

ООО «Алгоритмика-Пермь»

УТВЕРЖДАЮ

Директор ООО «Алгоритмика-Пермь».



Скокова А.В.

Дополнительная общеобразовательная программа –

Дополнительная общеразвивающая программа

«Основы фронтенд-разработки»

Направленность: техническая

Уровень программы: ознакомительный

Возраст учащихся: 15–18 лет

Срок реализации: 8 месяцев (64 академических часа)

Авторы-составители:

Скокова А.В.

г. Пермь, 2024 год

1. Пояснительная записка

1.1. Направленность и уровень программы

Информатизация современного общества постоянно усиливается, что определяет новые задачи в развитии технического творчества. Специалисты, которые способны объединить в практической деятельности технические и информационные знания, становятся всё более востребованы на рынке труда. Веб-разработка — это область информационных технологий, зародившаяся в результате стремительного роста сети Интернет. Этот перспективный вид деятельности позволяет воплощать всевозможные творческие идеи специалистов в области информационных технологий. На рынке труда возникают новые специальности, в том числе — frontend-разработчики. Увлечённость людей ресурсами Интернета растёт с каждым годом, грамотно разработанные веб-сайты пользуются большой популярностью. Поэтому подготовка специалистов в данной области становится всё более актуальной задачей.

Разработка веб-сайта — это сложный комплексный процесс, который предполагает наличие большого количества этапов и задач, а распределение обязанностей между backend- и frontend-специалистами в компании позволяет оптимизировать многие процессы и упростить логику распределения ответственности. Поэтому работать в индустрии веб-сайтов могут не только крупные компании, но и отдельные самостоятельные разработчики.

Frontend-разработчик занимается презентационной частью веб-сайта, то есть определяет то, как будет выглядеть пользовательский интерфейс и связанные с ним компоненты. Языки HTML и CSS обладают достаточно низким порогом вхождения, не требуют особых навыков у ученика перед стартом обучения. Знание основ этих двух языков позволяет разработать простейший веб-сайт. А язык программирования JavaScript даёт возможность создавать динамически изменяющийся контент, управлять мультимедийными файлами и реагировать на действия пользователя на странице.

Современная тенденция заключается в том, что всё больше людей регулярно используют веб-сайты в сети Интернет, так как они предоставляют пользователям множество полезных инструментов и знаний. Но это приводит и к популярной проблеме — ребёнок не всегда умеет правильно определять веб-сайты, которые несут полезную информацию. Поэтому важно научить ребёнка

осознанно взаимодействовать с ними, развить его мышление и выработать умение получать знания.

Направленность дополнительной образовательной программы — *техническая*. Она заключается в развитии технического и творческого мышления у детей старшего подросткового возраста через разработку веб-сайтов. Также у детей развивается умение составлять план деятельности, растёт стремление к познавательным активностям, улучшаются исследовательские и прикладные способности.

Развитие навыков веб-разработки является одним из способов формирования интереса учащихся к техническим направлениям деятельности, а также совершенствования алгоритмического, критического, пространственного и креативного мышления.

Курс предусматривает отбор учеников по возрасту и наличию начальных навыков программирования на любом языке. Программа курса составлена с учётом актуальных трендов сферы frontend-разработки и задач национального проекта «Образование».

1.2. Актуальность

В современном образовании увеличивается количество часов на изучение информатики и компьютерной грамотности. Современные пользовательские интерфейсы, представленные в сети Интернет, относительно легко осваиваются детьми старшего подросткового возраста. Поэтому это поколение подростков имеет возможность стать поколением «продвинутых пользователей».

Новизна программы заключается в комплексном подходе овладения навыками. Сначала учащиеся изучают технологию, а затем применяют её на практике, причем они не только изучают приемы разработки, но и учатся комбинировать их в практических задачах. В процессе обучения активно используются принципы интеграции теоретического обучения с практической и исследовательской деятельностью.

Педагогическая целесообразность данной программы заключается в том, что она создаёт благоприятные условия для социально-культурного и профессионального самоопределения, развития познавательной активности и творческой самореализации личности ребёнка, а также способствует подготовке

к самостоятельной жизни в современном обществе со стремительно развивающимися информационными технологиями.

В течение курса ученики осваивают всё большее количество инструментов, создают разнообразные проекты на основе получаемых знаний и развивающихся умений.

