Syllabus

Web Development Fundamentals: From Front-End to Back-End

Course Overview

This course is designed to take you from zero to a capable, full-stack web developer. You will learn why web development is a critical and rewarding skill, master the core technologies of the front-end (what users see), and dive into the back-end (the server logic) that powers modern web applications. By the end, you will have the skills to build, style, and add interactivity to your own websites and connect them to a database.

Course Modules

Module 1: Why Learn Web Development?

Duration: 1 Week
Objective: To understand the motivation, opportunities, and landscape of web development.

- **The Digital Economy:** Why every business needs a web presence.
- **Career Opportunities:** Front-End, Back-End, and Full-Stack Developer roles. Freelancing vs. In-house.
- **The Power of Creation:** How coding turns ideas into reality.
- **The Toolbox of a Developer:** A high-level overview of the technologies we will cover (HTML, CSS, JS, PHP, SQL).
- **Setting Up for Success:** Installing necessary software and joining developer communities.

Module 2: Software Fundamentals & The Core Trio

Duration: 5 Weeks
Objective: To build a solid foundation in the three essential front-end technologies.

Part 2.1: HTML (HyperText Markup Language) - The Skeleton

- **What is HTML?** Structuring web content.
- **Essential Tags:** <html>, <head>, <body>, <title>

- **Text & Media:** Headings (<h1>), Paragraphs (<p>), Lists (<ul>, <ol>), Images (<img>), Links (<a>).
- **Organization:** <div>, <span>, and Semantic HTML5 tags (<header>, <nav>, <section>, <footer>).
- **User Input:** Forms (<form>), Inputs (<input>, <textarea>), Buttons (<button>).

Part 2.2: CSS (Cascading Style Sheets) - The Styling

- **What is CSS?** Adding visual design to the HTML structure.
- **Selectors:** How to target HTML elements (tag, class, ID).
- **The Box Model:** Margin, Border, Padding, and Content.
- **Layout Techniques:**
    - **Flexbox:** For one-dimensional layouts.
    - **CSS Grid:** For two-dimensional layouts.
- **Styling:** Colors, fonts, backgrounds, and basic animations.
- **Responsive Design:** Making websites look good on all devices using media queries.

Part 2.3: JavaScript - The Interactivity

- **What is JavaScript?** Making web pages dynamic and interactive.
- **Core Syntax:** Variables (let, const), Data Types (Strings, Numbers, Booleans, Arrays, Objects).
- **Operators & Control Flow:** Conditional statements (if/else), Loops (for, while).
- **Functions:** Defining and calling functions.
- **The Document Object Model (DOM):** How to manipulate HTML and CSS with JavaScript.
- **Events:** Responding to user actions (clicks, form submissions, key presses).

Module 3: Development Tools & Environment

Duration: 1 Week
Objective: To become proficient with the tools professional developers use daily.

- **VS Code (Visual Studio Code):**
    - Installation and setup.

    - Essential extensions (Live Server, Prettier, ESLint).
    - Keyboard shortcuts and efficient workflow.
- **Browser Developer Tools:** Inspecting elements, debugging JavaScript, and analyzing network activity.
- **Version Control with Git:** Basics of git init, add, commit, and connecting to a remote repository on GitHub.

Module 4: Introduction to Other Languages (Ruby & SQL)

Duration: 2 Weeks
Objective: To understand the role of other key programming languages in web development.

Part 4.1: SQL (Structured Query Language) - Talking to Databases

- **What is a Database?** Why websites need to store data.
- **Relational Databases:** Understanding tables, rows, and columns.
- **Core SQL Commands:**
    - CREATE TABLE to structure data.
    - INSERT to add data.
    - SELECT to retrieve data (using WHERE, ORDER BY, JOIN).
    - UPDATE and DELETE to modify data.

Part 4.2: Ruby - A Glimpse into Another Language

- **Why Ruby?** Introduction to its philosophy and use in web development (especially the Ruby on Rails framework).
- **Basic Syntax:** Variables, methods, and loops in Ruby.
- **Comparison with PHP & JavaScript:** Understanding that different languages can solve similar problems.

Module 5: Server-Side Development with PHP

Duration: 4 Weeks
Objective: To learn how to build the server-side logic of a web application, connecting the front-end to a database.

- **What is Server-Side Development?** Client-Server model explained.
- **Introduction to PHP:** How PHP code is executed on the server.
- **PHP Syntax:** Variables, arrays, control structures, and functions.
- **Working with Forms:** Handling form data sent via POST and GET methods.
- **Connecting to a Database:** Using PHP to connect to a MySQL database with PDO or MySQLi.
- **Dynamic Web Pages:** Using PHP to pull data from the database and inject it into your HTML.
- **Sessions & Cookies:** Managing user state and login systems.
- **Building a Simple CRUD Application:** Create, Read, Update, and Delete entries from a database (e.g., a simple blog or contact list).

Final Project

Duration: 2-3 Weeks
Objective: To apply all learned concepts by building a full-stack web application from scratch.

- **Project Proposal:** Plan a dynamic website (e.g., a personal portfolio with a blog CMS, a simple event registration site, a product catalog).
- **Execution:**
  - Build the front-end with HTML, CSS, and JavaScript.
  - Implement the back-end logic with PHP.
  - Use a MySQL database to store and retrieve data.
  - Use Git for version control and host the final code on GitHub.
- **Presentation & Code Review.**

Assessment & Grading

- **Weekly Coding Assignments:** 30%
- **Module Projects (e.g., a styled static site, a JS interactive app, a PHP/MySQL app):** 40%
- **Final Project & Presentation:** 30%

Required Software & Tools

- **Visual Studio Code (Free)**
- **XAMPP / MAMP (Free)** - A local server environment for running PHP and MySQL.
- **Git (Free)**
- **A Modern Web Browser (Chrome, Firefox, Edge)**

This syllabus provides a logical progression, ensuring students understand the "why" before diving into the "how," and builds up to the powerful combination of PHP and SQL for creating data-driven websites.