

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное образовательное учреждение высшего образования  
«Национальный исследовательский ядерный университет «МИФИ»  
**Новоуральский технологический институт**  
филиал федерального государственного автономного образовательного учреждения высшего  
образования «Национальный исследовательский ядерный университет «МИФИ»  
**(НТИ НИЯУ МИФИ)**  
**Колледж НТИ**

---

Цикловая методическая комиссия информационных технологий

ОДОБРЕНО

Учёным Советом НТИ НИЯУ МИФИ

Протокол № 5 от 02 сентября 2025 г.

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ**  
**ПОПРОФЕССИОНАЛЬНОМУ МОДУЛЮ ПМн.03**  
**«РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ НА СТОРОНЕ**  
**КЛИЕНТА»**

для студентов колледжа НТИ НИЯУ МИФИ,  
обучающихся по программе среднего профессионального образования

специальность 09.02.09

«Веб-разработка»

очная форма обучения

на базе основного общего образования

квалификация

разработчик веб-приложений

Новоуральск 2025

РАССМОТРЕНО:  
на заседании  
цикловой методической комиссии  
информационных технологий  
Протокол № 8 от 01.09.2025 г.

Разработан на основе рабочей  
программы профессионального  
модуля ПМн.03 «Разработка веб-  
приложения на стороне клиента»  
по специальности 09.02.09 «Веб-  
разработка»

Председатель ЦМК ИТ

 И.И. Горницкая

Фонд оценочных средств по профессиональному модулю ПМн.03 «Разработка веб-приложения на стороне клиента» – Новоуральск: Изд-во колледжа НТИ НИЯУ МИФИ, 2025. – 51с.

## АННОТАЦИЯ

Фонд оценочных средств предназначен для текущего контроля и промежуточной аттестации обучающихся по специальности 09.02.09 Веб-разработка на соответствие их персональных достижений поэтапным требованиям программы подготовки специалистов среднего звена по профессиональному модулю ПМн.03 «Разработка веб-приложения на стороне клиента». Комплектация фонда оценочных средств: паспорт, программа оценивания, оценочные средства текущего контроля и промежуточной аттестации, критерии оценивания. В паспорте фонда оценочных средств указаны: место профессионального модуля в структуре программы подготовки специалистов среднего звена, требования ФГОС СПО к результатам освоения профессионального модуля, перечень формируемых компетенций, компоненты фонда оценочных средств.

Разработчики:

И.И. Горницкая, преподаватель, председатель ЦМК информационных технологий

А.Н. Лебедева, преподаватель ЦМК информационных технологий

А.В. Тарасова, преподаватель ЦМК информационных технологий,  
администратор информационной безопасности НТИ НИЯУ МИФИ

Редактор: Горницкая И.И.

## СОДЕРЖАНИЕ

1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ ПО ПРОФЕССИОНАЛЬНОМУ МОДУЛЮ ПМн.03 «РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ НА СТОРОНЕ КЛИЕНТА» .....	4
2. ПРОГРАММА ОЦЕНИВАНИЯ МОДУЛЯ ПМн.03 «РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ НА СТОРОНЕ КЛИЕНТА» .....	8
3. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕГО КОНТРОЛЯ ПРОФЕССИОНАЛЬНОМУ МОДУЛЮ .....	9
3.1 Тестовая форма контроля .....	11
3.2 Критерии оценки результатов выполнения тестовых заданий .....	14
3.3 Письменная форма контроля .....	15
3.4. Критерии оценки результатов выполнения письменных заданий .....	33
4. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ .....	35
4.1 Критерии оценивания знаний, обучающихся на дифференцированном зачете .....	35
4.2 Критерии оценивания знаний, обучающихся при защите результатов курсового проектирования .....	44

# 1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ ПО ПРОФЕССИОНАЛЬНОМУ МОДУЛЮ ПМн.03 «РАЗРАБОТКА ВЕБ- ПРИЛОЖЕНИЯ НА СТОРОНЕ КЛИЕНТА»

Фонд оценочных средств является составной частью учебно-методических документов, обеспечивающих реализацию основной образовательной программы СПО по специальности 09.02.09 Веб-разработка.

Фонд оценочных средств предназначен для текущего контроля и промежуточной аттестации по профессиональному модулю ПМн.03 «Разработка веб-приложения на стороне клиента»

**1.1. Цель и планируемые результаты освоения профессионального модуля:** Модуль входит в профессиональный цикл, направлен освоение основных видов профессиональной деятельности.

## 1.1.1 Общие компетенции.

Код	Наименование общих компетенций
<b>ОК 01.</b>	Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам;
<b>ОК 02.</b>	Использовать современные средства поиска, анализа и интерпретации информации, и информационные технологии для выполнения задач профессиональной деятельности;
<b>ОК 03.</b>	Планировать и реализовывать собственное профессиональное и личностное развитие, предпринимательскую деятельность в профессиональной сфере, использовать знания по финансовой грамотности в различных жизненных ситуациях;
<b>ОК 04.</b>	Эффективно взаимодействовать и работать в коллективе и команде
<b>ОК 05.</b>	Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста;
<b>ОК 06.</b>	Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей, в том числе с учетом гармонизации межнациональных и межрелигиозных отношений, применять стандарты антикоррупционного поведения;

<b>ОК 07.</b>	Содействовать сохранению окружающей среды, ресурсосбережению, применять знания об изменении климата, принципы бережливого производства, эффективно действовать в чрезвычайных ситуациях;
<b>ОК 08.</b>	Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности;
<b>ОК 09.</b>	Пользоваться профессиональной документацией на государственном и иностранном языках.

### Профессиональные компетенции

Код	Наименование видов деятельности и профессиональных компетенций
<b>ВД 3</b>	Разработка веб приложения на стороне клиента
<b>ПК 3.1.</b>	Проектировать структуры разделов информационных ресурсов с целью создания эскиза и прототипа интерфейса пользователя.
<b>ПК 3.2.</b>	Разрабатывать интерфейс пользователя для информационных ресурсов с использованием стандартов в области веб-разработки.
<b>ПК 3.3.</b>	Создавать структуру кода веб-страницы информационных ресурсов в соответствии с дизайн-макетом.
<b>ПК 3.4.</b>	Создавать программный код на стороне клиента в соответствии с техническим заданием (спецификацией) с использованием языков программирования, библиотек и фреймворков.

В результате освоения профессионального модуля студент должен:

Владеть навыками	<ul style="list-style-type: none"> <li>– разработки эскизов, схем, прототипов интерфейса пользователя информационного ресурса;</li> <li>– разработки дизайна компонентов интерфейса пользователя в соответствии со стандартами и требованиями заказчика;</li> <li>– разработки программного кода веб-страниц информационного ресурса, в том числе с использованием готовых технических решений;</li> <li>– разработки кроссбраузерной верстки веб-страниц информационного ресурса;</li> </ul>
------------------	---

	<ul style="list-style-type: none"> <li>– разработки клиентской части web-приложения в соответствии с техническим заданием (спецификацией).</li> </ul>
Уметь	<ul style="list-style-type: none"> <li>– применять программные средства для разработки интерфейса;</li> <li>– применять инструменты для оценки эффективности и удобства созданного интерфейса;</li> <li>– применять полученные данные для оптимизации интерфейса;</li> <li>– применять нормативные документы, определяющие требования к оформлению интерфейсов информационных ресурсов;</li> <li>– создавать адаптивный интерфейс web-ресурса;</li> <li>– применять специализированное программное обеспечение для верстки страниц информационных ресурсов;</li> <li>– использовать язык разметки страниц информационных ресурсов;</li> <li>– применять выбранные языки программирования для написания программного кода; <ul style="list-style-type: none"> <li>– использовать выбранную среду программирования и средства системы управления базами данных;</li> <li>– использовать возможности имеющейся программной архитектуры информационного ресурса.</li> </ul> </li> </ul>
Знать	<ul style="list-style-type: none"> <li>– современные принципы построения интерфейсов пользователя;</li> <li>– основные требования, предъявляемые к дизайну графических интерфейсов;</li> <li>– способы представления информации с учетом особенностей пользователя: возрастных, особенностей ограниченных возможностей здоровья и др.;</li> <li>– особенности отображения элементов интерфейсов web-ресурсов в различных браузерах;</li> <li>– правила реализации адаптивного интерфейса web-ресурса;</li> <li>– методы повышения читаемости программного кода;</li> <li>– синтаксис выбранного языка программирования, особенностей программирования на этом языке, стандартных библиотек языка программирования;</li> <li>– отраслевую нормативную техническую документацию;</li> </ul>