1.3. Цель и задачи программы;

Цель курса — заложить основы профессиональных знаний в области разработки веб-сайтов, научить обучающихся ориентироваться и продуктивно действовать в информационном интернет-пространстве и способствовать развитию их творческого потенциала.

Обучающие задачи:

1. Познакомить с основными инструментами для создания веб-страниц: языком гипертекстовой разметки HTML, каскадными таблицами стилей CSS и языком программирования JavaScript.

2. Познакомить с возможностями программ, которые используют frontend-разработчики: графический редактор Figma, редактор кода Visual Studio Code, веб-сервис GitHub.

3. Сформировать навык использования графического редактора при создании веб-сайта или веб-приложения по макету.

4. Сформировать навык разработки сайта со строгим техническим заданием.

6. Сформировать навык работы в команде.

7. Сформировать навык разработки собственного дизайна веб-страницы.

8. Сформировать навык проектной работы (от постановки цели и задач до презентации результата).

Развивающие задачи:

1. Формировать и развивать логическое, алгоритмическое и критическое мышление.

2. Формировать навык публичных выступлений и презентаций.

3. Формировать навык планирования деятельности и основ тайм-менеджмента.

4. Развивать насмотренность и восприимчивость к эстетике в дизайне веб-страниц.

5. Совершенствовать диалогическую речь: умение слушать собеседника, понимать вопросы, смысл заданий, задавать вопросы, отвечать на них.

Воспитательные задачи:

1. Воспитывать умение сотрудничать со сверстниками и умение подчинять свои интересы определённым правилам.

2. Формировать культуру взаимодействия с чужой работой и авторским правом.

1.4. Формы и режим занятий

Занятия проходят 1 раз в неделю и длятся 2 академических часа (1 академический час — 40 минут). Дети занимаются в классе с учителем, самостоятельно работая за компьютерами (непрерывно — не более 20 минут). Занятие разделено перерывом, который длится 5 минут. Основные формы работы — индивидуальная, фронтальная. Печатные материалы не предусмотрены.

Формы обучения:

- Обучение от общего к частному (дедуктивный метод) и от частного к общему (индуктивный).

- Поощрение вопросов и свободных высказываний по теме.

- Уважение и внимание к каждому ученику.

- Создание мотивационной среды обучения.

- Создание условий для дискуссий и развития мышления учеников при достижении учебных целей вместо простого одностороннего объяснения темы преподавателем.

Занятие состоит из следующих блоков:

- Обсуждение и постановка целей на урок (5 минут).
- Повторение (5 минут).
- Изучение нового материала (15 минут).
- Практика (15 минут).
- Перерыв (5 минут)
- Изучение нового материала (15 минут).
- Практика (20 минут).
- Рефлексия: подведение итогов занятия (5 минут).

1.5. Срок реализации программы

Содержание курса «Основы фронтенд-разработки» для детей 15–18 лет рассчитано на обучение в течение 8 месяцев (32 занятия — 64 академических часа). Отбор на курс осуществляется по возрасту и наличию начальных навыков программирования на любом языке.

1.6. Планируемые результаты

Достижение цели и задач образовательной программы предполагает получение следующих результатов:

Планируемый результат	Способ достижения	Критерий достижения образовательного результата
Личностные результаты обучения		
Освоение социальной роли учащегося и формирование личностного смысла учения.	Демонстрация связи между способностью выполнить интересную задачу и наличием/отсутствием соответствующих знаний.	За отведённое время ученик пытается справиться с базовой задачей и разнообразить решение собственными деталями.
Развитие навыков сотрудничества с взрослыми и сверстниками и умения находить выход из спорных ситуаций.	Представление frontend-разработки как сферы, в которой актуальны навыки работы в команде.	Ученик не боится просить помощи и сам пытается помогать одноклассникам и учителю.
Овладение навыками адаптации в динамично изменяющемся и развивающемся мире.	Обсуждение актуальных направлений и трендов frontend-разработки, демонстрирующих необходимость не только овладения предметными навыками, но и умения непрерывно учиться.	Работа над проектом начинается с исследования предложенной темы.

