

Web Development

Web development is a wide-ranging area of software development, primarily developing websites and web applications. There are numerous roles in web development across companies of all sizes. This field moves quickly, with new tools and frameworks released weekly. As a result, there is no "right" way to do web development. On YETI, we seek to emulate industry standards. Our stack primarily utilizes TypeScript, Next.js, and PostgreSQL.

This book aims to provide resources for learning our stack and more general skills in web development. If you pursue a more web development-oriented path of software development, these skills will be career-relevant and give you a leg up in your early career.

- [UI/UX Basics](#)
- [UI Design](#)
- [Programming for Web Development](#)

UI/UX Basics

What is UI/UX?

These are perhaps two of the most fundamental concepts in developing for the web, and can make or break a site/application.

- UI - User Interface
- UX - User Experience

Difference Between UI and UX

User Interface	User Experience
Colors & Fonts	Target audience & users
Interactivity	Flow of site/app
Layout	

[Here's an article from Figma that explains the differences.](#)

Fundamentals of UI/UX

The following is a great crash course into the world of UI/UX, and how to get started with learning design.

<https://www.youtube.com/embed/wluVvCuijhU?si=nzoR9SPwX6jBkE8G>

Materials and Resources

[Figma](#) is a great, industry-standard resource with a free tier for learning how to design, and creating designs.

- Figma also provides [terrific articles](#) that are a great primer into designing for web.

UI Design

Resources

General resources for learning good UI design.

Figma

Figma provides a terrific set of resources for learning the basics of UI design. Some articles to pay particular attention to:

- Visual Hierarchy
- 5 Key Principles
- Simplicity
- Consistency
- Constraints

Programming for Web Development

Overview

This guide primarily focuses on the skills necessary to work with YETI's software stack (TypeScript, Next.js/React, and PostgreSQL). With that being said, there are plenty of other commonly used frameworks. Below is a table containing some of the more popular choices:

JavaScript/Type Script	PHP	Java	Python	Ruby	C#
Next.js	Laravel	SpringBoot	Flask	Ruby on Rails	.NET
Remix			Django		
Astro					
TanStack Start					
Angular					
Gatsby					
SvelteKit					

JavaScript/TypeScript frameworks are currently very commonplace. One of the key driving factors of this is the choice of language. Browsers can only run JavaScript (and more recently, Web Assembly). JS/TS frameworks allow you to write your frontend and backend code entirely in one language. This allows these sites to get off the ground much faster, as developers need to spend less time learning new languages to work with the code. As a result, these frameworks are becoming increasingly common in industry (particularly Next.js and Remix). We

We chose Next.js due to our mentors' familiarity with the framework. Multiple YETI mentors use this framework for work or personal websites. We can provide substantial help with learning the framework and debugging issues that arise. Next.js is also a common framework to see in the industry, and several of the world's most recognizable sites (Nike, OpenAI, Claude, TikTok, Notion, Hulu...) currently use it (<https://nextjs.org/showcase>).

L1 Standards

Learning HTML/CSS

HTML and CSS are the building blocks for the web. HTML controls the general layout/blocks of the page, while CSS controls the appearance and how those blocks are displayed. These aren't programming languages in the traditional sense. HTML (or Hypertext Markup Language) is a "markup language," which essentially means it encodes the structure of the text/content it contains. CSS (or Cascading Style Sheets) is a "stylesheet" language. Stylesheets are files documenting how the page should be displayed. CSS is a way of representing these styles in a text-based format.

Codecademy has great (free) resources for learning both what HTML and CSS do. To get started, check out the following courses:

- HTML Fundamentals: <https://www.codecademy.com/courses/learn-html-fundamentals>
- Learn CSS: <https://www.codecademy.com/courses/learn-css>

After completing these two courses, congrats! You'll be able to build a static website.

Learning JavaScript

JavaScript is a language that lets you dynamically update content on websites. This is useful for forms and other interactive parts of websites. Together with HTML and CSS, it creates the traditional web development stack.

The course on Codecademy is great for getting started.

- Introduction to JavaScript: <https://www.codecademy.com/enrolled/courses/introduction-to-javascript>

After completing this, congrats! You'll be ready for the L1 web development test.