- |  |  |
|--|--|
|  | <ul style="list-style-type: none"><li>– особенности выбранной среды программирования;</li><li>– компоненты программно-технических архитектур информационных ресурсов, существующих приложений и интерфейсов взаимодействия с ними;</li><li>– сетевые протоколы и основы web-технологий;</li><li>– современные стандарты взаимодействия компонентов распределенных приложений;</li><li>– программные средства и платформы для разработки web-ресурсов;</li><li>– основы информационной безопасности web-ресурсов.</li></ul> |
|--|--|

**2. ПРОГРАММА ОЦЕНИВАНИЯ МОДУЛЯ ПМн.03 «РАЗРАБОТКА  
ВЕБ-ПРИЛОЖЕНИЯ НА СТОРОНЕ КЛИЕНТА»**

№ П/П	КОНТРОЛИРУЕМЫЕ ЭЛЕМЕНТЫ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ	КОНТРОЛИРУЕМЫЕ КОМПЕТЕНЦИИ	ФОРМА АТТЕСТАЦИИ
<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>
1	МДКн 03.01 Проектирование и дизайн интерфейсов	ПК 3.1, ПК 3.2 ОК 1-9	IV семестр дифференцированн ый зачет
2	МДКн 03.02 Верстка страниц	ПК 3.2, ПК 3.3 ОК 1-9	V семестр дифференцированн ый зачет
3			V семестр защита курсового проекта
4	МДКн 03.03 Разработка клиентской части информационных ресурсов	ПК 3.4 ОК 1-9	VI семестр дифференцированн ый зачет
5			VI семестр дифференцированн ый зачет
6	УП 02.01 Учебная практика	ПК 3.1, ПК 3.2, ПК 3.3, ПК 3.4 ОК 1-9	VI семестр комплексный дифференцированн ый зачет
7	ПП 02.01 Производственная практика	ПК 3.1, ПК 3.2 ПК 3.3, ПК 3.4	
8	ПМн 03.	ПК 3.1, ПК 3.2 ПК 3.3, ПК 3.4 ОК 1-9	Экзамен по модулю

### **3. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕГО КОНТРОЛЯ ПРОФЕССИОНАЛЬНОМУ МОДУЛЮ**

Текущий контроль знаний, умений и навыков осуществляется в ходе образовательного процесса по всем изучаемым междисциплинарным курсам в формах, определяемых преподавателем в рабочей программе профессионального модуля. Он может проводиться в ходе всех видов занятий в форме, избранной преподавателем.

Текущий контроль успеваемости может проводиться в следующих формах:

- письменная (письменный опрос, выполнение практических работ, контрольных и других проверочных работ и т. д.);
- тестовая (устное, письменное, компьютерное тестирование).

Возможны и другие виды текущего контроля успеваемости, которые определяются преподавателями.

Система текущего контроля успеваемости предусматривает разнообразные по форме и содержанию контрольные мероприятия, учитывающие все виды аудиторной и самостоятельной учебной деятельности обучающегося.

Данные текущего контроля используются преподавателями для анализа (мониторинга) качества освоения обучающимися основной образовательной программы среднего образования, для организации консультаций.

<b>ФОРМА КОНТРОЛЯ</b>	<b>МДКн 03.01</b>	<b>МДКн 03.02</b>	<b>МДКн 03.03</b>
тестовая	Тема 1.1 Проектирование и дизайн интерфейсов Тест№1, 2	Тема 2.1 Верстка страниц Тест№1,2,3,4,5	Тема 3.1 Разработка клиентской части информационных ресурсов Тест№1
письменная	<p>Практическое занятие № 1. «Создание оригинал-макетов, элементов дизайна сайта»</p> <p>Практическое занятие № 2. «Создание макета сайта, буклета»</p> <p>Практическое занятие № 3. «Разработка схемы интерфейса веб-приложения»</p>	<p>Практическое занятие № 1. «Применение тегов HTML при создании web-страниц»</p> <p>Практическое занятие № 2. «Форматирование web-страниц с использованием каскадных таблиц стилей»</p> <p>Практическое занятие № 3. «Вёрстка Landing page»</p> <p>Практическое занятие № 4. «Вёрстка для мобильных устройств»</p> <p>Практическое занятие № 5. «Применение фреймворков при вёрстке»</p> <p>Самостоятельная работа №1. Верстка веб-страниц</p>	<p>Практическое занятие № 1. «Создание клиентских сценариев средствами JavaScript»</p> <p>Практическое занятие № 2. «Использование встроенных структур данных»</p> <p>Практическое занятие № 3. «Создание скриптов, используя особенности замыканий»</p> <p>Практическое занятие № 4. «Написание приложения с применением ООП»</p> <p>Практическое занятие № 5. «Написание приложения используя фреймворки и библиотеки»</p>

### 3.1 Тестовая форма контроля

МДК.03.01. Проектирование и дизайн интерфейсов

Тема 1.1. Проектирование и дизайн интерфейсов.

Изучаемые понятия:

- Особенности векторной графики. Редактор векторной графики
- Особенности растровой графики. Редактор растровой графики
- Редактор разработки мультимедийного контента
- Особенность разработки интерфейса.
- Визуализация элементов интерфейса
- Юзабилити web-сайтов и приложений для мобильных устройств.

Тест №1. Особенности разработки интерфейса

Тузовский, А. Ф. Проектирование и разработка web-приложений : учебник для среднего профессионального образования / А. Ф. Тузовский. — Москва : Издательство Юрайт, 2025. — 219 с. — (Профессиональное образование). — ISBN 978-5-534-16767-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. с. 96 — URL: <https://urait.ru/bcode/565693>

<https://urait.ru/quiz/run-test/EE3379BF-59B4-4A20-989F-F737F8166384/119C3C0F-A76B-4AE3-94F1-41C341845826>

Тест №2. Юзабилити веб-сайтов и приложений

Тузовский, А. Ф. Проектирование и разработка web-приложений : учебник для среднего профессионального образования / А. Ф. Тузовский. — Москва : Издательство Юрайт, 2025. — 219 с. — (Профессиональное образование). — ISBN 978-5-534-16767-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. с. 96 — URL: <https://urait.ru/bcode/565693>

<https://urait.ru/quiz/run-test/243654AF-4C2A-4DB5-A981-CF2F87A114F7/C0F3CB95-F06B-4942-AABB-EFEDA3473E79>

Тест №3. Особенности организации разработки веб-приложений.

Полуэктова, Н. Р. Разработка веб-приложений : учебник для среднего профессионального образования / Н. Р. Полуэктова. — 2-е изд. — Москва : Издательство Юрайт, 2025. — 204 с. — (Профессиональное образование). — ISBN 978-5-534-18644-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. с. 28 — URL: <https://urait.ru/bcode/567621>  
<https://urait.ru/quiz/run-test/599CC9CE-D0A4-420D-A48F-D21C0085D311/5985FE96-A88F-484E-A79B-8082245D6834>

МДК.03.02. Верстка страниц

Тема 2.1. Верстка страниц

Изучаемые понятия:

- Язык разметки HTML. Синтаксис и семантика HTML
- Каскадные таблицы стилей (CSS)
- Веб-стандарты и их поддержка
- HTML-фреймворки
- CSS-фреймворки. Динамический CSS.

Тест №1. Верстка страниц

Тузовский, А. Ф. Проектирование и разработка web-приложений : учебник для среднего профессионального образования / А. Ф. Тузовский. — Москва : Издательство Юрайт, 2025. — 219 с. — (Профессиональное образование). — ISBN 978-5-534-16767-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. с. 96 — URL: <https://urait.ru/bcode/565693>  
<https://urait.ru/quiz/run-test/075A0CB2-59BD-432E-94CC-757476650A8E/7783D9C7-F78D-496A-A82B-F61B2469E036>

Тест №2 Веб-стандарты и их поддержка

Тузовский, А. Ф. Проектирование и разработка web-приложений : учебник для среднего профессионального образования / А. Ф. Тузовский. — Москва : Издательство Юрайт, 2025. — 219 с. — (Профессиональное образование). — ISBN 978-5-534-16767-2. — Текст : электронный //

Образовательная платформа Юрайт [сайт]. с. 96 — URL: <https://urait.ru/bcode/565693>

<https://urait.ru/quiz/run-test/E5166653-FAAD-4B51-AB35-A9D1DBA08B21/312DCC64-3082-4FE7-A7F5-66111B6ED670>

Тест№3. Язык гипертекстовой разметки.