Метапредметные навыки		
Умение выделять интересные для себя направления. Готовность самостоятельно разбираться в новом функционале программ.	При изучении новой темы учитель показывает интересные и перспективные направления, выходящие за рамки темы.	Ученик проявляет интерес к предметной области и пытается дополнять базовые задания собственными идеями.
Умение работать над задачей, осознавая ограничения, связанные с авторским правом.	В рамках плана работы над задачей учитель предлагает бесплатные ресурсы с материалами для работы.	Ученик предпочитает собственную уникальную работу хорошей копии чужой работы.
Умеет презентовать свою работу.	Презентация своих проектов. Учитель и другие ученики дают обратную связь, учитель также даёт советы, каким образом лучше это делать.	Ученик старается быть понятным аудитории и заинтересовать её выполненной работой.
Способность быть частью профессионального сообщества.	Мотивация изучать работы коллег, делиться собственной работой и воспринимать обратную связь по ней в сообществе.	Ученик выкладывает работы в профессиональное интернет-сообщество и адекватно воспринимает реакцию сообщества на неё.
Предметные навыки		
Знание актуальных направлений frontend-	Непрерывное знакомство с новыми возможностями	Ученик устойчиво ориентируется в

разработки и соответствующего программного обеспечения.	инструментов веб-разработки.	возможностях HTML, CSS и языка программирования JavaScript.
Знание функциональных возможностей языка JavaScript.	Выполнение учебных проектов в редакторах по плану, предложенному учителем.	Ученик умеет подбирать подходящие инструменты языка JavaScript для решения поставленной задачи.
Применение общих правил современного веб-дизайна.	Обсуждение этих правил и демонстрация их использования в учебных проектах.	Ученик проявляет устойчивый интерес к эстетике в дизайне веб-сайтов и стремится сделать свою работу красивой и современной.
Умение дорабатывать творческий проект по полученной обратной связи.	Учитель создаёт комфортную атмосферу и организует постоянный взаимовыгодный обмен обратной связью по итогам работы.	Ученик проявляет инициативу и запрашивает обратную связь. Дорабатывает проект, опираясь на отзывы.

В течение курса ученики создадут несколько индивидуальных проектов и один групповой проект. Важным результатом курса является публикация проектов на хостинге и в социальной сети ВКонтакте.

1.7. Нормативно-правовое обеспечение

1. Федеральный Закон РФ от 29.12.2012 г. № 273 «Об образовании в Российской Федерации» (в редакции Федерального закона от 31.07.2020 № 304-ФЗ «О внесении изменений в Федеральный закон “Об образовании в Российской Федерации” по вопросам воспитания обучающихся») (далее

– 273-ФЗ).

2. Приказ Минпросвещения России от 09.11.2018 г. № 196 «Об утверждении порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»; Концепцию развития дополнительного образования детей от 04.09.2014 г. № 1726-р.
3. Постановление Главного государственного санитарного врача РФ от 28.09.2020 № 28 «Об утверждении санитарных правил СП 2.4.3648-20 “Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи”».

2. ие программы

Содержан

2.1. Учебный (тематический) план

№ п/п	Наименование раздела/темы	Количество часов			Форма аттестации (контроля)
		всего	теория	практика	
Модуль 1. Основы HTML		8	4	4	
1	Знакомство с HTML	2	1	1	Задания на платформе
2	Работа с текстом и изображениями	2	1	1	Задания на платформе
3	Работа с таблицами	2	1	1	Задания на платформе
4	Создание сайта-визитки	2	1	1	Задания на платформе
Модуль 2. Основы CSS		8	4	4	
5	Селекторы и классы	2	1	1	Задания на платформе
6	Контейнеры	2	1	1	Задания на платформе
7	Дизайн макета	2	1	1	Задания на платформе
8	Создание стилей по макету	2	1	1	Задания на платформе
Модуль 3. Программирование на языке JavaScript		14	7	7	
9	Введение в язык JavaScript. Переменные и типы данных, DOM	2	1	1	Задания на платформе
10	Функции в JavaScript	2	1	1	Задания на платформе
11	Управляющие конструкции в JavaScript: условия, циклы, массивы	2	1	1	Задания на платформе
12	ООП в JavaScript	2	1	1	Задания на платформе
13	Анимация	2	1	1	Задания на платформе
14	Асинхронность и BOM в JavaScript	2	1	1	Задания на платформе
15	Работа со строками и cookie	2	1	1	Задания на платформе
Модуль 4. Проект «Квиз»		8	4	4	
16	Создание структуры документа. Предварительная верстка	2	1	1	Задания на платформе
17	Чистовая верстка. Работа со стилями	2	1	1	Задания на платформе

