

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

**федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский Нижегородский государственный университет
им. Н.И. Лобачевского»**

Институт информационных технологий, математики и механики

УТВЕРЖДЕНО

решением Ученого совета ННГУ

протокол № 10 от 02.12.2024 г.

Рабочая программа дисциплины

Основы фронтенд разработки

Уровень высшего образования

Бакалавриат

Направление подготовки / специальность

09.03.04 - Программная инженерия

Направленность образовательной программы

Разработка программно-информационных систем

Форма обучения

очная

г. Нижний Новгород

2025 год начала подготовки

1. Место дисциплины в структуре ОПОП

Дисциплина Б1.В.05 Основы фронтенд разработки относится к части, формируемой участниками образовательных отношений образовательной программы.

2. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями и индикаторами достижения компетенций)

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине (модулю), в соответствии с индикатором достижения компетенции		Наименование оценочного средства	
	Индикатор достижения компетенции (код, содержание индикатора)	Результаты обучения по дисциплине	Для текущего контроля успеваемости	Для промежуточной аттестации
ПК-12: Способен осуществлять концептуальное, функциональное и логическое проектирование систем и подсистем малого и среднего масштаба и сложности.	ПК-12.1: Знает методы концептуального, функционального и логического проектирования ПК-12.2: Знает типовые архитектурные стили и паттерны проектирования ПК-12.3: Умеет применять архитектурные стили и паттерны проектирования при решении типовых задач ПК-12.4: Владеет навыками разработки проекта программной системы с учетом возможностей и ограничений	ПК-12.1: Знает методы концептуального, функционального и логического проектирования ПК-12.2: Знает типовые архитектурные стили и паттерны проектирования ПК-12.3: Умеет применять архитектурные стили и паттерны проектирования при решении типовых задач ПК-12.4: Владеет навыками разработки проекта программной системы с учетом возможностей и ограничений	Практическое задание	Зачёт: Тест Проект

3. Структура и содержание дисциплины

3.1 Трудоемкость дисциплины

	очная
Общая трудоемкость, з.е.	2
Часов по учебному плану	72
в том числе	

аудиторные занятия (контактная работа):	
- занятия лекционного типа	16
- занятия семинарского типа (практические занятия / лабораторные работы)	16
- КСР	1
самостоятельная работа	39
Промежуточная аттестация	0 Зачёт

3.2. Содержание дисциплины

(структурированное по темам (разделам) с указанием отведенного на них количества академических часов и виды учебных занятий)

Наименование разделов и тем дисциплины	Всего (часы)	в том числе			
		Контактная работа (работа во взаимодействии с преподавателем), часы из них			Самостоятельная работа обучающегося, часы
		Занятия лекционного типа	Занятия семинарского типа (практические занятия/лабораторные работы), часы	Всего	
	0 Ф 0	0 Ф 0	0 Ф 0	0 Ф 0	0 Ф 0
1. Верстка HTML и CSS. Знакомство с JavaScript	5	1	1	2	3
2. Базовые CSS-свойства	5	1	1	2	3
3. JavaScript. Начало	5	1	1	2	3
4. Флексбокс-вёрстка	8	2	2	4	4
5. Расширенные возможности HTML и CSS	5	1	1	2	3
6. HTML: видео, iframe, embed, API	8	2	2	4	4
7. Анимации и трансформации	7	2	2	4	3
8. Разметка и вёрстка форм	5	1	1	2	3
9. Адаптивная вёрстка и работа с макетом	5	1	1	2	3
10. Работа с макетом	5	1	1	2	3
11. Grid Layout	5	1	1	2	3
12. Разработка интерфейса для разных устройств	8	2	2	4	4
Аттестация	0				
КСР	1				1
Итого	72	16	16	33	39

Содержание разделов и тем дисциплины

1. Верстка HTML и CSS. Знакомство с JavaScript

1.1. Интро в спринт

1.1.1. Что делают разработчики?