Полуэктова, Н. Р. Разработка веб-приложений : учебник для среднего профессионального образования / Н. Р. Полуэктова. — 2-е изд. — Москва : Издательство Юрайт, 2025. — 204 с. — (Профессиональное образование). — ISBN 978-5-534-18644-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. с. 28 — URL: <https://urait.ru/bcode/567621>

<https://urait.ru/quiz/run-test/B5FC3131-F801-4D11-AC31-165A620883A8/2ABB69BE-728D-427A-9E3D-F50F5AB0A1C8>

Тест№4. Каскадные таблицы стилей CSS

Полуэктова, Н. Р. Разработка веб-приложений : учебник для среднего профессионального образования / Н. Р. Полуэктова. — 2-е изд. — Москва : Издательство Юрайт, 2025. — 204 с. — (Профессиональное образование). — ISBN 978-5-534-18644-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. с. 28 — URL: <https://urait.ru/bcode/567621>

<https://urait.ru/quiz/run-test/76A17405-CBB1-4F8B-BAF8-22091F7F9CC6/ADE66139-1EDA-4364-97B4-22B565317E82>

Тест№5. Технологии адаптивной верстки сайтов.

Полуэктова, Н. Р. Разработка веб-приложений : учебник для среднего профессионального образования / Н. Р. Полуэктова. — 2-е изд. — Москва : Издательство Юрайт, 2025. — 204 с. — (Профессиональное образование). — ISBN 978-5-534-18644-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. с. 28 — URL: <https://urait.ru/bcode/567621>

<https://urait.ru/quiz/run-test/C6451906-FC6C-4C3C-8F28-AACDE494DAD2/C84B4DF4-C3AE-4C79-A854-2B9739BA725B>

МДК.03.03. Разработка клиентской части информационных ресурсов

### Тема 3.1. Разработка клиентской части информационных ресурсов

Изучаемые понятия:

- Структуры данных JavaScript
- Замыкания, область видимости. Методы объектов и контекст вызова
- Документ и объекты страницы. Работа с событиями
- ООП в функциональном и прототипном стиле
- Фреймворки и библиотеки.

Тест №1. Структуры данных JavaScript

Полуэктова, Н. Р. Разработка веб-приложений : учебник для среднего профессионального образования / Н. Р. Полуэктова. — 2-е изд. — Москва : Издательство Юрайт, 2025. — 204 с. — (Профессиональное образование). — ISBN 978-5-534-18644-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. с. 28 — URL: <https://urait.ru/bcode/567621>  
<https://urait.ru/quiz/run-test/F6F160C5-A870-43EF-93CE-6AA69CF329C9/EE51918E-E4AB-47B4-B3B0-C7702E552F62>

### 3.2 Критерии оценки результатов выполнения тестовых заданий

В качестве критерия освоения материала профессионального модуля берётся выполнение 50% заданий от их общего числа в тест-задании.

Для интерпретации результатов тестирования используется переводная шкала, с помощью которой результаты теста преобразуют в оценку знаний студентов по пятибалльной системе. Переводная шкала представлена в Таблице 1.

Таблица 1 – Переводная шкала для интерпретации результатов выполнения тест-задания

<b>% ошибочных ответов</b>	<b>Интерпретация результатов тестирования по 5-ой системе</b>
до 10%	5
от 15% до 30%	4
от 35% до 50%	3
более 50%	2

### **3.3 Письменная форма контроля.**

#### **Примеры практических работ.**

#### **МДКн 03.01. Проектирование и дизайн интерфейсов**

#### **Практическое занятие №1. «Создание оригинал-макетов, элементов дизайна сайта»**

Цель: получить представление о технологии создания сайтов с применением CMS-систем.

Инструментарий:

- 1) любой современный браузер;
- 2) пакет разработчика Open Server.

Задания

1) Создать сайт, реализующий примерно те же функции, что и сайт, созданный в предыдущих практических работах, с использованием возможностей CMS WordPress.

2) Создать собственный сайт с использованием той же технологии.

Выполнение задания 1

Шаг 1. Создание базы данных сайта. WordPress — это CMS система, которая хранит все данные сайта в базе данных `mysql`. Поэтому нам необходимо создать пустую базу данных, которая будет подключена к будущему сайту. Запустим локальный сервер `OpenServer`, после его запуска выберем в меню Дополнительно пункт `PHPMyAdmin`. В браузере откроется программа для создания базы данных.

При входе система запросит логин и пароль, введем логин — `root`, а поле «Пароль» оставим пустым. Базу данных называем, например, «`myDB`», устанавливаем кодировку `utf-8 general-c` и нажимаем кнопку «Создать». Появится окно «`MyDB` была создана». При этом нужно помнить, что если не указывать ничего иного, то у этой БД имеется пользователь с максимальными правами под именем `root`, у которого по умолчанию нет пароля.

## Шаг 2. Установка WordPress.

Скачиваем CMS-систему со страницы [https://ru.wordpress.org/latest-ru\\_RU.zip](https://ru.wordpress.org/latest-ru_RU.zip). Распаковываем содержимое этого архива в папку, которая соответствует имени вашего сайта (например «new»): `..\OSPanel\domains\new`.

Указываем в адресе браузера путь: `http://localhost/new/`. Должно появиться окно начала настройки WordPress. Выбираем русский язык. Указываем следующие настройки: Далее запускаем установку, снова подтверждаем выбор языка и заполняем следующее окно так: Пароль необходимо ввести и запомнить, он понадобится для входа в административную панель.

Теперь мы можем работать в двух режимах — в режиме администрирования сайта и в режиме его просмотра. Для перехода к администрированию требуется вводить в адресной строке: `new/wp-admin`. Для просмотра сайта нужно ввести адрес `new/`. При этом результат будет таким: Для перехода между этими режимами также могут использоваться две кнопки в верхнем меню программы.

Итак, сайт уже оформлен в соответствии с некоторой темой оформления, содержит одну запись «Привет, мир!» и имеет некоторые элементы управления. Далее предлагается следующий план создания сайта:

- выбрать и настроить тему оформления для сайта;
- создать несколько статических страниц;
- создать несколько записей, которые будут размещены на главной странице в виде новостей;
- настроить меню сайта;
- настроить дополнительные возможности при помощи виджетов;"

Таким же образом можно создать и другие страницы. Список всех страниц можно просмотреть, отредактировать настройки и содержимое каждой из них через меню Страницы «Все страницы».

## Шаг 5. Создание записей.

Кроме статических страниц большинство современных сайтов содержат новостные страницы, которые обновляются автоматически, по мере добавления в базу данных новостей или записей.

Перед началом ввода записей нужно продумать, нужно ли вам, чтобы пользователи могли их комментировать, в каком виде они будут отображаться и т. д. Эти и другие настройки можно указать через пункты консоли Настройки → Написание/Чтение/Обсуждение.

Для создания записей выбираем в консоли Записи → Добавить новую и заполняем нужные поля: Таким образом, по мере добавления новостей последние будут отображаться на специальной странице новостей. Для этого нужно создать специальную страницу, в макете которой указать тип Блог.

#### Шаг 6. Настройка меню сайта.

На сайте может быть одно или несколько меню, это зависит от выбранной темы. В данном случае мы настроим главное.

Внешний вид → Виджеты перетащить, например, виджет «Календарь» в соответствующий сайдбар. Вообще, все активные области, которые допускает эта тема, можно увидеть на странице виджетов.

#### Шаг 7. Авторизация и регистрация.

Для реализации функций регистрации и авторизации посетителей на сайте в WordPress имеются встроенные средства. Их настройка может быть выполнена через меню Настройки. После настройки возможностей оставлять отзывы пользователь может зарегистрироваться или авторизоваться и оставлять комментарии на странице записей.

#### Шаг 8. Настройка плагинов.

Плагин, в отличие от виджета, — это внешняя подпрограмма, которая выполняет дополнительные функции, не предусмотренные в базовом наборе WordPress. В стандартной поставке WordPress имеется несколько установленных плагинов, например Akismet Anti-Spam, который позволяет защитить ваш сайт от спама. Для реализации дополнительных функций можно скачать плагины сторонних разработчиков.

