Course Objectives

This course will give you an understanding of how the Web is put together. It will introduce the key technologies, help you see how they work in tandem, and inform you of some of the key things that can (and often do) go wrong in the course of development of a traditional Web page, a Web app, or an IoT device. After taking this course you'll be in a better position to manage a team of developers working with Web technologies or be better able to recruit them, and will be able to use the basic topic terminology correctly.

Course Content

It's usually a bad idea to bet against the Web. Over the past few decades it has completely reshaped the way we do everything from shopping to learning to traveling, and now with the Internet of Things it's in the process now of shaping the way all of our cars, homes, and appliances interoperate. This course is a just-technical-enough approach to provide you with the information you need to work with Web developers using a meaningful shared vocabulary. This course provides a meaningful overview without wasting time on irrelevant details.

Student Background

Before attending this course you should be a comfortable user of the Web. You don't need to know how to make anything, but filling out a Web form shouldn't be beyond your skill level.

Computer Requirements

The Web can be accessed on most hardware in use today. For the purposes of this class we'd recommend you use a reasonably new laptop (running something like Mac OS X 10.6+, Microsoft Windows 7+, or any recent version of Linux), a Chromebook, or a reasonably current iPad or Android Tablet.

Suggested Reading

The Mozilla Developer Network has made the best Web technology reference available online at: https://developer.mozilla.org/en-US/docs/Web/ but do note that it goes into far more depth than this course does. After taking this course the documents in this extensive collection will be far more approachable to you.

Class Schedule

This class will run for two eight-hour days. Each day will be broken up with a break in the morning, afternoon, and at lunch time. Work periods will consist of a lecture on a topic followed by exercises to reinforce it.

Day 1: The Big Picture -- How It All Fits Together

Servers and clients. URLs. HTML (and HTML5). CSS. JavaScript. SQL. MathML. Python. Node. PHP. Java. RDF. Images and other multimedia. SVG. Canvas. WebGL. JSON. XML. Configuration files. RPC (SOAP, XML-RPC, JSON-RPC, etc.). ReST. WebSockets. WebRTC. Web beacons. Brief overview of some popular servers and database systems.

Day 2: Things That Go Wrong -- Protecting Your Project

HTTP vs. HTTPS. Authentication and authorization. Spiders and robots. Security. Cookies. Sessions. Compression. Minimization. Media queries. Mobile vs. desktop. Native apps vs. Web apps. SEO. Accessibility. CDN. Browser differences. Version control. Testability. Production environments versus staging environments versus development environments. Reproducibility. VMs and containers.