1.1.2. Базовые элементы. HTML

1.1.3. Мысли о стилях. CSS

1.1.4. Сценарное мастерство. JavaScript

- 1.1.5. Трюки и финты
- 1.2. Что такое HTML и CSS
 - 1.2.1. Нужен плейлист
 - 1.2.2. Теги HTML
 - 1.2.3. Заголовки
 - 1.2.4. Абзац
 - 1.2.5. Ссылка
 - 1.2.6. Об атрибутах
 - 1.2.7. Один тег в другом. Родители и дети
 - 1.2.8. Изображения
- 1.3. Базовые CSS-свойства
 - 1.3.1. Собрать лендинг
 - 1.3.2. Размеры в пикселях
 - 1.3.3. Размеры в процентах и долях
 - 1.3.4. Цвета в HTML
 - 1.3.5. Фон элемента
 - 1.3.6. Позиция, размер, повтор фона
 - 1.3.7. Прозрачность
 - 1.3.8. Коробка в коробке
 - 1.3.9. Наследование
 - 1.3.10. Типографика
 - 1.3.11. Больше вёрстки
- 1.4. Больше CSS
 - 1.4.1. Пик вёрстки
 - 1.4.2. Новая секция
 - 1.4.3. Классы
 - 1.4.4. Приятное и полезное
 - 1.4.5. Типографика под контролем
 - 1.4.6. Поток и блочная модель
 - 1.4.7. Внешние и внутренние отступы
 - 1.4.8. Короткая запись свойств
 - 1.4.9. Границы
 - 1.4.10. Внешний и внутренний фокус элемента
 - 1.4.11. Блочные + строчные
 - 1.4.12. Расположение элементов по центру: `margin: auto;`
 - 1.4.13. Несколько классов
 - 1.4.14. Тени
 - 1.4.15. Подвал сайта
 - 1.4.16. Центрировать по вертикали
 - 1.4.17. Один шаг
 - 1.4.18. Вы справились!
- 1.5. JavaScript. Начало
 - 1.5.1. Новая цель
 - 1.5.2. JavaScript?
 - 1.5.3. Незаменимый инструмент
 - 1.5.4. Числа
 - 1.5.5. Строки
 - 1.5.6. Арифметика строк
 - 1.5.7. Переменные
 - 1.5.8. Магия присваивания

- 1.5.9. Что происходит?
- 1.6. JavaScript. Приземление в реальность
 - 1.6.1. Мастер прокрастинации
 - 1.6.2. Подключите JavaScript к HTML
 - 1.6.3. Массивы
 - 1.6.4. Случайные числа
 - 1.6.5. Математика ниже плинтуса
 - 1.6.6. Функции
 - 1.6.7. Функции с аргументами
 - 1.6.8. Возвращаемое значение
 - 1.6.9. Выбор и изменение элементов страницы
 - 1.6.10. Реакция на действия пользователя
 - 1.6.11. Булевы величины
 - 1.6.12. Условия в теории
 - 1.6.13. Условия на практике
 - 1.6.14. Объекты
 - 1.6.15. Подключение внешних библиотек
 - 1.6.16. Разделение труда
 - 1.6.17. Циклы
- 1.7. Флексбокс-вёрстка
 - 1.7.1. Флексбокс-вёрстка. Введение
 - 1.7.2. Преобразование во флекс-контейнер
 - 1.7.3. Флекс-контейнер с несколькими элементами
 - 1.7.4. Терминология флекса
 - 1.7.5. Вложенные флекс-контейнеры
 - 1.7.6. Направление внутри флекс-контейнера
 - 1.7.7. Управление положением содержимого флекс-контейнера
 - 1.7.8. Выравнивание флекс-элементов в контейнере
 - 1.7.9. Перенос элементов при сжатии окна браузера
 - 1.7.10. Свойства флекс-элементов. Порядок
 - 1.7.11. Свойства флекс-элементов. Выравнивание отдельного элемента
 - 1.7.12. Свойства флекс-элементов. Базовая ширина элемента
 - 1.7.13. А теперь сделаем всё красиво
 - 1.7.14. Что мы еще не сказали о флекс?
 - 1.7.15. Заключение
- 1.8. Расширенные возможности HTML и CSS
 - 1.8.1. Расширенные возможности HTML и CSS. Введение
 - 1.8.2. Язык документа и его элементов
 - 1.8.3. Описание сайта для браузеров и поисковых машин
 - 1.8.4. Разнообразие контейнеров HTML5
 - 1.8.5. Семантика и выразительность HTML
 - 1.8.6. Методология БЭМ
 - 1.8.7. Блоки
 - 1.8.8. Элементы
 - 1.8.9. Модификаторы в БЭМ
 - 1.8.10. Способы модификации блоков
 - 1.8.11. Приведём в порядок код
 - 1.8.12. Закрепим навык разметки и понимание семантики
 - 1.8.13. Идентификаторы
 - 1.8.14. Псевдоклассы

