачет/экзамен	**экзамен**	**экзамен**
	Итого:	**144**	**144**

3.1.2. Заочная форма обучения

№ п/п	Вид учебной работы	Количество часов	Семестры
			5
1	**Аудиторные занятия**	**16**	**16**
	В том числе:		
1.1	Лекции	8	8
1.2	Семинарские/практические занятия		
1.3	Лабораторные занятия	8	8
2	**Самостоятельная работа**	**128**	**128**
	В том числе:		
2.1	Подготовка и выполнение лабораторных работ	128	128
3	**Промежуточная аттестация**		
	Зачет/диф.зачет/экзамен	**экзамен**	**экзамен**
	Итого:	**144**	**144**

3.2. Тематический план изучения дисциплины

3.2.1. Очная форма обучения

		Трудоемкость, час		
			Аудиторная работа	

№ п/п	Разделы/темы дисциплины	Всего	Лекции	Семинарские/ практические занятия	Лабораторные занятия	Практическая подготовка	Самостоятельная работа
1.1	Тема 1. Введение в веб-программирование	4	2				2
1.2	Лабораторная работа № 1 «Подготовка и настройка инструментов для создания одностраничных приложений»	8			4		4
2.1	Тема 2. Введение Typescript и основы Typescript	4	2				2
3.1	Тема 3. Объекты в Typescript	4	2				2
4.1	Тема 4. ООП в Typescript	4	2				2
4.2	Лабораторная работа № 2 «Программирование на TypeScript»	8			4		4
5.1	Тема 5. Введение в создание одностраничных приложений	4	2				2
5.2	Лабораторная работа № 3 «Создание страницы интернет-магазина на основе функциональных компонентов»	8			4		4
5.3	Лабораторная работа № 4 «Создание страницы интернет-магазина на основе class-base компонентов»	8			4		4
6.1	Тема 6. Введение в React Bootstrap и Material UI	4	2				2
6.2	Лабораторная работа № 5 «Создание одностраничного приложения «Календарь», с использованием модульного подхода»	8			4		4
6.3	Лабораторная работа № 6 «Создания одностраничного приложения «Портфолио с фильтрами» и работа с состояниями»	8			4		4
7.1	Тема 7. Композиция компонентов, жизненный цикл и работа с HTTP, Хуки, Context API	4	2				2
7.2	Лабораторная работа № 9 «Создания одностраничного приложения «Выпадающий список» с применением форм»	8			4		4
7.3	Лабораторная работа № 10 «Создания одностраничного	8			4		4
	приложения «Список предложений» и работа с props»						
7.4	Лабораторная работа № 11 «Работа с CRUD»	8			4		4
7.5	Лабораторная работа № 12 «Функции высшего порядка»	8			4		4

7.6	Лабораторная работа № 13 «Работа с hooks»	8			4		4
8.1	Тема 8. Перегрузка методов и маршрутизация и Redux Route	4	2				2
8.2	Лабораторная работа № 14 «React Route»	8			4		4
9.1	Тема 9. Управление состояниями в React	4	2				2
9.2	Лабораторная работа № 15 «Redux Toolkit»	12			6		6
Итого		**144**	**18**		**54**		**72**

3.2.2. Заочная форма обучения

№ п/п	**Разделы/темы дисциплины**	**Трудоемкость, час**					
		Всего	**Аудиторная работа**				**Самостоятельная работа**
			Лекции	Семинарские/ практические занятия	Лабораторные занятия	Практическая подготовка	
1.1	Тема 1. Введение в веб-программирование	4,5	0,5				4
1.2	Лабораторная работа № 1 «Подготовка и настройка инструментов для создания одностраничных приложений»	6,5			0,5		6
2.1	Тема 2. Введение Typescript и основы Typescript	4,5	0,5				4
3.1	Тема 3. Объекты в Typescript	5	1				4
4.1	Тема 4. ООП в Typescript	5	1				4
4.2	Лабораторная работа № 2 «Программирование на TypeScript»	6,5			0,5		6
5.1	Тема 5. Введение в создание одностраничных приложений	5	1				4
5.2	Лабораторная работа № 3 «Создание страницы интернет-магазина на основе функциональных компонентов»	6,5			0,5		6
5.3	Лабораторная работа № 4 «Создание страницы интернет-магазина на основе class-base компонентов»	6,5			0,5		6
6.1	Тема 6. Введение в React Bootstrap и Material UI	5	1				4
6.2	Лабораторная работа № 5 «Создание одностраничного приложения «Календарь», с использованием модульного подхода»	6,5			0,5		6
6.3	Лабораторная работа № 6 «Создания одностраничного приложения «Портфолио с фильтрами» и работа с состояниями»	6,5			0,5		6

