

# Фулстек-разработчик

16 месяцев

продолжительность курса

13 проектов

в портфолио



HTML, CSS, JavaScript	Вводный модуль	31 час
HTML и CSS: лэйаут, доступность и интерактив	Спринт 1. Основы вёрстки Спринт 2. Доступность и подходы к организации стилей	6 недель
Дизайн, адаптивная вёрстка, декорирование	Спринт 3. Адаптивность и графика Спринт 4. Декорирование, подходы и инструменты	6 недель
Базовый JavaScript	Спринт 5. Основы JavaScript Спринт 6: Работы с событиями и формами в JS. Сборка кода и модульность	6 недель
Продвинутый JavaScript	Спринт 7: Продвинутая работа с данными, валидация, работа с API	3 недели
TypeScript и ООП	Спринт 8: TypeScript и ООП Спринт 9. Принципы разработки и архитектура ПО	6 недель
Интерфейсы с использованием React	Спринт 10: Основы React	3 недели
Роутинг и авторизация	Спринт 11: Роутинг и авторизация	3 недели
Тестирование React-приложений	Спринт 12: Тестирование React-приложений	3 недели
Проектный месяц	Работа над крупным проектом	1 месяц
Основы Node.js, Express и MongoDB	Спринт 13: Основы Node.js, Express и MongoDB	3 недели
PostgreSQL и Nest.js	Спринт 14: Разработка бэкенда на Nest.js Спринт 15: Базы данных и PostgreSQL	5 недель
Деплой и автоматизация	Спринт 16: Тестирование бэкенда, деплой и автоматизация	3 недели
Обеспечение безопасности приложения	Спринт 17: Обеспечение безопасности приложения	3 недели
Проектный месяц	Работа над крупным проектом	2 месяца

[31 час]

Вы узнаете, как устроены программы и обучение. Ознакомитесь с организационными деталями. Почувствуете себя в роли разработчика и поймёте, хотите ли развиваться в этом направлении. Изучите базовый синтаксис HTML и CSS. Научитесь размещать блоки на веб-странице, менять шрифт и цвета. Напишите простую программу на JavaScript и сделаете страницу интерактивной.

## Темы

---

1. Как все устроено

2. Первый код

3. Что такое HTML и CSS

4. Базовые CSS-свойства

5. Больше CSS

6. JavaScript. Начало

7. JavaScript. Приземление  
в реальность

8. О том, что дальше

9. Как устроено обучение

10. Вёрстка — продолжение  
дизайна

11. Готовимся писать код

12. Файлы в проекте

# HTML и CSS: лэйаут, доступность и интерактив

02

[6 недель],  
[2 проекта]

Создадите одностраничный сайт и напишете CSS для работающего приложения.

Научитесь работать со шрифтами в вебе. Разберётесь, как писать код, корректный для браузеров и поисковых систем. Изучите все способы организации каркаса вашей страницы. Научитесь настраивать страницы для поисковых роботов. Поработаете с видео-, аудио- и другими интерактивными элементами страниц. Поймёте, как и с какими событиями на странице может работать CSS. Разберётесь, какие элементы на странице может создавать язык стилей. Научитесь стилизовать элементы в разных состояниях и в разных ситуациях (например, в зависимости от порядка их появления). Разберётесь в основных принципах и инструментах создания доступных веб-страниц.

## Сprint 1. Основы вёрстки

- 1. Семантика и глобальные атрибуты
- 2. Шрифты и типографика
- 3. Флексбокс-вёрстка
- 4. Позиционирование элементов
- 5. Grid Layout
- 6. Кодстайл
- 7. Основы Bash и Git

## Сprint 2. Доступность и подходы к организации стилей

- 1. Настройка страницы и мета
- 2. Внешний встраиваемый контент: видео, iframe, API
- 3. Дополнение блочной модели
- 4. Псевдоклассы и псевдоэлементы
- 5. Доступность
- 6. Разметка форм
- 7. Чудные селекторы
- 8. Стилизация форм
- 9. Продвинутый Git и Bash

### Каникулы

[1 неделя]  
Время отдохнуть  
или доработать проекты

# Дизайн, адаптивная вёрстка, декорирование

---

03

[6 недель],  
[2 проекта]

Сделаете адаптивный сайт с разными цветовыми темами, а затем самостоятельно соберёте веб-приложение.