- 1.8.15. Псевдоэлементы и селекторы атрибутов
- 1.8.16. Заключение
- 1.9. Позиционирование элементов
 - 1.9.1. Позиционирование элементов. Введение
 - 1.9.2. Понятие потока (flow) и статическое позиционирование
 - 1.9.3. Относительное позиционирование
 - 1.9.4. Фиксированное позиционирование
 - 1.9.5. Абсолютное позиционирование
 - 1.9.6. z-index
 - 1.9.7. Заключение
- 1.10. Структура файлов, пути к файлам
 - 1.10.1. Структура файлов, пути к файлам. Введение
 - 1.10.2. Файл
 - 1.10.3. Линейная структура и иерархия. Разделение файлов по типам
 - 1.10.4. Абсолютные и относительные пути к файлам
 - 1.10.5. Абсолютный путь и тег link
 - 1.10.6. Относительные пути к файлам
 - 1.10.7. CSS-директивы. @import
 - 1.10.8. Заключение
- 1.11. Git для самых маленьких
 - 1.11.1. Git для самых маленьких. Введение
 - 1.11.2. Установка Git
 - 1.11.3. Ещё один инструмент — командная строка
 - 1.11.4. Шпаргалка: командная строка
 - 1.11.5. Настройка Git
 - 1.11.6. Подключаем Git к проекту
 - 1.11.7. Делаем первый коммит
 - 1.11.8. Командная работа. GitHub
 - 1.11.9. Регистрация на GitHub
 - 1.11.10. Генерируем SSH-ключи
 - 1.11.11. Связываем локальный и удалённый репозитории
 - 1.11.12. Синхронизация локального и удалённого репозитория
 - 1.11.13. Шпаргалка: Git
 - 1.11.14. Клонирование репозитория
- 1.12. Дока о самом интересном
- 1.13. Декомпозиция проекта по вёрстке
- 1.14. Проект. Научиться учиться: вёрстка
 - 1.14.1. Перед проектом
 - 1.14.2. Проект. Научиться учиться: вёрстка
 - 1.14.3. Чек-лист проекта «Научиться учиться: вёрстка»
- 2. Расширенные возможности HTML и CSS
 - 2.1. Интро в спринт
 - 2.2. Как думать, как разработчик
 - 2.3. Полезно знать перед стартом
 - 2.3.1. Полезно знать перед стартом. Введение
 - 2.3.2. Кроссбраузерность, вендорные префиксы
 - 2.3.3. Подключение шрифтов к странице
 - 2.4. HTML: видео, iframe, embed, API
 - 2.4.1. HTML: видео, iframe, embed, API. Введение
 - 2.4.2. iframe

- 2.4.3. Кроссбраузерность iframe
- 2.4.4. Виджеты-информеры
- 2.4.5. Тег
- 2.4.6. Тег
- 2.4.7. API YouTube
- 2.4.8. JS Scroll
- 2.4.9. Заключение
- 2.5. Анимации и трансформации
 - 2.5.1. Анимации и трансформации. Введение
 - 2.5.2. Трансформации
 - 2.5.3. Плавность переходов
 - 2.5.4. Тени
 - 2.5.5. Линейный градиент
 - 2.5.6. Радиальный градиент
 - 2.5.7. Переполнение блоков
 - 2.5.8. Кейфреймы
 - 2.5.9. Анимации
 - 2.5.10. Закрепим навыки
 - 2.5.11. Заключение
- 2.6. Разметка и вёрстка форм
 - 2.6.1. Разметка и вёрстка форм. Введение
 - 2.6.2. Что такое форма?
 - 2.6.3. Поля ввода с тегом input
 - 2.6.4. Минимальные и максимальные значения
 - 2.6.5. Поля загрузки, сброса и отправки данных
 - 2.6.6. Поля ввода с другим синтаксисом
 - 2.6.7. Ярлыки
 - 2.6.8. Передаваемые значения
 - 2.6.9. Поля множественного и единичного выбора
 - 2.6.10. Подсказки в текстовых полях
 - 2.6.11. Обязательные поля
 - 2.6.12. Особенности стилизации текстовых полей ввода, кнопок и ярлыков
 - 2.6.13. Селекторы атрибутов
 - 2.6.14. Стилизация переключателей и полей выбора
 - 2.6.15. Стилизация плейсхолдеров
 - 2.6.16. В работу!
 - 2.6.17. Заключение
- 2.7. Пути к файлам и организация файлов по БЭМ
 - 2.7.1. Пути к файлам и организация файлов по БЭМ. Введение
 - 2.7.2. Объясним всё и сразу аналогией
 - 2.7.3. Подключаем нужные блоки, элементы и модификаторы к странице
 - 2.7.4. Структура для блоков
 - 2.7.5. Структура для элементов
 - 2.7.6. Структура для модификаторов
 - 2.7.7. Глобальные стили
 - 2.7.8. Подключаем код сторонних разработчиков
 - 2.7.9. Заключение
- 2.8. Продвинутый Git
 - 2.8.1. Как работать в командной строке быстро
 - 2.8.2. Копирование и перемещение файлов