7.1	Тема 7. Композиция компонентов, жизненный цикл и работа с HTTP, Хуки, Context API	5	1				4
7.2	Лабораторная работа № 9 «Создания одностраничного приложения «Выпадающий список» с применением форм»	6,5			0,5		6
7.3	Лабораторная работа № 10 «Создания одностраничного приложения «Список предложений» и работа с props»	10,5			0,5		10
7.4	Лабораторная работа № 11 «Работа с CRUD»	6,5			0,5		6
7.5	Лабораторная работа № 12 «Функции высшего порядка»	6,5			0,5		6
7.6	Лабораторная работа № 13 «Работа с hooks»	7			1		6
8.1	Тема 8. Перегрузка методов и маршрутизация и Redux Route	7	1				6
8.2	Лабораторная работа № 14 «React Route»	7			1		6
9.1	Тема 9. Управление состояниями в React	7	1				6
9.2	Лабораторная работа № 15 «Redux Toolkit»	13			1		12
Итого		**144**	**8**		**8**		**128**

3.3. Содержание дисциплины

Тема 1. Введение в веб-программирование

- Введение
- Способы создания одностраничного веб-приложения
- Демонстрация основных возможностей библиотеки React

Тема 2. Введение Typescript и основы Typescript

- Различия между JS и TS
- Установка Node.js
- Компиляция программ на TS из командной строки
- Способы создания, запуска и отладки программ на TS с использованием сред программирования Visual Studio и Visual Studio Code
- Способы установки дополнительных библиотек, утилиты npm и yaml
- Переменные и константы
- Типы данных
- Функции – определение, параметры, результаты выполнения, необязательные параметры, Значения по умолчанию
- Тип функции, стрелочные функции, функция как параметр функции null и undefined

Тема 3. Объекты в Typescript

- Объекты
- Не обязательные свойства
- Объекты в функциях
- Оператор in
- Декомпозиция объектов

- Псевдонимы и ключевое слово type. Преобразование к типу
- Массивы и их декомпозиция
- Кортежи

Тема 4. ООП в Typescript

- Классы
- Наследование
- Интерфейсы
- Обобщения

Тема 5. Введение в создание одностраничных приложений

- Общие сведения о React
- DOM-дерево
- Классовые и функциональные компоненты
- Введение в JSX
- Props в React и Typescript

Тема 6. Введение в React Bootstrap и Material UI

- Установка
- Особенности использования
- Ключевые компоненты
- Способы создания одностраничных приложений с использованием библиотек React Bootstrap и Material UI
- Введение в формы
- Контролируемы компоненты
- Не контролируемые компоненты

Тема 7. Композиция компонентов, жизненный цикл и работа с HTTP, Хуки, Context API

- Props
- Библиотека fetch
- Библиотека axios
- XHR
- Понятия hook в react
- Основные возможности и назначения hook
- Contex API

Тема 8. Перегрузка методов и маршрутизация и Redux Route

- Перезагружаемы методы и способы работы с ними
- Способы маршрутизации и библиотека react-route-dom

Тема 9. Управление состояниями в React

- Введение в управление состояниями
- Библиотека Redux
- Особенности использования библиотеки Redux Toolkit
- RTK Query

3.1. Тематика семинарских/практических и лабораторных занятий

3.1.1. Семинарские/практические занятия

Семинарские и практические занятия не предусмотрены.

3.1.2. Лабораторные занятия

Лабораторная работа № 1 «Подготовка и настройка инструментов для создания одностраничных приложений»

В данной работе рассмотрено, как подготовить программную среду для создания одностраничных приложений. Приведены способы установки IDE, ее настройки, установки необходимых плагинов. Проведены рекомендации по способом создания приложения и установки в него библиотек.

Лабораторная работа № 2 «Программирование на TypeScript»

В данной работе рассмотрено, создание программ на **Typescript**, способы ее компиляции и отладки. Вырабатываются навыки применения программных инструментов входящих в язык программирования

Лабораторная работа № 3 «Создание страницы интернет-магазина на основе функциональных компонентов»

В данной работе изучаются способы создания простых одностраничных веб-приложения, на основе функциональных компонентов.