## МДКн 03.02. Верстка страниц

### Практическое занятие № 1. «Применение тегов HTML при создании web-страниц»

Цель: Изучить основные возможности языка HTML для создания и связывания web-документов.

Инструментарий:

1) любой редактор программного кода, например, notepad ++, который можно бесплатно скачать с сайта разработчика по адресу <https://notepad-plus-plus.org/downloads/>;

2) любой современный браузер.

Задания 1. Выполнить все шаги практической работы, в результате которых должен получиться трехстраничный сайт-визитка туристической компании.

2. По аналогии реализовать свой сайт, используя только возможности языка HTML, на любую интересующую вас тему. Меню вашего сайта должно быть вертикальным. Необходимо задействовать все известные вам типы тегов: заголовки, абзацы, списки, изображения, таблицы, формы.

Выполнение задания 1

Шаг 1. Создание главной страницы сайта. На этом шаге мы показываем реализацию в среде Notepad ++. Следующие шаги описываются только кодами страниц, которые вы также можете вводить и отлаживать в Notepad ++. Итак, в среде Notepad ++ создадим новый документ и сохраним его в отдельной папке под именем index.html. Введем следующий текст. Старайтесь соблюдать структуру документа — отступы с использованием пробелов или табуляции, которые позволяют описать иерархию тегов.

```
< !-Начало HTML документа ->
< !DOCTYPE html >
< html >
< !-Начало заголовка документа ->
< head >"
```

```
< !-Служебная информация о кодировке документа -> < meta
http-equiv = "Content-Type"
content = "text/html; charset = utf-8" > < !-Информация, которая
будет отображаться на вкладке браузера ->
< title >
Сайт туристической компании
< /title > < !-Конец заголовка документа -> < /head > < !-Начало
тела документа -> < body > < !-Начало блока, содержащего всю страницу ->
< div id = "content" >
< !-Начало блока заголовка страницы ->
< header >
< !-Заголовок страницы ->
< h1 >
ПУТЕШЕСТВУЙ С НАМИ!
< /h1 >
< !-Конец блока заголовка страницы ->
< /header >
< !-Начало блока меню страницы ->
< nav >
< !-Гиперссылка на главную страницу сайта ->
< a href = "index.html" >
Главная страница
< /a >
< !-Гиперссылка на страницу с информацией о турах ->
< a href = "turs.html" >
Наши туры
< /a >
< !-Гиперссылка на страницу с контактной информацией ->
< a href = "contacts.html" >
О нас
```

```
</a >
< !-Конец блока меню ->
</nav >
< !-Начало блока с основным содержимым страницы ->
< div id = "main" >
< !-Начало нового абзаца ->
< p >
```

Концепция отдыха с нашей компанией это отдых в новом формате, где учтены интересы всех поколений.

```
</p >
```

```
< p >
```

Наши основные направления:

```
< !-Начало блока с перечнем пунктов ->
```

```
< ul >
```

```
< !-Блоки перечня пунктов ->"
```

```
" < li > Крым </li > < li > Кавказ </li > < li > Алтай </li > </ul > <
/p > < p >
```

```
Акции и скидки </p > < !-Конец блока с основным содержимым -
> </div > < !-Начало блока " подвала " страницы -> < footer >
```

```
Это сайт, предназначенный для обучения < !-Конец блока "
подвала " страницы -> </footer > < !-Конец блока с полным содержимым
страницы -> </div > < !-Конец тела документа -> </body > < !-Конец
документа -> </html > Если открыть этот файл в любом из браузеров, должен
получиться примерно такой результат:
```

Шаг 2. Описание структуры страницы.

Попытаемся отформатировать страницу так, чтобы ее блоки занимали определенные области на экране за счет применения стилей к каждому из блоков.

В область < head > ... </head > необходимо добавить следующий код:

```
< !-Начало описания стилей --! >
```

```

< style >
/*-Описание стиля всей страницы --*/
div#content {
    /*-Ширина 80 % от всего экрана --*/ width: 80 %; /*-Рамка
отсутствует --*/ border: none; /*-Выравнивание по центру занимаемой
области --*/ margin: auto; } /*-Описание стиля заголовка и подвала --! >
header, footer {
    /*-Отступ текста от края блока 1em текущий размер шрифта, для
увеличения или уменьшения можно брать любые пропорции от текущего :
2em, 0.5em --*/
    padding: 1em;
    /*-Цвет шрифта-белый --*/
    color: white;
    /*-Цвет фона --*/
    background-color: #007196;
    /*-Отменяет обтекание с левого края элемента, все другие
элементы на этой стороне будут располагаться под текущим элементом --*/
    clear: left;
    /*-Выравнивание текста по центру --*/
    text-align: center;
}
/*-Описание стиля блока меню --*/
nav {
    /*-Высота блока в пикселях --*/
    height: 40px;
    /*-Ширина и цвет линии обрамления --*/
    border: solid #333;
    padding: 1em;
    text-align: center;
}

```

```

/*-Описание стиля ссылки в меню --! >
nav a {
/*-Для ссылок используется оформление по умолчанию --*/
text-decoration: none;
padding: 1em;
}
/*-Описание стиля основного блока страницы --*/
#main {
padding: 1em;
} </style >"

```

Шаг 3. Добавление оформления к элементам таблицы. В папке, содержащей файл index.html, создадим папку img для хранения картинок, которые будут использоваться на сайте. Скопируем в нее картинку по тематике сайта, и пусть, например, она имеет имя logo.jpg. Поместим картинку logo.jpg слева от заголовка сайта. Для этого в блоке < header > ... < /header > нужно создать два блока — один для вывода логотипа, другой — для названия. Соответствующие изменения нужно внести в описание стилей.

Изменения в структуре:

```

< header >
< !-Начало блока для логотипа ->
< div id = "logo" >
< !--Вывод логотипа->
< img src = "img/logo.jpg" >
< !-Конец блока для логотипа ->
< /div >
< !-Начало блока для названия ->
< div id = "name" >
< h1 > ПУТЕШЕСТВУЙ С НАМИ! < /h1 >
< !-Конец блока для названия ->
< /div >

```

< /header >

Добавление к стилям:

/\*-для тега img, который размещен в блоке с идентификатором logo : ширина200 пикселей, высота150 пикселей, прижать к левой части родительского блока и заставить следующий блок обтекать его справа --\*/

< !-Первая строка таблицы -> < tr >

< !-Первый столбец первой строки будет далее делиться на два ->

< td rowspan = 2 > Туры в Крым < /td >

< !-Второй столбец первой строки ->

< td > Отель "Волна" < /td >

< /tr >

< !-Вторая строка ->

< tr >

< !-Второй столбец второй строки,  
первый столбец " растянулся " из первой строки ->

< td > Отель "Море" < /td >

< /tr >

< !-Третья строка ->

< tr >

< td rowspan = 2 > Туры на Кавказ < /td >

< td > Курортный отдых < /td >

< tr >

< td > Горный туризм < /td >

< /tr >

< tr >

< td rowspan = 2 > Туры на Алтай < /td >

< td > Зеленый туризм < /td >

< tr >

< td > Конные туры < /td >

```

</tr >
</table > </div > Добавим стили для таблицы: /*-Для всей
таблицы : --*/ table#tours {
/*-Ширина 80 % от ширины родительского блока --*/
width:80 %;
/*-Выравнивание по центру --*/
margin:auto; } /*-Для отдельной ячейки таблицы : --*/ table#tours
td{/*-Текст выровнен по левому краю --*/
text-align: left;
/*-Сплошная красная рамка толщиной 2 пикселя --*/
border: 2px solid red; }"

```

Шаг 6. Создание форм. На примере страницы contacts.html продемонстрируем использование форм. Пусть на третьей странице сайта необходимо создать форму для обращения пользователя к администратору сайта. Для более аккуратного вывода форму можно также разместить в виде таблицы. Для этого в блок main третьей страницы добавим следующий код:

```

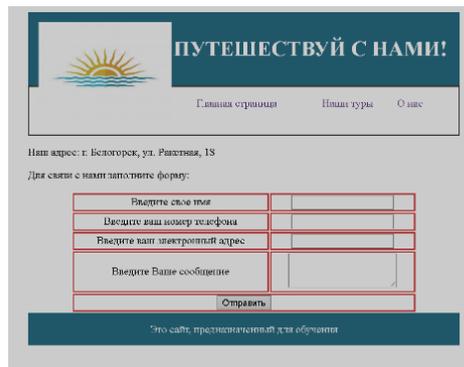
< p >
Наш адрес: г. Белогорск, ул. Ракетная, 18.
</p >
< p >
Для связи с нами заполните форму:
</p >
< !-Начало формы, программа, которая будет ее обрабатывать
пока неизвестна ->
< form method = "get" action = "#" >
< !-Начало таблицы ->
< table id = "form" >
< tr >
< td > Введите свое имя </td >
< !-Простое текстовое поле для ввода имени ->

```

```

< td >< input name = "name" >< /td >
< /tr >
< tr >
< td > Введите ваш номер телефона < /td >
< !-Простое текстовое поле для ввода телефона ->
< td >< input name = "tel" >< /td >
< /tr >
< tr >"

```



### МДКн 03.03. Разработка клиентской части информационных ресурсов

#### Практическое занятие №1. «Создание клиентских сценариев средствами JavaScript»

Цель: Ознакомиться с возможностями языка JavaScript для клиентской обработки web-страниц.