18	Работа с Figma. Создание собственной айдентики. Разработка UI для проекта на JavaScript.	2	1	1	Задания на платформе, разработка макета в графическом редакторе Figma
19	Проектирование UX и введение в юзабилити-тестирование.	2	1	1	Задания на платформе, разработка макета в графическом редакторе Figma
Модуль 5. Git и GitHub		8	4	4	
20	Введение Git. Создание и настройка репозитория, развертывание проекта	2	1	1	Задания на платформе
21	VSCode. Знакомство, настройка, функционал	2	1	1	Задания на платформе, разработка проекта в редакторе кода VSCode
22	Основные команды. Первый commit	2	1	1	Задания на платформе, разработка проекта в редакторе кода VSCode
23	Модернизация проекта «Квиз» при помощи VSC и деплой на GitHub	2	1	1	Задания на платформе, разработка проекта в редакторе кода VSCode
Модуль 6. Групповой проект Moody		8	4	4	
24	Особенности командной разработки. Распределение ролей	2	1	1	Задания на платформе, разработка проекта в редакторе кода VSCode
25	Введение в Miro. Создание структуры и mind map проекта	2	1	1	Задания на платформе, разработка проекта в редакторе кода VSCode
26	Групповая работа над проектом в GitHub и VSCode	2	1	1	Задания на платформе, разработка проекта в редакторе кода VSCode
27	Создание проекта: макеты, верстка, стили, программирование, сборка, деплой	2	1	1	Задания на платформе, разработка проекта в редакторе кода VSCode
Модуль 7. Деплой проекта на сервер		8	4	4	
28	Знакомство с сервисом Heroku. Регистрация и настройка	2	1	1	Задания на платформе, разработка проекта в редакторе кода VSCode
29	Создание виртуального окружения	2	1	1	Задания на платформе, разработка проекта в редакторе кода VSCode
30	Деплой проекта «Квиз» и публикация в социальной сети ВКонтакте	2	1	1	Задания на платформе, разработка проекта в редакторе кода VSCode
31	Деплой проекта Moody и создание аккаунта проекта в Telegram.	2	1	1	Задания на платформе, разработка проекта в редакторе кода VSCode
Выпускной		2	1	1	

Выпускной	2	1	1	Задания на платформе
Всего часов:	64	32	32	

2.2. Содержание учебного (тематического) плана

Модуль 1. Основы HTML (8 часов)

1.1. Знакомство с HTML

Теория (1 час): Понятия «структура сайта», «тег». Знакомство с редактором кода на платформе.

Практика (1 час): Выполнение заданий на платформе.

1.2. Работа с текстом и изображениями

Теория (1 час): Понятия «атрибут», «блочный элемент», «внешний отступ».

Практика (1 час): Выполнение заданий на платформе.

1.3. Работа с таблицами

Теория (1 час): Понятия «таблицы в HTML», «внутренний отступ». Вёрстка календаря с помощью HTML-таблиц.

Практика (1 час): Выполнение заданий на платформе.

1.4. Создание сайта-визитки

Теория (1 час): Повторение основ HTML в форме викторины. Ознакомление со структурой сайта-визитки.

Практика (1 час): Выполнение заданий на платформе.

Модуль 2. Основы CSS (8 часов)

2.1. Селекторы и классы

Теория (1 час): Понятие «каскадные таблицы стилей». Изучение основных принципов работы с CSS, виды селекторов и их применение.

Практика (1 час): Выполнение заданий на платформе.

2.2. Контейнеры

Теория (1 час): Понятие «контейнер». Управление расположением элементов с применением свойства position.

Практика (1 час): Выполнение заданий на платформе.

2.3. Дизайн макета

Теория (1 час): Изучение инструментов графического редактора Figma.

Практика (1 час): Выполнение заданий на платформе и работа с графическим онлайн-редактором Figma.

2.4. Создание стилей по макету

Теория (1 час): Структура веб-страницы интернет-магазина. Анализ макета интернет-магазина в редакторе Figma.

Практика (1 час): Выполнение заданий на платформе и работа с графическим редактором Figma.

Модуль 3. Программирование на языке JavaScript (14 часов)

3.1. Введение в язык JavaScript. Переменные и типы данных, DOM

Теория (1 час): Понятие «DOM-дерево». Переменные и типы данных в JavaScript.

Практика (1 час): Выполнение заданий на платформе.