Научитесь комплексу приёмов, которые позволяют корректно отображать страницы на разных устройствах и с разными настройками. Подготовите графику для работы с разными экранами. Научитесь организовывать код для создания цветовой темы сайта. Разберёте инструменты декорирования ваших страниц. Градиенты, тени, фильтры, режимы смешивания — всё это здесь. Научитесь использовать векторную графику на страницах и пользоваться интерактивными инструментами, которые встроены в HTML. Вишенка на торте спринта — анимации страниц.

## Спринт 3. Адаптивность и графика

---

1. Подходы	5. Grid Layout, часть 2	8. Подходы к написанию вёрстки
2. Растворная графика	6. Разработка интерфейса для разных устройств	9. Git для взрослых
3. CSS-переменные	7. Кросбраузерность и вендорные префиксы	
4. Единицы измерения и функции		

## Спринт 4. Декорирование, подходы и инструменты

---

1. Использование SVG	5. Декорирование
2. Переходы и 2D трансформации	6. 3D трансформации
3. Анимации	7. Современные интерактивные элементы
4. Вариативные шрифты	

## Каникулы

[1 неделя]  
Время отдохнуть или доработать проекты

[6 недель],  
[1 проект]

Создадите интерактивную веб-страницу: сначала сверстаете основу, а затем шаг за шагом добавите JavaScript-логику.

Научитесь работать с пользовательскими действиями — кликами, формами, вводом данных. Пропишете поведение элементов страницы, разделите код на модули и соберёте проект с помощью современного инструмента Vite.

## Сprint 5. Основы JavaScript

1. От разметки  
к программированию  
2. Переменные и типы данных

3. Функции: простые  
концепции  
4. Основы работы с данными

5. Дебаггинг

## Сprint 6: Работы с событиями и формами в JS. Сборка кода и модульность

1. Что делать, когда не знаешь,  
что делать  
2. Знакомство с DOM  
3. Создание и изменение  
элементов страницы  
4. Работа с событиями

5. Работа с элементами форм и  
кнопками  
6. Модули в JS  
7. Сборка кода на Vite

[3 недели],  
[1 проект]

Научитесь делать запросы к серверу, обрабатывать ответы и строить взаимодействие интерфейса с реальными данными.

Создадите формы с валидацией, подключите пагинацию, реализуете логику, при которой пользователь получает обратную связь от сервера и видит результат своих действий на странице.

В финале соберёте полноценную таблицу, которая умеет загружать данные с сервера, обрабатывать ввод и обновляться без перезагрузки.

## Сprint 7: Продвинутая работа с данными, валидация, работа с API

1. Асинхронность

2. Работа с API: запросы к серверу

3. Флоу пользовательского действия с подтверждением сервера

4. Пагинация

5. Регулярные выражения

6. Валидация формы

## Каникулы

[1 неделя]

Время отдохнуть или доработать проекты

[6 недель],  
[1 сквозной проект]

Разработаете с нуля полноценное клиентское веб-приложение WEB-ларёк.

Сначала перепишете привычный JavaScript-код с использованием TypeScript, структурируете его с помощью классов и интерфейсов, а затем добавите архитектуру и паттерны проектирования и подготовите окружение для масштабируемой разработки.

В результате вы получите хорошо организованное приложение с чистым кодом, строгой типизацией и продуманной структурой.

## Сprint 8: TypeScript и ООП

- 1. Введение в TypeScript
- 2. Типизация кода
- 3. Введение в ООП
- 4. Принципы ООП
- 5. Основы TypeScript

## Сprint 9: ООП в интерфейсах: простые (и не только) концепции

- 1. Архитектура приложения
- 2. Расширенный TypeScript и ООП
- 3. Паттерны проектирования
- 4. Работа с окружением

## Каникулы

[1 неделя]  
Время отдохнуть  
или доработать проекты

# Интерфейсы с использованием React

---

07

[3 недели],  
[1 проект]

Реализуете страницу блога с настраиваемым через CSS-переменные дизайном.