Инструментарий:

- 1) Редактор программного кода, например, notepad ++ ;
- 2) любой современный браузер.

Задания

1. Добавить к сайту, созданному в предыдущей практической работе расчет и вывод стоимости тура.
2. Придумать и выполнить аналогичные (по технологии) действия, имеющие смысл в контексте собственного приложения.

Выполнение задания 1.

Шаг 1. Создание кнопок для вызова скрипта. Предположим, что у вас имеется такая страница сайта, имеющая имя tour.html: А при изменении размера экрана страница будет выглядеть так: Код этой страницы приведен ниже (только область main).

```
< main class = "flex" >
    < div class = "row" >
    < div class = "col" >
        Наши туры
    < /div >
    < /div >
    < div class = "container-fluid" >
    < div class = "row" >
    < div class = "col" >
    < div class = "tour" >
    < img src = "img/crimea.jpg" class = "tour_img" >
    < p > Это прекрасная возможность отдохнуть всей семьей < /p >
    < !-Код компонента кнопки вызова модального окна bootstrap со
страницы https :// getbootstrap.com/docs/4.4/ components/modal/. Значение data-
target задайте свое! ->
    < button type = "button" class = "btn btn-primary"
data-toggle = "modal" data-target = "#myModal" >
        Рассчитать стоимость
    < /button >
    < /div >
    < /div >
    < /div >
    < /div >
    < div class = "row" >
    < div class = "col" >
    < div class = "tour" >
    < img src = "img/kavkaz.jpg" class = "tour_img" >
```

```

< p > Море, солнце и горы! < /p >
< button type = "button" class = "btn btn-primary"
data-toggle = "modal" data-target = "#myModal" >"
"    Рассчитать стоимость < /button > < /div > < /div > < /div > < div class =
"row" >
    < div class = "col" >
    < div class = "tour" >
    < img src = "img/altay.jpg" class = "tour_img" >
    < p > Незабываемые впечатления! < /p >
    < button type = "button" class = "btn btn-primary"
data-toggle = "modal" data-target = "#myModal" >
    Рассчитать стоимость
    < /button >
    < /div >
    < /div > < /div > < /div > < /div > < /main >

```

То есть на странице есть три области, каждая из которых занимает целиком строку, в которой расположены картинка, некоторый текст и кнопка для расчета стоимости соответствующего тура. Нажатие на каждую кнопку должно вызывать открытие модального окна с идентификатором myModal.

Далее необходимо добавить стили к таблице стилей:

```

.tour{
border: solid #007196;
margin: auto;
height: 100 %;
/* Установили возможность использования технологии flex */
display: flex;
/* Указали направление расположения внутренних элементов по строке
( row ), и одновременно задали возможность для них переноситься на другую
строку при уменьшении экрана */
flex-flow: row wrap;

```

```

    }
    .tour img {
    width: 250px;
    height: 170px;
    padding: 5px;
    }
    .tour p .tour img {
    margin: 10px;
    }"
" /* Чтобы прижать кнопку к правому краю, если элемент, который
нужно прижать к правому краю, не последний в контейнере */ .btn {
margin-left: auto;
order:999;
background-color:#66CDAA;
box-shadow: 20px 20px -20px #000 inset, -20px 20px -20px #000 inset;
color:white; }

```

Шаг 2. Создание модального окна. Организуем вывод модального окна, которое открывается после нажатия на кнопку «Рассчитать стоимость». Это окно создадим с помощью Bootstrap. Скопируем код модального окна (<https://getbootstrap.com/docs/5.3/components/modal/>) в верхнюю часть кода страницы.

```

< body class = "body-top" >
...
< !-Modal ->
< !-Укажите id этого модульного окна таким же, как был параметр
data-target в кнопке его вызова на предыдущем шаге ->
< div class = "modal fade" id = " myModal " tabindex = "-1"
role = "dialog"
aria-labelledby = "exampleModalScrollableTitle"
aria-hidden = "true" >

```

```

< div class = "modal-dialog modal-dialog-scrollable" role =
"document" >
< div class = "modal-content" >
< div class = "modal-header" >
< h5 class = "modal-title"
id = "exampleModalScrollableTitle" > Modal title < /h5 >
< button type = "button" class = "close"
data-dismiss = "modal" aria-label = "Close" >
< span aria-hidden = "true" > &times; < /span >
< /button >
< /div >
< div class = "modal-body" >
...
< /div >
< div class = "modal-footer" >
< button type = "button" class = "btn btn-secondary" dataddismiss = "modal"
> Close < /button >
< button type = "button" class = "btn btn-primary" > Save
changes < /button >
< /div >
< /div >"

```

В результате, при нажатии на любую из кнопок «Рассчитать стоимость» должно получиться: То есть в Bootstrap уже созданы все необходимые CSS и JSкрипты, организующие вывод этого пустого модального окна.

Шаг 3. Создание собственной формы в модальном окне.

Создадим в модальном окне форму для запроса у пользователя необходимых нам данных для расчета стоимости тура. Предположим, что пользователь должен выбрать название тура, дату заезда, количество участников.

```

< !-Начало описания модального окна bootstrap ->
< div class = "modal-body" style = "max-width: 600px" >

```

< !-Начало формы ->

< form method = "get" action = "#" id = "orderForm" >

< !-Далее в блоках класса tour ( чтобы к ним были применены

те же стили, что и на основной странице сайта ) описываются поля формы для ввода данных.

В том числе поле для выбора страны ( идентификатор, который будет использоваться для дальнейшей обработки inp1 ); поле для ввода даты ( типdate, идентификаторinp2 ); поле для выбора количества участников тура ( идентификатор inp3 ) и поле для ввода email пользователя, идентификатор inp4 ->

< div class = "tour" >

< label class = "label" for = "name" >

Выберите тур:

< /label >

< !-id формы ( orderForm ) будет использоваться в скрипте обработки формы, рассматриваемом ниже ->

< select id = " inp1 " >

< option value = "Крым" selected >

Крым

< /option >

< option value = "Кавказ" >

Кавказ

< /option >

" < option value = "Алтай" >

Алтай < /option > < /select > < div > < div class = "tour" >

< label class = "label" for = "data" >

Выберите дату начала:

< /label >

< input type = "date" id = " inp2 " > < /div > < div class = "tour" >

< label class = "label" for = "number" >

Выберите количество участников:

```
</label >  
< select id = " inp3 " >  
< option value = "1" > 1 </option >  
< option value = "2" selected > 2 </option >  
< option value = "3" > 3 </option >  
< option value = "3" > 4 </option >  
</select > </div > < div class = "tour" >  
< label class = "label" for = "email" >
```

Ваш E-mail:

```
</label >  
< input type = "email" id = " inp4 " class = "input-xlarge" style = "width:  
350px;" required = "required" >  
</div >
```

<!--ОБРАТИТЕ ВНИМАНИЕ! Для упрощения обработки, перенесите кнопки управления Закреть и Отправить из футера модального окна в форму. Футер модального окна при этом можно удалить ->

```
< button type = "button" class = "btn btn-primary" data-dismiss = "modal" >  
Закреть </button >
```

```
< button type = " submit " class = "btn btn-primary" id = "submit" >  
Отправить </button > </form >
```

В результате при нажатии любой из кнопок вы получите модальное окно «Расчет стоимости тура».

Шаг 4. Создание скрипта для обработки формы, заполненной пользователем в модальном окне.