3.2. Функции в JavaScript

Теория (1 час): Понятия «функция» и «события». Обработка базовых событий.

Практика (1 час): Выполнение заданий на платформе.

3.3. Управляющие конструкции в JavaScript: условия, циклы, массивы

Теория (1 час): Логический тип данных и логические выражения.

Практика (1 час): Выполнение заданий на платформе.

3.4. ООП в JavaScript

Теория (1 час): Понятия «класс» и «объект». Принципы работы с ними.

Практика (1 час): Выполнение заданий на платформе.

3.5. Анимация

Теория (1 час): Принцип создания анимаций в CSS и JavaScript.

Практика (1 час): Выполнение заданий на платформе.

3.6. Асинхронность и BOM в JavaScript

Теория (1 час): Понятия «Окружение браузера» и BOM. Принцип работы асинхронного кода в JavaScript.

Практика (1 час): Выполнение заданий на платформе.

3.7. Работа со строками и cookie

Теория (1 час): Основные методы строк, принцип работы cookie-файлов.

Практика (1 час): Выполнение заданий на платформе.

Модуль 4. Проект «Квиз» (8 часов)

4.1. Создание структуры документа. Предварительная верстка

Теория (1 час): Основные инструменты и правила проектирования визуальной части приложения. Планирование и начало работы над проектом.

Практика (1 час): Выполнение заданий на платформе.

4.2. Чистовая верстка. Работа со стилями

Теория (1 час): Особенности работы со стилями разных объектов.

Практика (1 час): Выполнение заданий на платформе.

4.3. Работа с Figma. Создание собственной айдентики. Разработка UI для проекта на JavaScript

Теория (1 час): Понятие UI. Приёмы создания UI в графическом редакторе Figma.

Практика (1 час): Выполнение заданий на платформе и работа с графическим редактором Figma.

4.4. Проектирование UX и введение в юзабилити-тестирование

Теория (1 час): Понятие UX. Окончание работы над проектом.

Практика (1 час): Выполнение заданий на платформе и работа с графическим редактором Figma.

Модуль 5. Git и Github (8 часов)

5.1. Введение Git. Первое знакомство с GitHub

Теория (1 час): Особенности работы с распределённой системой управления версиями Git. Создание и настройка репозитория в GitHub.

Практика (1 час): Выполнение заданий на платформе и в системе GitHub.

5.2. VSCode. Знакомство, настройка, функционал

Теория (1 час): Особенности работы с редактором кода VSCode. Первичная настройка.

Практика (1 час): Выполнение заданий в редакторе кода VSCode.

5.3. Основные команды. Первый commit

Теория (1 час): Основные команды в Git.

Практика (1 час): Выполнение заданий в редакторе кода VSCode.

5.4. Модернизация проекта «Квиз» при помощи VSC и деплой на GitHub

Теория (1 час): Перенос проекта в редактор кода VSCode. Деплой проекта в репозиторий GitHub.

Практика (1 час): Выполнение заданий в редакторе кода VSCode.

Модуль 6. Групповой проект Moody (8 часов)

6.1. Особенности командной разработки. Распределение ролей

Теория (1 час): Организация работы над проектом в команде. Планирование и начало работы над проектом.

Практика (1 час): Выполнение заданий на платформе.

6.2. Введение в Miro. Создание структуры и mind map проекта

Теория (1 час): Знакомство с платформой для совместной работы распределённых команд Miro.

Практика (1 час): Выполнение заданий в Miro.

6.3. Групповая работа над проектом в GitHub и VSCode

Теория (1 час): Настройка репозитория и редактора кода для проекта.

Практика (1 час): Выполнение заданий в системе GitHub и в редакторе кода VSCode.

6.4. Создание проекта: макеты, верстка, стили, программирование, сборка, деплой

Теория (1 час): Разработка проекта согласно распределённым ролям.

Практика (1 час): Выполнение заданий в редакторе кода VSCode.

Модуль 7. Деплой проекта на сервер (8 часов)

7.1. Знакомство с сервисом Heroku. Регистрация и настройка

Теория (1 час): Начало работы с хостингом Heroku. Регистрация на платформе.

Практика (1 час): Выполнение заданий в редакторе кода VSCode.

7.2. Создание виртуального окружения

Теория (1 час): Понятие «виртуальное окружение». Особенности работы с виртуальным окружением.

Практика (1 час): Выполнение заданий в редакторе кода VSCode.