Лабораторная работа № 4 «Создание страницы интернет-магазина на основе class-base компонентов»

В данной работе изучаются способы создания простых одностраничных веб-приложения, на основе объектно-ориентированных компонентов.

Лабораторная работа № 5 «Создание одностраничного приложения «Календарь», с использованием модульного подхода»

В данной работе изучается способы разбиения программы на основе библиотеки React на модули, и методология взаимодействия между ними.

Лабораторная работа № 6 «Создания одностраничного приложения «Портфолио с фильтрами» и работа с состояниями»

В данной работе изучается работа с событиями и состояниями.

Лабораторная работа № 7 «Создания одностраничного приложения «Расположение товаров» и применение событий к JSX»

В данной работе изучается работа с событиями и состояниями применительно к JSX разметке.

Лабораторная работа № 9 «Создания одностраничного приложения «Выпадающий список» с применением форм»

В данной работе рассматривается возможности библиотеки React по работе с формами.

Лабораторная работа № 10 «Создания одностраничного приложения «Список предложений» и работа с props»

В данной работе вырабатывается навык по работе с Props.

Лабораторная работа № 11 «Работа с CRUD»

В данной работе рассматривается жизненный цикл компонента, его фазы, вырабатывается навыки загрузки данных в веб-приложении.

Лабораторная работа № 12 «Функции высшего порядка»

В данной работе происходит знакомство с концепцией чистой функции, с подходом функции высшего порядка. Вырабатываются навыки создания компонентов высшего порядка.

Лабораторная работа № 13 «Работа с hooks»

В данной работе происходит знакомство с hook в React, рассматриваются основные возможности и назначения hooks, вырабатываются навыки работы с Contex API.

Лабораторная работа № 14 «React Route»

В данной работе происходит знакомство с концепцией управления роутинга в react, рассматриваются основные инструменты роутинга, вырабатываются навыки применения библиотеки react-route-dom.

Лабораторная работа № 15 «Redux Toolkit»

В данной работе происходит знакомство с концепцией управления состояний в React, применение библиотеки Redux и вырабатываются навыки ее практического применения.

3.2. Тематика курсовых проектов (курсовых работ)

Курсовой проект не предусмотрен

4. Учебно-методическое и информационное обеспечение

4.1. Нормативные документы и ГОСТы

1. https://fgos.ru/fgos/fgos-01-03-02-prikladnaya-matematika-i-informatika-9/2 .

"Положения об организации образовательного процесса в Московском Политехническом университете"

4.2. Основная литература

1. Ефромеев, Н. М. Основы web-программирования : учебное пособие / Н. М. Ефромеев, Е. В. Ефромеева. — Саратов : Вузовское образование, 2019. — 128 c. — ISBN 978-5-4487-0529-8. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/86300.html (дата обращения: 23.03.2024)
2. Поляков, Е. А. Web-дизайн : учебное пособие / Е. А. Поляков. — Саратов : Вузовское образование, 2019. — 188 c. — ISBN 978-5-4487-0489-5. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/81868.html (дата обращения: 23.03.2024)
3. Документация по React
4. Документация по Redux
5. Алекс Бэнкс, Ева Порселло ; [перевел с английского С. Черников]. - 2-е изд. - Санкт-Петербург [и др.] : Питер ; Минск : Питер, 2022. - 315, [2] с. : ил.; 23 см.; ISBN 978-5-4461-1492-4 : 700 экз. Изучаем React : практическое руководство по созданию веб-приложений при помощи React и Redux : [12+] / Кируп Чиннатамби ; пер. с англ. М. А. Райтмана. - 2-е изд. - Москва : Эксмо, 2019. - 365 с. : ил., цв. ил.; 25 см. - (Мировой компьютерный бестселлер).;ISBN 978-5-04-098028-4 : 2000 экз.
6. Изучаем TypeScript 3 : создавайте промышленные веб-приложения корпоративного класса с использованием TypeScript 3 и современных фреймворков : [перевод с английского] / Натан Розенталс. - Москва : ДМК Пресс, 2019. - 623 с. : ил., табл.; 24 см.; ISBN 978-5-97060-757-2 : 200 экз.
7. React и Redux [Текст] : функциональная веб-разработка : [16+] / Алекс Бэнкс, Ева Порселло ; [перевод с английского Н. Вильчинского]. - Санкт-Петербург : Питер ; Минск : Питер, 2019. - 335 с. : ил., табл.; 23 см. - (Серия "Бестселлеры O'Reilly").; ISBN 978-5-4461-0668-4