- 2.8.3. Просмотр и редактирование файлов в командной строке
- 2.8.4. Шпаргалка: bash
- 2.8.5. Ещё раз про git add и git commit
- 2.8.6. Логирование и хеширование
- 2.8.7. Просмотр изменений: git diff
- 2.8.8. Что делать, если закоммитили что-то не то
- 2.8.9. Где git изменения хранит?
- 2.8.10. Шпаргалка: git
- 2.8.11. Оформление репозитория. README.md
- 2.9. Дебаггинг в вёрстке
- 2.10. Дока о самом интересном
- 2.11. Проект. Научиться учиться: анимация, шрифты, фреймы
 - 2.11.1. 1.Проект. Научиться учиться: анимация, шрифты, фреймы
 - 2.11.2. 2.Чек-лист проекта «Научиться учиться: анимация, шрифты, фреймы»
- 3. Адаптивная вёрстка и работа с макетом
 - 3.1. Асфальт реальности. Макет и адаптивная вёрстка. Введение
 - 3.2. Работа с макетом
 - 3.2.1. Работа с макетом. Введение
 - 3.2.2. Создание цифрового продукта. Часть I.
 - 3.2.3. Что такое макет и где его делают?
 - 3.2.4. Знакомство с Figma
 - 3.2.5. Страница —> Фрейм —> Группа —> Элемент
 - 3.2.6. Работа с отступами, текстом и цветами
 - 3.2.7. Экспорт элементов и копирование кода
 - 3.2.8. Pixel perfect и диалог с дизайнером
 - 3.2.9. Заключение
 - 3.3. Дизайн в мире множества устройств
 - 3.3.1. Дизайн в мире множества устройств. Введение
 - 3.3.2. Разница между резиновой и адаптивной вёрсткой
 - 3.3.3. Мобильный или десктоп: с чего начать вёрстку?
 - 3.3.4. Адаптивный макет
 - 3.4. Неполный бриф, неточный макет
- 3.5. Grid Layout, часть 1
 - 3.5.1. Grid Layout, часть 1. Введение
 - 3.5.2. Что такое гриды
 - 3.5.3. Терминология гридов
 - 3.5.4. Столбцы и строки внутри грид-контейнера
 - 3.5.5. Создание отступов
 - 3.5.6. Функция repeat
 - 3.5.7. Единица измерения: фракция
 - 3.5.8. Расположение элементов в грид-контейнере
 - 3.5.9. Грид-области
 - 3.5.10. Заключение
- 3.6. Grid Layout, часть 2
 - 3.6.1. Grid Layout, часть 2. Введение
 - 3.6.2. Размеры неявных строк и колонок
 - 3.6.3. Свойство grid-auto-flow
 - 3.6.4. Функции minmax() и fit-content()
 - 3.6.5. Параметры auto-fill и auto-fit
 - 3.6.6. Выравнивание содержимого строк и колонок

- 3.6.7. Выравнивание элементов
- 3.6.8. Наложение элементов
- 3.6.9. Заключение
- 3.7. Разработка интерфейса для разных устройств
- 3.7.1. Разработка интерфейса для разных устройств. Введение
- 3.7.2. Относительные размеры блоков
- 3.7.3. Вычисляемые значения, функция calc()
- 3.7.4. Размеры margin и padding в процентах
- 3.7.5. Относительные размеры текстовых элементов
- 3.7.6. Особенности использования растровых изображений
- 3.7.7. Особенности использования векторных изображений
- 3.7.8. Оптимизация шрифтов под устройства с различными разрешениями
- 3.7.9. Метатеги корректного масштабирования страницы
- 3.7.10. Медиазапросы
- 3.7.11. Подходы к построению медиазапросов
- 3.7.12. Заключение
- 3.8. Что делать, если не получается сделать страницу адаптивной
- 3.9. GIT для взрослых
- 3.9.1. Git для взрослых. Введение и повторение
- 3.9.2. Ветки
- 3.9.3. Слияние
- 3.9.4. Деплой. Как сдавать проектную работу
- 3.9.5. Шпаргалка. Ветки в Git
- 3.9.6. Заключение
- 3.10. Дока о самом интересном
- 3.11. Проект. Путешествие по России

4. Учебно-методическое обеспечение самостоятельной работы обучающихся

Самостоятельная работа обучающихся включает в себя подготовку к контрольным вопросам и заданиям для текущего контроля и промежуточной аттестации по итогам освоения дисциплины приведенным в п. 5.