7.3. Деплой проекта «Квиз» и публикация в социальной сети ВКонтакте

Теория (1 час): Деплой проекта «Квиз» на веб-хостинг. Публикация проекта в социальной сети ВКонтакте.

Практика (1 час): Выполнение заданий в редакторе кода VSCode.

7.4. Деплой проекта Moody и создание аккаунта проекта в Telegram

Теория (1 час): Деплой группового проекта Moody на хостинг. Создание аккаунта проекта в Telegram.

Практика (1 час): Выполнение заданий в редакторе кода VSCode.

Выпускной (2 часа)

8.1 Создание портфолио frontend-разработчика

Теория (1 час): Правила создания портфолио frontend-разработчика. Инструменты создания цифрового портфолио.

Практика (1 час): Создание цифрового портфолио с описанием всех созданных на курсе проектов.

2.3 Формы аттестации и оценочные материалы

Текущий контроль сформированности результатов освоения программы осуществляется с помощью нескольких инструментов на нескольких уровнях:

- **на каждом занятии:** опрос, активность на занятии, выполнение обязательных и дополнительных заданий;
- **в конце каждого модуля:** проведение презентации (по желанию) финальных проектов модуля и их оценка, суммарная оценка достижений на занятиях модуля.

Для *промежуточного контроля* сформированности результатов освоения программы с помощью выполнения заданий на платформе используются следующие механики:

- классический тест (выбор одного или нескольких правильных ответов),
- заполнение пропусков,
- классификация,
- сопоставление,
- сортировка.

Оценка результатов, достигаемых на курсе, происходит в течение всего обучения и завершается на выпускном уроке:

- Преподаватель оценивает выполнение теоретических и практических заданий, а также учитывает успеваемость в течение всего курса.
- Преподаватель оценивает работу ученика над проектом (идея, процесс работы над идеей, публикация результата сообществу, восприятие обратной связи).

Уровни освоения программы:

- высокий — все проекты выполнены самостоятельно, ученик представлял результаты перед группой и воспринимал обратную связь;
- средний — все проекты выполнены с небольшой поддержкой учителя; ученик с помощью учителя представил результат перед группой;

- низкий — выполнено не менее 50% проектов, помощь учителя оказывалась на всех этапах выполнения проекта.

Требования по уровню подготовки выпускников:

по окончании курса ученик должен знать:

- как проходить полный цикл разработки проекта от задумки до деплоя;
- как представить свои проекты в виде портфолио.

по окончании курса ученик должен уметь:

- создавать интерактивные веб-страницы на HTML, CSS и JavaScript;
- работать над разработкой веб-страниц в редакторе кода Visual Studio Code;
- работать с макетами в графическом онлайн-редакторе Figma;
- публиковать проекты на удалённом сервере;

по окончании курса ученик должен использовать:

- графический редактор Figma для верстки веб-страниц по макету и редактор кода VSCode.

3. Организационно-педагогические условия реализации программы

3.1. Организационно-педагогические условия реализации программы.

Материально-технические условия реализации программы:

- помещение (предпочтительно изолированное);
- качественное освещение и возможность проветривания;
- санузел поблизости от аудитории;
- 10 рабочих мест: стол, стул, розетка;
- компьютеры на каждое рабочее место;
- Интернет (не менее 15 Мбит/сек);
- проектор, подключённый к компьютеру для учителя, выводящий изображение на экран;
- пульт для переключения слайдов (желательно);
- платформа «Алгоритмика», открывается через браузер Google Chrome;
- логины и пароли учеников для входа на платформу;
- магнитно-маркерная доска или флипчарт;
- мел или маркер для учителя;
- канцелярия: бумага А4, ручки/карандаши;
- среда кроссплатформенной разработки Visual Studio Code;

- в конце курса диплом (вручается ученику на выпускном);

Требования к ПК:

- обязательно: колонки, микрофон, монитор диагональю не менее 15" 1366×768;
- опционально: наушники, веб-камера;
- операционная система Windows 8.1,10 / Mac OS X High Sierra 10.13+ / Linux: Ubuntu 20.04, Ubuntu 18.04;
- браузер Google Chrome, обновлённый до последней версии.

Учебно-методическое и информационное обеспечение программы:

- поурочные методические рекомендации к занятиям;
- тематические презентации;
- задания на платформе;
- справочный теоретический материал, размещённый на платформе;
- бонусные задания на платформе;
- задачи со свободным решением для реализации в среде разработки на базе платформы «Алгоритмика».