Напишем скрипт на JavaScript с применением возможностей библиотеки JQuery, который позволит обработать данные, введенные пользователем в модальное окно и показать результат.

В нижней части страницы tours.html, после строк подключения библиотек нужно ввести следующий код:

```
< script >
```

< !-jQuery позволяет обращаться прямо к идентификатору формы.

По событию связанному со срабатыванием кнопки submit на форме, выполняется следующая функция ->

```
$('#orderForm').on('submit', function ()
```

```
{
```

< !-получаем название тура ->

```
let tour = $("#inp1").val();
```

< !-получаем дату ->

```
let date = new Date($("#inp2").val());
```

< !-получаем месяц из даты ->

```
let month = date.getMonth();
```

< !--получаем количество участников тура и преобразуем его в число ->

```
let kol = $("#inp3").val();
```

```
kol = Number(kol);
```

< !-получаем email ->

```
let email = $("#inp4").val();
```

< !-Выполняем некоторый условный расчет, например, если тур в Крым, то в определенные месяцы цена выше, а в остальных ниже ->

```
let stoim = 0;
```

```
if (tour == 'Крым')
```

```
{
```

```
if((month == 5)||(month == 6)||(month == 7)||(month == 8))
```

```
{ stoim = kol*500; }
```

```
else { stoim = kol*300; }
```

```
}
```

```
if (tour == 'Кавказ')
```

```
{
```

```

if((month == 12)||(month == 1)||(month == 2)||(month ==
5)||(month == 6)||(month == 7)||(month == 8))
{stoim = kol*300;}
else {stoim = kol*250;}
}
if (tour == 'Алтай')
{
{stoim = kol*1000;}
}
< !-Выводим полученный результат ->"
alert ('Примерная стоимость вашего тура на ${kol} человек составит
${stoim} у.е. Мы свяжемся с ВАМИ!!');
}
); < /script > В результате получим следующее:"

```



### 3.4. Критерии оценки результатов выполнения письменных заданий

Оценка «5» (отлично) – краткая, ясная и четкая описание решения задания, верно, на высоком профессиональном уровне, с правильными пояснениями. Обоснованность ответа. Необходимо мотивировать выбранный курс действий, приводящих к разрешению ситуации, и объяснить причины и рациональность его выбора.

Оценка «4» (хорошо) – задание выполнено в целом верно, но с небольшими неточностями, имеются неточности в пояснении.

Оценка «3» (удовлетворительно) – задание выполнено верно, но не доведен до конца, либо в нем имеются ошибки, которые, однако, не приводят к принципиально неверному решению.

Оценка «2» (неудовлетворительно) – задание не выполнено или выполнено неверно

## **4. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ**

Промежуточная аттестация по профессиональному модулю ПМн.03 «Проектирование и разработка веб-приложения на стороне клиента» состоит из следующих компонентов:

### **IV семестр:**

комплексный дифференцированный зачет – МДКн 03.01;

### **V семестр:**

комплексный дифференцированный зачет - МДКн 03.02;

защита курсового проекта - МДКн 03.02;

### **VI семестр:**

комплексный дифференцированный зачет - МДКн 03.03;

защита курсового проекта - МДКн 03.03.

### **4.1 Критерии оценивания знаний, обучающихся на дифференцированном зачете**

До даты проведения дифференцированного зачета необходимо пройти оценивание всех работ, выполненных в течение семестра: теоретическое обучение, тестирование, практические занятия.

К установленной дате дифференцированного зачета обучающийся должен выполнить 100% работ, предусмотренных рабочей программой междисциплинарного курса.

Форма проведения дифференцированного зачета – выполнение заданий двух типов: ответ на теоретические вопросы и выполнение практического задания»

Дифференцированный зачет оценивается по пятибалльной шкале путем вычисления среднего балла за выполнение заданий.

## МДКн 03.01. Проектирование и дизайн интерфейсов

### Вопросы к дифференцированному зачету:

1. Что такое семантическая разметка и зачем она нужна в HTML?
2. Какие этапы включает процесс веб-разработки?
3. Какие методы помогут сделать HTML-код более читаемым и поддерживаемым?
4. Как используются Media Queries для создания адаптивных веб-страниц?
5. Какие преимущества дает использование фреймворков Bootstrap и Foundation для адаптивной верстки?
6. Какое значение имеют UI и UX для веб-дизайна?
7. Какие принципы следует учитывать при создании удобного пользовательского интерфейса?
8. Какие этапы включает процесс проектирования интерфейса?
9. Что такое CSS-препроцессоры и зачем они нужны?
10. Какие основные отличия между SASS и LESS?
11. Как Git помогает в управлении версиями кода при разработке веб-проектов?
12. Какие метрики используются для оценки качества пользовательского интерфейса?
13. Как провести юзабилити-тестирование и почему оно важно?
14. Какие тренды в веб-дизайне популярны в настоящее время?

### Пример задания к дифференцированному зачету

1. Создайте веб-страницу, используя HTML, которая будет включать следующие элементы: Заголовок страницы (например, <title>) с названием "Мой первый веб-сайт".
  - Основной заголовок (<h1>) с текстом "Добро пожаловать на мой веб-сайт".
  - Параграф (<p>) с кратким описанием о себе или вымышленного проекта.
  - Изображение (<img>) с любым свободно доступным изображением

(можно использовать изображение с локального компьютера).

- Несколько гиперссылок (<a>) на страницы с социальными сетями или другие сайты.

Используя CSS, оформите эту страницу:

- Добавьте цвет фона для всего документа.
- Измените шрифт основного текста и заголовков.
- Сделайте изображение круглым и добавьте ему тень.
- Добавьте стиль к гиперссылкам: измените цвет текста и добавьте эффект при наведении (hover).

Сделайте страницу адаптивной:

- Используйте @media запросы, чтобы изменить расположение элементов на странице в зависимости от размера экрана (например, изменение ширины контейнера или размеров шрифтов).

2. Создайте макет страницы, используя Flexbox: 1.1. Создайте основную структуру страницы с header, main и footer.

- В header добавьте логотип (текст или изображение) и навигационное меню с тремя-четырьмя пунктами (например, "Главная", "О нас", "Контакты").
- В main добавьте две колонки: одна для основного контента (например, статья или текст), другая — для боковой панели (sidebar) с дополнительной информацией или ссылками.
- В footer разместите контактную информацию и ссылки на социальные сети.

Примените основные принципы UI/UX-дизайна:

- Используйте правила контраста и размер шрифтов для улучшения читабельности текста.
- Добавьте пространство между элементами (используя margin и padding) для улучшения визуальной структуры.
- Примените цветовую схему с учетом принципов доступности (Accessibility), чтобы все элементы были различимы для пользователей

с нарушениями зрения.

3. Проверьте, как выглядит макет на мобильных устройствах:

Используйте инструменты разработчика в браузере (например, DevTools в Google Chrome) для проверки, как макет отображается на разных устройствах (мобильных, планшетах, десктопах).

### **МДКн 03.02. Верстка страниц**

#### **Вопросы к дифференцированному зачету**

1. Понятие веб-браузера, основные возможности браузеров для веб-разработчика их преимущества и недостатки
2. Строгий режим, использование, особенности
3. Тег `<script />`, назначение, использование, атрибуты `defer` и `async`
4. Статический анализатор кода ESLint, назначение, подключение к проекту
5. Форматтер кода Prettier, назначение, подключение к проекту
6. Типы данных в JavaScript, примитивы и объекты
7. Операторы ветвления в JavaScript, тернарный оператор
8. Операторы циклов в JavaScript
9. Функции в JavaScript, передача параметров
10. Понятия кодировки символов и таблицы символов
11. Кодировки UTF-8 и UTF-16
12. Методы `RegExp` и `String` для работы с регулярными выражениями
13. Регулярные выражения, основные классы символов
14. Тип данных `object` в JavaScript, использование
15. Конструкторы объектов в JavaScript, оператор `new`
16. Массивы в JavaScript, синтаксис
17. Методы для обработки массивов (`.map`, `.filter`, `.reduce` и другие)
18. Механизм замыканий, назначение и использование
19. Понятие контекста функций в JavaScript
20. Контекст в функциях-конструкторах объектов
21. Понятие прототипа объекта в JavaScript
22. Цепочка прототипов объектов, `null prototype`

23. Методы Object.prototype
24. ООП в JavaScript, синтаксис
25. Оператор try .. catch .. finally, назначение и синтаксис
26. Выбрасывание исключительных ситуаций, оператор throw
27. Объект ошибки, основные поля
28. Основные события мыши и клавиатуры на веб-страницах
29. События элементов управления (кнопок, форм и так далее)
30. Способы назначения обработчиков событий
31. Объект события, основные методы и поля
32. Callback-функции в JavaScript
33. Промисы в JavaScript, создание цепочки промисов
34. Асинхронные функции в JavaScript, операторы async/await
35. Цикл событий JavaScript
36. Понятие блокирующей и неблокирующей функции
37. Функции-генераторы в JavaScript, назначение
38. Разделение кода на модули, синтаксис, операторы import и export
39. Методы для поиска элементов в браузере

### **Задания к дифференцированному зачету**

1. Настройте webpack для использования плагинов и лоадеров.
2. Создайте и используйте TypeScript для создания и использования конструкторов и деструкторов классов.
3. Используйте TypeScript для создания асинхронных функций и работы с Promise.
4. Сохраните данные в куки и затем прочитайте их.
5. Создайте конфигурацию webpack для минификации и оптимизации проекта.
6. Используйте TypeScript для создания и использования наследования классов
7. Создайте и используйте TypeScript для создания и использования абстрактных классов и методов.