нет

5. Фонд оценочных средств для текущего контроля успеваемости и промежуточной аттестации по дисциплине (модулю)

5.1 Типовые задания, необходимые для оценки результатов обучения при проведении текущего контроля успеваемости с указанием критериев их оценивания:

5.1.1 Типовые задания (оценочное средство - Практическое задание) для оценки сформированности компетенции ПК-12:

Задание 2.1

В файле index.html создана базовая структура, подключён файл стилей. Создайте прототип входного экрана страницы о четырёх правилах вёрстки. Между тегами body добавьте элемент <header></header>.

Он ведёт себя точно так же, как и div, но браузеру и поисковым системам по такому названию проще понять местонахождение входной секции сайта.

Файл index.html

```
<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Четыре правила вёрстки</title>

<link rel="stylesheet" href="style.css">

<link rel="icon" type="image" href="https://code.s3.yandex.net/web-code/rules-favicon.ico">

</head>

<body>


</body>

</html>
```

Решение:

```
<body>

<header>


</header>

</body>
```

Критерии оценивания (оценочное средство - Практическое задание)

Оценка	Критерии оценивания
зачтено	Продемонстрированы все основные умения, решены все основные задачи с отдельными несущественным недочетами, выполнены все задания в полном объеме.

Оценка	Критерии оценивания
не зачтено	Уровень знаний ниже минимальных требований. Имели место грубые ошибки. При решении стандартных задач не продемонстрированы основные умения и базовые навыки

5.2. Описание шкал оценивания результатов обучения по дисциплине при промежуточной аттестации

Шкала оценивания сформированности компетенций

Уровень сформированности компетенций (индикатора достижения компетенций)	плохо	неудовлетворительно	удовлетворительно	хорошо	очень хорошо	отлично	превосходно
	не зачтено			зачтено			
<u>Знания</u>	Отсутствие знаний теоретического материала. Невозможность оценить полноту знаний вследствие отказа обучающегося от ответа	Уровень знаний ниже минимальных требований. Имели место грубые ошибки	Минимально допустимый уровень знаний. Допущено много негрубых ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько негрубых ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько несущественных ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Ошибок нет.	Уровень знаний в объеме, превышающем программу подготовки.
<u>Умения</u>	Отсутствие минимальных умений. Невозможность оценить наличие умений вследствие отказа обучающегося от ответа	При решении стандартных задач не продемонстрированы основные умения. Имели место грубые ошибки	Продemonстрированы основные умения. Решены типовые задачи с негрубыми ошибками. Выполнены все задания, но не в полном объеме	Продemonстрированы все основные умения. Решены все основные задачи с негрубыми ошибками. Выполнены все задания в полном объеме, но некоторые с недочетами	Продemonстрированы все основные умения. Решены все основные задачи. Выполнены все задания в полном объеме, но некоторые с недочетами.	Продemonстрированы все основные умения. Решены все основные задачи с отдельными несущественными недочетами, выполнены все задания в полном объеме	Продemonстрированы все основные умения. Решены все основные задачи. Выполнены все задания, в полном объеме без недочетов
<u>Навыки</u>	Отсутствие базовых навыков. Невозможность оценить наличие навыков вследствие отказа обучающегося от ответа	При решении стандартных задач не продемонстрированы базовые навыки. Имели место грубые ошибки	Имеется минимальный набор навыков для решения стандартных задач с некоторыми	Продemonстрированы базовые навыки при решении стандартных задач с некоторыми недочетами	Продemonстрированы базовые навыки при решении стандартных задач без ошибок и недочетов	Продemonстрированы навыки при решении нестандартных задач без ошибок и недочетов	Продemonстрирован творческий подход к решению нестандартных задач