4.3. Дополнительная литература

1. Маркин, А. В. Web-программирование : учебник / А. В. Маркин. — Москва : Ай Пи Ар Медиа, 2021. — 286 c. — ISBN 978-5-4497-1002-4. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/104883.html (дата обращения: 23.03.2024).
2. Гумерова, Л. З. Основы web-программирования : учебное пособие / Л. З. Гумерова. — Красноярск : Научно-инновационный центр, 2019. — 104 c. — ISBN 978-5-6042232-6-0. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/97112.html (дата обращения: 23.03.2024)
3. TypeScript быстро : [16+] / Яков Файн, Антон Моисеев ; [перевел с английского Д. Акуратер]. - Санкт-Петербург [и др.] : Питер, 2021. - 524 с. : ил., табл.; 24 см. (Серия "Для профессионалов").; ISBN 978-5-4461-1725-3 : 700 экз.
4. React 17. Разработка веб-приложений на JavaScript / Владимир Дронов. - Санкт-Петербург : БХВ-Петербург, 2022. - 384 с. : ил.; 23 см. - (Профессиональное программирование).; ISBN 978-5-9775-9683-1 : 1000 экз.

4.4. Электронные образовательные ресурсы

ЭОР в системе СДО «Веб-программирование и дизайн»:
https://online.mospolytech.ru/course/view.php?id=13533

4.5. Лицензионное и свободно распространяемое программное

обеспечение

1. Visual Studio
2. Visual Studio Code
3. Node.js

4.6. Современные профессиональные базы данных и информационные справочные системы

1. ОП "Юрайт" https://urait.ru/
2. IPR Smart https://www.iprbookshop.ru/
3. ЭБС "Лань" https://e.lanbook.com/

5. Материально-техническое обеспечение

Компьютерные классы со следующей оснащенностью: столы, стулья, аудиторная доска, использование переносного мультимедийного комплекса (переносной проектор, персональный ноутбук). Персональные компьютеры, мониторы, мышки, клавиатуры. Рабочее место преподавателя: стол, стул.

Программное обеспечение: Microsoft Windows или Linux на основе deb-пакетов (Debian, Ubuntu, Astra и т.д.), сервер с системой контроля версий GIT (GitLab)

6. Методические рекомендации

6.1. Методические рекомендации для преподавателя по организации обучения

Методика преподавания дисциплины «Веб-программирование и дизайн» предусматривает использование онлайн-курса в системе дистанционного обучения Университета, групповых и индивидуальных консультаций обучающихся, аудиторных занятий в сочетании с внеаудиторной работой с целью формирования и развития профессиональных навыков обучающихся.

Лабораторные работы по дисциплине «Веб-программирование и дизайн» осуществляются в форме самостоятельной проработки теоретического материала обучающимися; выполнения практического задания; защиты преподавателю лабораторной работы (знание теоретического материала и выполнение практического задания по теме лабораторной работы).

6.2. Методические указания для обучающихся по освоению дисциплины

Изучение дисциплины осуществляется в соответствии с учебным планом.

На занятиях осуществляется закрепление полученных, в том числе и в процессе самостоятельной работы, знаний. Особое внимание обращается на умение применять полученные знания на практике, в том числе при решении реальных задач, отличающихся от проработанных.

В процессе самостоятельной работы студенты закрепляют и углубляют знания, полученные во время аудиторных занятий, самостоятельно знакомятся с теоретическим материалом, выполняют лабораторные работы, готовятся к текущему контролю и промежуточной аттестации.

Выполнение и сдача лабораторных работ проводится с применением системы контроля версий GIT.

Текущий контроль осуществляется на аудиторных занятиях в виде защиты лабораторных работ. Критериями оценки результатов являются:

- уровень освоения теоретического материала;
- уровень владения практическими навыками (в виде вопросов по процессу

выполнения лабораторных работ);

– умения обучающегося использовать теоретические знания при выполнении практических задач (в виде дополнительных заданий);

– сформированность компетенций;

– оформление материала в соответствии с требованиями.

Промежуточный контроль осуществляется на зачете в форме тестирования в системе дистанционного обучения Университета, включающего вопросы на знание практической части языка Typescript, библиотеки React, а также другого материала входящего в дисциплину.

7. Фонд оценочных средств

7.1. Методы контроля и оценивания результатов обучения

В процессе обучения используются следующие оценочные формы самостоятельной работы студентов, оценочные средства текущего контроля успеваемости и промежуточных аттестаций: **лабораторные работы**, **зачёт**.