Изучите библиотеку React и её экосистему. Научитесь создавать и структурировать проекты на React, создавать компоненты и управлять потоками данных. Освойте функциональный подход к разработке на React и научитесь писать эффективный производительный код с помощью хуков.

## Сprint 10: Основы React

---

1. Концепция SPA и фреймворки	4. Состояния и эффекты
2. Настройка и сборка проекта	5. Библиотеки компонент. Работа в изоляции. Storybook
3. Основы React	

# Роутинг и авторизация

---

08

[3 недели],  
[0,5 проекта]

Реализуете глобальное состояние, роутинг и авторизацию.

Научитесь ускорять работу React-приложений, проводить их тестирование, выводить их в продакшен и релизить на собственном удалённом сервере.

## Сprint 11: Роутинг и авторизация

---

1. Продвинутый React	4. React Router
2. Глобальное состояние	5. Механизм авторизации
3. Библиотека Redux	

[3 недели],  
[0,5 проекта]

Покроете своё приложение юнит-тестами, а также интеграционными тестами с помощью библиотек Jest и Cypress.

Научитесь ускорять работу React-приложений, проводить их тестирование, выводить их в продакшен и релизить на собственном удалённом сервере.

## Сprint 12: Тестирование React-приложений

1. Оптимизация (ускорение работы приложения)

2. Тестирование React-приложений: Jest, Cypress

3. Подготовка проекта к продакшенну

### Каникулы

[1 неделя]  
Время отдохнуть или доработать проекты

## Проектный месяц

[1 месяц],  
[1 проект]

Работа над крупным проектом. В команде с менеджерами проектов, тимлидами, системными аналитиками, дизайнерами и фулстек-разработчиками подготовите фронтенд приложения.

### Каникулы

[1 неделя]  
Время отдохнуть или доработать проекты

# Основы Node.js, Express и MongoDB

---

10

[3 недели],  
[1 проект]

Разработаете бэкенд для ранее созданного фронтенда интернет-магазина. Реализуете его на TypeScript с использованием Express и хранением данных в MongoDB.

Познакомитесь со средой выполнения серверного кода на JavaScript. Освоите сторонние библиотеки и научитесь добавлять их в проект с помощью менеджера NPM. Научитесь работать с фреймворком Express. Воспользуетесь гибкой производительной noSQL базой данных MongoDB.

## Сprint 13: Основы Node.js, Express и MongoDB

---

1. Введение в бэкенд-разработку	5. Принципы построения API REST
2. Node.js. Экосистема NPM	6. Введение в базы данных. MongoDB
3. Основы Express	7. Работа с БД и авторизация. JWT
4. Обработка пользовательского ввода	

[5 недель],  
[0,5 проекта]

Разработаете бэкенд с модульным API и выделенным слоем данных для одновременной поддержки двух баз данных — MongoDB и PostgreSQL. А ещё вы сможете сами выбрать, с каким фреймворком работать — Express или Nest.js.

Освоите фреймворк Nest.js: основные компоненты, авторизация и логирование, концепции внедрения зависимостей. Познакомитесь с реляционными базами данных и языком SQL. Подключите базу данных PostgreSQL к проекту с использованием библиотеки TypeORM.

## Спринт 14: Разработка бэкенда на Nest.js

---

1. Введение в разработку бэкенда на Nest.js

2. Основы разработки бэкенда на Nest.js

3. Рутинные задачи

4. Passport.js. Авторизация через соцсети

## Спринт 15: Базы данных и PostgreSQL

---

1. Введение в реляционные БД и SQL

2. Продолжение работы с реляционными БД и SQL

3. Связи в реляционных БД

4. PostgreSQL на практике

5. Продвинутая практика с PostgreSQL

6. Моделирование данных

[3 недели],  
[0,5 проекта]

Задеплоите веб-сервис на сервере с использованием Docker-контейнеров.