Календарный учебный график

№ п/п	Дата	Тема занятия	Форма занятия	Кол-во часов	Форма контроля
1		Знакомство с HTML	Интерактивная	2	Задания на платформе
2		Работа с текстом и изображениями	Интерактивная	2	Задания на платформе
3		Работа с таблицами	Интерактивная	2	Задания на платформе
4		Создание сайта-визитки	Интерактивная	2	Задания на платформе
5		Селекторы и классы	Интерактивная	2	Задания на платформе
6		Контейнеры	Интерактивная	2	Задания на платформе
7		Дизайн макета	Интерактивная	2	Задания на платформе
8		Создание стилей по макету	Интерактивная	2	Задания на платформе
9		Введение в язык JavaScript. Переменные и типы данных, DOM	Интерактивная	2	Задания на платформе
10		Функции в JavaScript	Интерактивная	2	Задания на платформе
11		Управляющие конструкции в JavaScript: условия, циклы, массивы	Интерактивная	2	Задания на платформе
12		ООП в JavaScript	Интерактивная	2	Задания на платформе
13		Анимация	Интерактивная	2	Задания на платформе
14		Асинхронность и BOM в JavaScript	Интерактивная	2	Задания на платформе
15		Работа со строками и cookie	Интерактивная	2	Задания на платформе
16		Создание структуры документа. Предварительная вёрстка	Интерактивная	2	Задания на платформе
17		Чистовая вёрстка. Работа со стилями	Интерактивная	2	Задания на платформе
18		Работа с Figma. Создание собственной айдентики. Разработка UI для проекта на JavaScript	Интерактивная	2	Задания на платформе
19		Проектирование UX и введение в юзабилити-тестирование	Интерактивная	2	Задания на платформе
20		Введение Git. Создание и настройка репозитория, развёртывание проекта	Интерактивная	2	Задания на платформе

21		VSCode. Знакомство, настройка, функционал	Интерактивная	2	Задания на платформе и в редакторе кода VSCode
22		Основные команды. Первый commit	Интерактивная	2	Задания на платформе и в редакторе кода VSCode
23		Модернизация проекта «Квиз» при помощи VSC и деплой на GitHub	Интерактивная	2	Задания на платформе и в редакторе кода VSCode
24		Особенности командной разработки. Распределение ролей	Интерактивная	2	Задания на платформе и в редакторе кода VSCode
25		Введение в Miro. Создание структуры и mind map проекта	Интерактивная	2	Задания на платформе и в редакторе кода VSCode
26		Групповая работа над проектом в GitHub и VSCode	Интерактивная	2	Задания на платформе и в редакторе кода VSCode
27		Создание проекта: макеты, верстка, стили, программирование, сборка, деплой	Интерактивная	2	Задания на платформе и в редакторе кода VSCode
28		Знакомство с сервисом Heroku. Регистрация и настройка	Интерактивная	2	Задания на платформе и в редакторе кода VSCode
29		Создание виртуального окружения	Интерактивная	2	Задания на платформе и в редакторе кода VSCode
30		Деплой проекта «Квиз» и публикация в социальной сети ВКонтакте	Интерактивная	2	Задания на платформе и в редакторе кода VSCode
31		Деплой проекта Moody и создание аккаунта проекта в Telegram	Интерактивная	2	Задания на платформе и в редакторе кода VSCode
32		Выпускной	Интерактивная	2	Задания на платформе
Всего часов:				64	

Список литературы:

1. Джон Дакетт. HTML и CSS. Разработка и дизайн веб-сайтов. — М.: Эксмо, 2020. — 96 с.
2. Дэвид Макфарланд. Большая книга CSS. — М.: Питер, 2019. — 136 с.
3. Хавербеке Марейн. Выразительный JavaScript. — М.: Питер, 2022. — 59 с.
4. Дуглас Крокфорд. Как устроен JavaScript — М.: Питер, 2019. — 50 с.
5. Робсон Элизабет, Фримен Эрик. Изучаем программирование на JavaScript — М.: Питер, 2022. — 122 с.
6. Андрей Литвиненко. Figma — Основы работы. — Ridero, 2021. — 90 с.
7. Чакон Скотт, Штрауб Бен. Git для профессионального программиста. — М.: Питер, 2017. — 30 с.