8. Используйте Stream API для чтения и записи данных в браузере.

### **МДКн 03.03. Разработка клиентской части информационных ресурсов**

#### **Вопросы к дифференцированному зачету**

1. Объект document в браузере
2. Методы изменения дерева объектов в JavaScript
3. Порядок обработки событий в браузере, захват и всплытие
4. Механизм делегирования событий
5. Обработчики событий по умолчанию
6. Объект window, основные поля и методы
7. Объект location, предназначение, основные поля и методы
8. Объект history, предназначение, основные поля и методы
9. Локальное хранилище данных, основные методы
10. localStorage и sessionStorage, отличия
11. JSON, особенности формата
12. Объект FormData, отправка формы на сервер средствами JavaScript
13. Работа с файлами в JavaScript, объекты Blob, File и FileReader
14. Понятие JavaScript проекта, процесс создания
15. Файл package.json и package-lock.json, назначения, содержимое
16. Понятие npm модулей, команды для установки / публикации модулей
17. Сборщик модулей webpack, предназначение
18. Синтаксис и основные поля файла webpack.config.js
19. Синтаксис языка TypeScript
20. Интерфейсы объектов в TypeScript
21. Сложные типы в TypeScript
22. Введение в библиотеку ReactJS;
23. Хуки в reactJS (useState, useEffect, useContext и другие);
24. Библиотеки для создания и обработки форм (formik, yup);
25. Библиотека для обработки форм react-hook-forms;

26. Интеграция React приложений с сервером, функция fetch;
27. Маршрутизация в React приложениях, библиотека react-router-dom;
28. Библиотека redux, основные компоненты (store, actions, actionsCreators, reducers);
29. Библиотека redux-thunk, обработка асинхронных действий в redux;
30. Обработка асинхронных действий redux с помощью библиотеки redux-saga;
31. Эффекты fork и race в redux-saga, параллельная обработка действий в redux-saga;
32. Библиотека готовых компонентов react-bootstrap;
33. Библиотека готовых компонентов material-ui;
34. Тестирование JS-приложений, библиотека Jest;
35. Тестирование React-приложений с помощью библиотеки Jest;
36. Архитектура React-приложений;
37. Создание анимации на React;
38. Создание общедоступных приложений на React;
39. Сборка и развертывание React-приложений, разделение приложения на части

### **Задания к дифференцированному зачету**

1. Создайте форму для отправки сообщения на сервер
2. Используйте хуки для работы с данными
3. Используйте контекст при работе с данными
4. Создайте сайт с помощью библиотеки react-bootstrap
5. Создайте сайт с помощью библиотеки material-ui
6. Создайте анимацию на React с помощью библиотек
7. Используя библиотеку redux-thunk, обработайте асинхронных действий в redux
8. Создайте общедоступное приложение на React

Ответы на теоретические вопросы и выполнение практического задания на дифференцированный зачет оценивается по пятибалльной шкале

следующим образом:

Оценка «Отлично» ставится, если студент строит ответ логично в соответствии с планом, показывает максимально глубокие знания профессиональных терминов, понятий, категорий, концепций и теорий. Устанавливает содержательные межпредметные связи. Развернуто аргументирует выдвигаемые положения, приводит убедительные примеры. На этом уровне освоения вида деятельности, творческое действие – самостоятельное конструирование способа деятельности, поиск новой информации. Формулирование оценочных суждений на основе имеющихся фактов и заданных критериев.

Оценка «Хорошо» ставится, если студент строит свой ответ в соответствии с планом. В ответе представлены различные подходы к проблеме, но их обоснование недостаточно полно. Устанавливает содержательные межпредметные связи. Развернуто аргументирует выдвигаемые положения, приводит необходимые примеры, однако показывает некоторую непоследовательность анализа. Выводы правильны. Речь грамотна, используется профессиональная лексика. На этом уровне освоения вида деятельности, воспроизведение, репродуктивное действие – самостоятельное воспроизведение и применение информации для выполнения данного действия. Студент способен по памяти воспроизводить ранее усвоенную информацию и применять усвоенные алгоритмы деятельности для решения типовых задач.

Оценка «Удовлетворительно» ставится, если ответ недостаточно логически выстроен, план ответа соблюдается непоследовательно. Студент обнаруживает слабость в развернутом раскрытии профессиональных понятий. Выдвигаемые положения декларируются, но недостаточно аргументированы. Ответ носит преимущественно теоретический. На этом уровне освоения вида деятельности, применение, продуктивное действие – поиск и использование информации для самостоятельного выполнения нового действия (знания, умения, навыка). Этот уровень предполагает

комбинирование студентом известных алгоритмов и приемов деятельности, применения навыков эвристического мышления

Оценка «Неудовлетворительно» ставится при условии недостаточного раскрытия профессиональных понятий, категорий, концепций, теорий. Студент проявляет стремление подменить научное обоснование проблем рассуждениями обыденно повседневного бытового характера. Ответ содержит ряд серьезных неточностей. Выводы поверхностны. Репродуктивная деятельность (узнавание объектов, свойств, процессов при повторном восприятии информации о них или действий с ними). На этом уровне студент не способен самостоятельно, без помощи извне, воспроизводить и применять полученную информацию.

## **4.2 Критерии оценивания знаний, обучающихся при защите результатов курсового проектирования**

Курсовое проектирование является завершающим этапом изучения междисциплинарных курсов МДКн 03.02 «Верстка страниц» и МДКн 03.03. «Разработка клиентской части информационных ресурсов», в ходе которых осуществляется закрепление полученных умений и знаний при решении задач, связанных с профессиональной деятельностью веб-разработчиков.

Самостоятельная разработка курсового проекта подготавливает обучающегося к успешному выполнению дипломного проекта и является важным этапом в развитии общих компетенций и формировании профессиональных компетенций веб-разработчика, установленных во ФГОС СПО специальности 09.02.09

Выполнение курсового проекта проводится с целью:

- систематизации и закрепления полученных теоретических знаний и практических умений;
- углубления теоретических знаний в соответствии с заданной темой;
- формирования умений применять теоретические знания при решении поставленных вопросов;
- формирования умений использовать справочную, нормативную и правовую документацию;
- развития творческой инициативы, самостоятельности, ответственности и организованности;
- развития навыков работы в коллективе.

Задачами курсового проекта являются:

- изучение особенностей конкретной предметной области, относящихся к теме курсового проекта;
- анализ возможных подходов и методов решения с обоснованием выбранного метода;

- выбор или разработка модели (математической, структурной, информационной и т.д.), необходимой для достижения цели;
- выбор эффективных алгоритмов с учётом их точности, устойчивости, сходимости и т.д.;
- выбор инструментальных средств разработки и его обоснование;
- разработка информационного ресурса по заданной тематике;
- анализ полученных результатов работы;
- разработка программной и эксплуатационной документации.

По содержанию курсовой проект носит практический или опытно-экспериментальный характер.

Структура курсового проекта:

- титульный лист;
- оглавление;
- техническое задание;
- введение;
- основную часть;
- список использованных источников;
- приложения.

Техническое задание разрабатывается в соответствии с ГОСТом ЕСПД. В зависимости от типа курсового проекта в техническом задании указываются требования к программе и её основные характеристики.