Научитесь создавать удалённый сервер, запускать на нём приложения и автоматизировать доставку изменений кода. Создадите собственный Docker-контейнер, опубликуете его в хранилище Docker Hub и запустите на вашем сервере. Познакомитесь с Docker Compose, который позволяет управлять группой контейнеров как единым приложением.

## Сprint 16: Тестирование бэкенда, деплой и автоматизация

1. Тестирование бэкенда	4. Введение в мир контейнеров и Docker	6. Хранение и распространение Docker-образов. DockerHub
2. Linux на практике	5. Погружение в мир контейнеров и Docker	7. Запуск приложения из нескольких контейнеров
3. Деплой приложения на сервер		

[3 недели],  
[1 проект]

Поработаете с рефакторингом готового кода, у которого отсутствует валидация данных и есть ошибки безопасности.

Узнаете основные виды угроз для веб-приложений: SQL-инъекции, XSS, CSRF. Освоите инструменты и технологии, которые используются для обеспечения безопасности приложений. Разберёте основы сетевых протоколов HTTPS и SSL/TLS. Поймёте, как безопасно работать с базами данных.

## Сprint 17: Обеспечение безопасности приложения

---

1. Основы безопасности веб-приложений  
2. Межсайтовый скрипting и инъекции

3. Безопасность авторизации и межсайтовая подделка запросов  
4. Защита инфраструктуры

### Каникулы

[1 неделя]  
Время отдохнуть или доработать проекты

## Проектный месяц

---

[2 месяца],  
[1 проект]

Работа над крупным проектом. В команде с менеджерами проектов, тимлидами, системными аналитиками, дизайнерами и фулстек-разработчиками подготовите бэкенд приложения.

### Каникулы

[1 неделя]  
Время отдохнуть или доработать проекты

## Выпуск

---



[60 часов]

Алгоритмы и структуры данных

14

Вы научитесь решать популярные алгоритмические задачи различной сложности. Научитесь анализировать сложность алгоритмов и подробнее разберёте функции.

[60 часов]

Софт-скилы для разработчиков

15

«Мягкие» навыки важны также, как и «жёсткие». Поэтому параллельно с основной программой вы будете проходить софтовый трек — «курс в курсе», где вы освоите инструменты, которые помогут вам в учёбе и в будущей работе. Вы узнаете, как организовать учебный процесс, адаптироваться в новой среде и организовать командную работу. Будет и теория, и учебные задания, которые помогут вам сделать и обучение хардовым навыкам, и пребывание на будущей работе легче и осознаннее.

## Темы

1. Рефлексия

3. Планирование и реализация

5. Стрессоустойчивость

2. Целеполагание

4. Работа в команде

6. Получение  
и предоставление обратной  
связи

# Параллельные модули: Программа трудоустройства



[60 часов]

## Программа трудоустройства

16

Карьерный трек состоит из двух этапов: программы трудоустройства и программы акселерации.

Здесь мы поможем вам собрать все необходимые для трудоустройства артефакты и приобрести нужные навыки. Вы составите резюме, соберёте портфолио, напишете мотивационное письмо. С нашей поддержкой разработаете стратегию поиска и подготовитесь к интервью.

## Программа акселерации

17

Это работа в поле. Вы будете подаваться на вакансии, писать сопроводительные письма, делать тестовые и ходить на собеседования. Скорее всего, вы получите много отказов и мы вместе будем анализировать ошибки, чтобы улучшить результаты. Этот этап заканчивается, когда вы получаете предложение о работе, на которое согласны. В среднем такой поиск занимает от одного до трёх месяцев.

## Темы

- 1. Рынок труда
- 2. Работа в продуктовой команде
- 3. Возможности развития в веб-разработке
- 4. Целеполагание и резюме

- 5. Мотивационное письмо и портфолио
- 6. Стратегии поиска работы
- 7. Адаптация артефактов
- 8. Реалии рынка труда, поиск работы в столице и в регионах

- 9. Собеседования
- 10. Подготовка к интервью
- 11. Программа активной акселерации. От отклика до приглашения на работу