			недочетами				
--	--	--	------------	--	--	--	--

Шкала оценивания при промежуточной аттестации

Оценка		Уровень подготовки
зачтено	превосходно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «превосходно», продемонстрированы знания, умения, владения по соответствующим компетенциям на уровне выше предусмотренного программой
	отлично	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «отлично».
	очень хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «очень хорошо»
	хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «хорошо».
	удовлетворительно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «удовлетворительно», при этом хотя бы одна компетенция сформирована на уровне «удовлетворительно»
не зачтено	неудовлетворительно	Хотя бы одна компетенция сформирована на уровне «неудовлетворительно».
	плохо	Хотя бы одна компетенция сформирована на уровне «плохо»

5.3 Типовые контрольные задания или иные материалы, необходимые для оценки результатов обучения на промежуточной аттестации с указанием критериев их оценивания:

5.3.1 Типовые задания (оценочное средство - Тест) для оценки сформированности компетенции ПК-12

1.1 Какие элементы HTML-кода определяют смысловую структуру и иерархию документа?

Комментарии

Мета-данные

+ Семантические и структурные теги

Символы юникода

Критерии оценивания (оценочное средство - Тест)

Оценка	Критерии оценивания
зачтено	Продemonстрированы все основные умения, решены все основные задачи с отдельными

Оценка	Критерии оценивания
	несущественным недочетами, выполнены все задания в полном объеме.
не зачтено	Уровень знаний ниже минимальных требований. Имели место грубые ошибки. При решении стандартных задач не продемонстрированы основные умения и базовые навыки

5.3.2 Типовые задания (оценочное средство - Проект) для оценки сформированности компетенции ПК-12

Разработать фронтенд для сайта с авторизацией

Критерии оценивания (оценочное средство - Проект)

Оценка	Критерии оценивания
зачтено	Продemonстрированы все основные умения, решены все основные задачи с отдельными несущественным недочетами, выполнены все задания в полном объеме.
не зачтено	Уровень знаний ниже минимальных требований. Имели место грубые ошибки. При решении стандартных задач не продемонстрированы основные умения и базовые навыки

6. Учебно-методическое и информационное обеспечение дисциплины (модуля)

Основная литература:

1. Диков А. В. Клиентские технологии веб-дизайна. HTML5 и CSS3 : учебное пособие для вузов / Диков А. В. - 2-е изд., стер. - Санкт-Петербург : Лань, 2023. - 188 с. - Книга из коллекции Лань - Информатика. - ISBN 978-5-507-46740-2., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=862749&idb=0>.
2. Янцев В. В. Разработка web-страниц на HTML, CSS и JavaScript : учебное пособие для вузов / Янцев В. В. - Санкт-Петербург : Лань, 2024. - 148 с. - Книга из коллекции Лань - Информатика. - ISBN 978-5-507-49640-2., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=900593&idb=0>.
3. Минязев Р. Ш. Скриптовые языки web-программирования (JavaScript, PHP, html/CSS) : учебно-методическое пособие / Минязев Р. Ш. - Казань : КНИТУ-КАИ, 2022. - 60 с. - Рекомендовано к изданию Учебно-методическим управлением КНИТУ-КАИ. - Книга из коллекции КНИТУ-КАИ - Информатика. - ISBN 978-5-7579-2632-2., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=895710&idb=0>.

Дополнительная литература:

1. Тиге Дж.К. DHTML и CSS : монография / Тиге Дж.К. - Москва : ДМК-пресс, 2023. - 560 с. - ISBN 978-5-89818-455-1., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=912682&idb=0>.

2. Сухов К.К. HTML5 - путеводитель по технологии : монография / Сухов К.К. - Москва : ДМК-пресс, 2023. - 353 с. - ISBN 978-5-89818-532-9., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=912759&idb=0>.

Программное обеспечение и Интернет-ресурсы (в соответствии с содержанием дисциплины):

Visual Studio Code (свободно распространяемый) <https://azure.microsoft.com/ru-ru/products/visual-studio-code/>

Visual Studio Code (свободно распространяемый): <https://code.visualstudio.com/>

Рекомендуемая версия: 1.98

7. Материально-техническое обеспечение дисциплины (модуля)

Учебные аудитории для проведения учебных занятий, предусмотренных образовательной программой, оснащены мультимедийным оборудованием (проектор, экран), техническими средствами обучения, компьютерами.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечены доступом в электронную информационно-образовательную среду.

Программа составлена в соответствии с требованиями ОС ННГУ по направлению подготовки/специальности 09.03.04 - Программная инженерия.

Автор(ы): Борисов Николай Анатольевич, кандидат технических наук, доцент.

Заведующий кафедрой: Баркалов Константин Александрович, доктор технических наук.

Программа одобрена на заседании методической комиссии от 02.12.2024, протокол № 5.