Во введении указываются актуальность и значимость темы, формулируются цель и задачи работы, область применения, перечень исходных данных и результатов работы программы. Объем не более 1 страницы.

Основная часть работы должна содержать следующие главы:

- Описание предметной области;
- Обоснование выбора инструментальных средств моделирования предметной области;
- Модель информационного ресурса - описание этапов разработки

модели предметной области;

- Оптимизация модели информационного ресурса и требования к её программному обеспечению;
- Обоснование выбора инструментальных средств проектирования программного модуля информационной системы;
- Проект программного кода;
- Набор тестов и сценариев для информационного ресурса;
- Рекомендаций по внедрению программного кода в информационный ресурс.

Главы должны иметь заголовки, отражающие их содержание. При этом заголовки глав не должны повторять название работы.

Список использованных источников включает в себя:

- нормативные правовые акты; ГОСТЫ ЕСПД
- литературу и материалы периодической печати;
- практические материалы
- сетевые ресурсы.

В список использованных источников включаются источники, изученные студентом в процессе подготовки работы в т. ч. те, на которые он ссылается.

В приложениях должны быть представлены тексты реализации программного модуля, графическое описание модели информационной системы и графические компоненты проекта информационной системы.

Законченные курсовые проекты в установленный срок сдаются обучающимися преподавателю-руководителю курсового проектирования, который проверяет качество выполненного проекта на соответствие поставленной задаче, а также определяет уровень теоретических знаний и практической подготовки обучающихся.

К защите допускаются обучающиеся, предоставившие в установленные сроки пояснительную записку к курсовому проекту,

программный модуль на диске, инструкцию пользователя в виде отдельного документа.

Обучаемый на защите должен быть готов:

- к краткому изложению основного содержания проекта;
- к собеседованию по отдельным, как правило, ключевым моментам проекта;
- к ответу на дополнительные и уточняющие содержание проекта, вопросы.

Если курсовой проект оценен неудовлетворительно, то после исправления ошибок представляется на повторное оценивание. При выявлении серьезных отклонений от предъявляемых требований к курсовому проекту обучающемуся предлагается устранить недостатки или разработать новую тему курсового проекта.

Окончательная оценка курсового проекта выставляется по итогам защиты. При получении неудовлетворительной оценки студент обязан повторно выполнить проект по новой теме или переработать прежний.

Курсовой проект должен быть защищен до сдачи дифференцированных зачётов междисциплинарным курсам МДКн 03.02 «Верстка страниц» и МДКн 03.03 «Разработка клиентской части информационных ресурсов» профессиональному модулю ПМн.03 «Разработка веб-приложения на стороне клиента».

Обучающийся, не защитивший в установленный срок курсовой проект к комплексному дифференцированному зачету не допускается.

Если обучающийся не смог представить проект к защите в установленный срок по уважительной причине (болезнь и т. п.), то в этом случае ему предоставляется дополнительное время для завершения проекта.

Основными показателями качества курсового проекта и уровня формирования общих компетенций являются:

- степень разработки темы;
- полнота охвата научной литературы;

- творческий подход к написанию курсового проекта;
- правильность и научная обоснованность выводов;
- аккуратность и правильное оформление курсового проекта.

Основными показателями уровня освоения профессиональных компетенций обучающегося являются:

- Правильность оформления документации на программный модуль в соответствии с требованиями ГОСТ.

- Правильность создания модели предметной области в соответствии используемой методологией;

- Правильность выбора инструментальных средств разработки в соответствии с требованиями к разработке;

- Правильность применение основных принципов технологии структурного и объектно-ориентированного программирования в соответствии с объектно-ориентированным подходом;

- Правильность и точность разработки кода информационного ресурса в соответствующей среде разработки веб-приложений;

- Правильность создания рекомендаций по внедрению информационного ресурса в соответствии с условиями его эксплуатации требованиями ГОСТ.

- Правильность применения основных принципов отладки и тестирования веб-приложений в соответствии с выбранной методикой;

- Точность использования инструментальных средств на этапе отладки веб-приложения;

- Правильность использования инструментальных средств для автоматизации оформления документации в соответствии со стандартами.

Оценивание результатов курсового проектирования производится по пятибалльной системе. Допустимые формы оценок: «5» (отлично), «4» (хорошо), «3» (удовлетворительно), «2» (неудовлетворительно):

оценка «5» (отлично). Пояснительная записка к курсовому проекту оформлена в соответствии с установленными требованиями. Комплектация курсового проекта представлена в полном объеме. В процессе защиты обучающийся демонстрирует всестороннее знание содержания курсового проекта. Исчерпывающе полно, последовательно излагает ответы на вопросы преподавателя по ключевым моментам проекта. В ответах логически стройно устанавливает связь теории с практикой;

оценка «4» (хорошо) Пояснительная записка к курсовому проекту оформлена с небольшими неточностями в соответствии с установленными требованиями. Комплектация курсового проекта представлена в полном объеме. В процессе защиты обучающийся демонстрирует всестороннее знание содержания курсового проекта. Ответы на дополнительные вопросы преподавателя по ключевым моментам проекта излагает со значительными упущениями. В ответах логически стройно устанавливает связь теории с практикой;

оценка «3» (удовлетворительно). Пояснительная записка к курсовому проекту оформлена не в соответствии с установленными требованиями. Комплектация курсового проекта представлена в не полном объеме. В процессе защиты обучающийся демонстрирует знание содержания курсового проекта в объеме, необходимом для дальнейшей учебы и работы. Ответы на дополнительные вопросы преподавателя по ключевым моментам проекта излагает не полно и слабо аргументировано;

оценка «2» (неудовлетворительно). Пояснительная записка к курсовому проекту не оформлена в соответствии с установленными требованиями. Комплектация курсового проекта не представлена в полном объеме. В процессе защиты обучающийся не обладает знанием содержания курсового проекта в объеме, необходимом для дальнейшей учебы и работы. Ответы на дополнительные вопросы преподавателя по ключевым моментам проекта излагает не верно.

### **Тематика курсовых проектов:**

Тематика курсовых проектов разрабатывается преподавателями цикловой методической комиссии информационных технологий, рассматривается на заседании ЦМК ИТ и утверждается директором колледжа НТИ НИЯУ МИФИ. Тема курсового проекта может быть предложена обучающимся, в соответствии с его интересами и возможностями раскрытия предлагаемой проблемы, при условии обоснования им ее целесообразности. Допускается выполнение курсового проекта по одной теме группой студентов.

Курсовой проект может стать составной частью – разделом, главой - дипломного проекта студента.

Общая тематика курсового проектирования направлена на разработку модулей программного обеспечения для интеграции их в информационные системы различной направленности.

Обучающийся должен создать функциональную или информационную модель предметной области, выполнить её оптимизацию, разработать спецификации внедряемого программного продукта, и осуществить его проектирование и реализацию.

Темы курсовых проектов представлены, как описание предметной области, подлежащей моделированию:

МДКн03.02 Верстка страниц

1. Разработка интернет-магазина "Продажа радиотоваров"
2. Разработка интернет-магазина "Элитные смартфоны"
3. Разработка интернет-магазина "Компьютерные игры"
4. Разработка интернет-магазина "Продажа книг по программированию"
5. Разработка интернет-магазина "Комплекующие и сборка ПК"
6. Разработка интернет-магазина "Детская одежда"
7. Разработка интернет-магазина "Тюнинг авто"
8. Разработка интернет-магазина "Стройматериалы с доставкой"
9. Разработка интернет-магазина "Спортивная одежда"

10. Разработка интернет-магазина "Канцелярские товары"

11. Разработка интернет-магазина "Продажа часов"

МДКн 03.03. Разработка клиентской части информационных ресурсов

1. Разработка интернет-магазина "Продажа радиотоваров",

2. Разработка интернет-магазина "Элитные смартфоны"

3. Разработка интернет-магазина "Компьютерные игры"

4. Разработка интернет-магазина "Продажа книг по программированию"

5. Разработка интернет-магазина "Комплектующие и сборка ПК"

6. Разработка интернет-магазина "Детская одежда"

7. Разработка интернет-магазина "Тюнинг авто"

8. Разработка интернет-магазина "Стройматериалы с доставкой"

9. Разработка интернет-магазина "Спортивная одежда"

10. Разработка интернет-магазина "Канцелярские товары"

11. Разработка интернет-магазина "Продажа часов"