7.2. Шкала и критерии оценивания результатов обучения

К промежуточной аттестации допускаются только студенты, выполнившие все виды учебной работы, предусмотренные рабочей программой по дисциплине «Веб-программирование и дизайн».

7.2.1. Критерии оценки ответа на экзамене

(формирование компетенций — ОПК-6)

«Отлично»:

Выполнены все виды учебной работы, предусмотренные учебным планом. Обучающийся демонстрирует прочные теоретические знания, практические навыки, владеет терминами, делает аргументированные выводы и обобщения, приводит примеры, оперирует приобретенными знаниями, умениями, навыками, применяет их в ситуациях повышенной сложности. При этом могут быть допущены незначительные ошибки, неточности, которые обучающийся может исправить самостоятельно.

«Хорошо»:

Выполнены все виды учебной работы, предусмотренные учебным планом. Обучающийся демонстрирует достаточные теоретические знания, практические навыки, владеет терминами, делает аргументированные выводы и обобщения, приводит примеры, оперирует приобретенными знаниями, умениями, навыками. При этом могут быть допущены незначительные ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации, которые обучающийся может исправить при незначительной коррекции преподавателем.

«Удовлетворительно»:

Выполнены все виды учебной работы, предусмотренные учебным планом. Обучающийся демонстрирует неполное соответствие теоретических знаний, практических навыков, владеет терминами, делает аргументированные выводы и обобщения, приводит примеры, оперирует приобретенными знаниями, умениями, навыками. При этом могут быть допущены ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации, которые обучающийся может исправить при коррекции преподавателем.

«Неудовлетворительно»:

Не выполнен один или более видов учебной работы, предусмотренных учебным планом. Обучающийся демонстрирует незнание теоретических основ предмета, отсутствие практических навыков, не умеет делать аргументированные выводы и приводить примеры, не владеет терминами, проявляет отсутствие логичности и последовательности изложения, делает ошибки, которые не может исправить даже при коррекции преподавателем,

отказывается отвечать на дополнительные вопросы, допускает значительные ошибки, испытывает значительные затруднения при оперировании знаниями и умениями при их переносе на новые ситуации.

7.2.2. Критерии оценки работы обучающегося на лабораторных занятиях:
(формирование компетенций — ОПК-6)

«5» (отлично): выполнены все практические задания, предусмотренные лабораторными работами, обучающийся четко и без ошибок ответил на все контрольные вопросы, проявил творческий подход при выполнении заданий, смог выполнить дополнительные задания.

«4» (хорошо): выполнены все практические задания, предусмотренные лабораторными работами, обучающийся с корректирующими замечаниями преподавателя ответил на все контрольные вопросы, проявил творческий подход при выполнении заданий, смог частично выполнить дополнительные задания.

«3» (удовлетворительно): выполнены все практические задания, предусмотренные лабораторными работами, с замечаниями преподавателя; обучающийся ответил на все контрольные вопросы с замечаниями, дополнительные задания выполнены с замечаниями.

«2» (неудовлетворительно): обучающийся не выполнил или выполнил неправильно практические задания, предусмотренные лабораторными работами, обучающийся ответил на контрольные вопросы с ошибками или не ответил на контрольные вопросы, дополнительные задания выполнены неверно или не выполнены.

7.3. Оценочные средства

7.3.1. Текущий контроль

Текущий контроль осуществляется на аудиторных занятиях в виде защиты лабораторных работ. Лабораторная работа – средство контроля усвоения учебного материала темы, раздела или разделов дисциплины, организованное как учебное занятие в виде демонстрации полученных навыков при решении поставленных практических задач.

Примеры вопросов к защите лабораторных работ (оцениваемые компетенции — ОПК-6).

Лабораторная работа № 1 «Подготовка и настройка инструментов для создания одностраничных приложений»

Примеры вопросов:

1. Опишите способы создания приложений базе библиотеке React. Укажите различия.
2. В чем разница создания приложения между скриптом create-react-app и библиотеки vite?
3. Опишите способы подключения дополнительных библиотек к приложению.

Лабораторная работа № 2 «Программирование на TypeScript»

Примеры вопросов:

1. В чем отличие между JavaScript и TypeScript? Опишите достоинства и недостатки языков.
2. Как описываются переменные и константы в языке TypeScript. Приведите примеры.
3. Для чего используются кортежи в языке TypeScript? Опишите способы декомпозиции кортежей.

Лабораторная работа № 3 «Создание страницы интернет-магазина на основе функциональных компонентов»

Примеры вопросов:

1. В чем разница между функциональными и классовыми компонентами? Опишите различия, укажите достоинства и недостатки.
2. Какой тип программирования используется при создании приложения на основе функциональных компонентов? Опишите особенности.
3. Опишите использование переменных и констант в функциональных компонентах.

Лабораторная работа № 4 «Создание страницы интернет-магазина на основе class-base компонентов»

Примеры вопросов:

1. Опишите особенности создания классов для библиотеки React
2. В каких случаях лучше использовать классы?
3. Опешите систему обновления компонентов при использовании class-base подхода

Лабораторная работа № 5 «Создание одностраничного приложения «Календарь», с использованием модульного подхода»

Примеры вопросов:

1. В чем особенность разбиения программы на модули?
2. Как осуществляется разбивка программы на модули?
3. Опишите структуру организации модулей.

Лабораторная работа № 6 «Создания одностраничного приложения «Портфолио с фильтрами» и работа с состояниями»

Примеры вопросов:

1. Для чего нужны состояния?
2. Как работает DOM-дерево?
3. Как происходит обновление компонентов

Лабораторная работа № 7 «Создания одностраничного приложения «Расположение товаров» и применение событий к JSX»

Примеры вопросов:

1. Что такое JSX? Как JSX применяется?
2. Как события влияют на JSX компонент?
3. Дайте примеры включения JSX компонента.

Лабораторная работа № 9 «Создания одностраничного приложения «Выпадающий список» с применением форм»

Примеры вопросов:

1. Как подключается библиотека React Bootstrap?
2. Как подключается библиотека Material UI?
3. Опишите создания формы на примере.

Лабораторная работа № 10 «Создания одностраничного приложения «Список предложений» и работа с props»

Примеры вопросов:

1. Что такое props?
2. Для чего применяется props?
3. Как передать состояния между модулями используя props?

4.

Лабораторная работа № 11 «Работа с CRUD»

Примеры вопросов:

1. Опишите способы загрузки данных в приложения?
2. Опишите на примере использование библиотеки fetch.
3. Опишите на примере использование библиотеки axios

Лабораторная работа № 12 «Функции высшего порядка»

Примеры вопросов:

1. Что такое функция высшего порядка?
2. Что такое чистая функция?
3. Опишите применение функции высшего порядка.

Лабораторная работа № 13 «Работа с hooks»

Примеры вопросов:

1. Что такое hooks? Как используется?
2. Какие стандартные хуки вы знаете? Опишите их применение.
3. Как создать собственный хук? Приведите пример.

Лабораторная работа № 14 «React Route»

Примеры вопросов:

1. Как осуществляется маршрутизация в одностраничном приложении?
2. Как задать маршруты в одностраничном приложении? Приведите пример.
3. Как осуществляются переходы между страницами? Приведите пример.

Лабораторная работа № 15 «Redux Toolkit»

Примеры вопросов:

1. Для чего используется библиотека Redux?
2. Опишите создания состояния с помощью библиотек Redux.
3. Опишите способы доступа к состояниям, хранящимся в библиотекеRedux/

7.3.2. Промежуточная аттестация

Промежуточная аттестация обучающихся в форме экзамена осуществляется по результатам выполнения всех видов учебной работы, предусмотренных учебным планом по данной дисциплине, при этом учитываются результаты текущего контроля успеваемости в течение семестра. Экзамен проводится в форме выполнения практического задания и проведение устного опроса включающей вопросы на знание библиотеке React и языка TypeScript, а также создания одностраничных приложений на их основе. По итогам промежуточной аттестации по дисциплине выставляется оценка.

Примеры заданий:

1. Необходимо реализовать меню для сайта гоночного такси с пунктами «Главная», «Дрифт-такси», «Time Attack» и «Forza Karting». При переходе настраницу соответствующий пункт меню должен подсветиться.
2. Необходимо реализовать один из компонентов чата — историю сообщений.Для отображения списка создайте компонент MessageHistory, который

   принимает следующие атрибуты: list — список сообщений, массив

объектов,по умолчанию пустой массив. Если список сообщений пуст, то компонент не должен иметь какого-либо представления в DOM.

3. Вам необходимо реализовать небольшое приложение, которое хранит данныео тренировках и прогулках, которые вы совершаете в течение недели.
   Реализуйте формы ввода, добавление, удаление и редактированиеинформации.