What is web development?

Web development refers to the creating, building, and maintaining of websites. It includes aspects such as web design, web publishing, web programming, and database management. It is the creation of an application that works over the internet i.e. websites.

The word Web Development is made up of two words, that is:

Web: It refers to websites, web pages or anything that works over the internet.

Development: It refers to building the application from scratch.

Web Development can be classified into two ways:

- Frontend Development
- Backend Development

• Frontend Development

- The part of a website where the user interacts directly is termed as front end. It is also referred to as the 'client side' of the application.

Popular Frontend Technologies

HTML: HTML stands for HyperText Markup Language. It is used to design the front end portion of web pages using markup language. It acts as a skeleton for a website since it is used to make the structure of a website.

CSS: Cascading Style Sheets fondly referred to as CSS is a simply designed language intended to simplify the process of making web pages presentable. It is used to style our website.

JavaScript: JavaScript is a scripting language used to provide a dynamic behavior to our website.

Bootstrap: Bootstrap is a free and open-source tool collection for creating responsive websites and web applications. It is the most popular CSS framework for developing responsive, mobile-first websites. Nowadays, the websites are perfect for all browsers (IE, Firefox, and Chrome) and for all sizes of screens (Desktop, Tablets, Phablets, and Phones).

Frontend Libraries and Frameworks

HTML

CSS

- **CSS Frameworks**
    - Bootstrap
    - Tailwind CSS
    - Bulma
    - Foundation
    - Primer CSS

        - Spectre CSS
        - Materialize CSS
        - Onsen UI
        - Semantic UI
        - Blaze UI
        - Pure CSS
    - **CSS Preprocessors**
        - SASS
        - LESS

JavaScript

- **JavaScript Technology**
    - ES6
    - TypeScript
- **JavaScript Frameworks**
    - AngularJS
        - Angular ngx Bootstrap
        - Angular PrimeNG
    - VueJS
        - NuxtJS
    - script.aculo.us
    - Ember.js
    - Handlebar.js
    - Backbone.js
- **JavaScript Libraries**
    - jQuery
        - jQuery UI
        - jQuery Mobile
        - jQWidgets
        - jQuery EasyUI
    - ReactJS

Backend Development

Backend is the server side of a website. It is part of the website that users cannot see and interact with. It is the portion of software that does not come in direct contact with the users. It is used to store and arrange data.

Popular Backend Technologies

- **PHP:** PHP is a server-side scripting language designed specifically for web development.
- **Java:** Java is one of the most popular and widely used programming languages. It is highly scalable.
- **Python:** Python is a programming language that lets you work quickly and integrate systems more efficiently.
- **Node.js:** Node.js is an open source and cross-platform runtime environment for executing JavaScript code outside a browser.

Back End Frameworks and Technology:

PHP

- **Framework:** Laravel
- **CMS:** WordPress

NodeJS

- **Framework:** Express

Python

- **Framework:** Django
- **Package Manager:** Python PIP

Ruby

- **Framework:** Ruby on Rails

Java

- **Framework:** Spring, Hibernate

C#

- **Framework:** .NET

Database:

- **Relation Database**
    - Postgre SQL
    - MariaDB
    - MySQL
- **NoSql Database**

- MongoDB

Frontend vs Backend:

Aspect	Front-End Development	Back-End Development
Focus	User interface and user experience	Server-side functionality and data
Technologies	HTML, CSS, JavaScript, UI frameworks	Server-side languages (e.g., Python, Java)
User Interaction	Responsible for what users see and interact with on a website or app	Responsible for data processing, storage, and server operations
Presentation	Design, layout, and visual elements	Data processing, logic, and algorithms
Communication	Handles client-side interactions with users	Handles server-side interactions
Responsiveness	Ensures the UI is responsive and accessible	Ensures efficient server performance
Examples of Tools & Frameworks	React, Angular, Vue.js	Node.js, Django, Ruby on Rails, Express
UI/UX Design	Focuses on the look and feel of the application	Focuses on the functionality and logic
Client-Side Rendering	Renders content on the user's device	Returns processed content to the client
Browser Compatibility	Ensures compatibility across different browsers	Irrelevant, as it operates on the server
Security	Client-side security measures	Server-side security measures
Real-time Interaction	Used for real-time user interactions	Used for real-time data processing

The distinctions among PHP, Java, and Python are as follows:

Facets	PHP	Java	Python
Type	Scripting language	Compiled language	Scripting language
Paradigm	Imperative, Object-Oriented	Object-Oriented	Multi-paradigm (OOP, procedural, functional)
Syntax	C-like syntax with HTML integration	C-like syntax with strong typing	Readable syntax with dynamic typing
Community	Large and active community	Large and active community	Large and active community
Web	Strong for web development (backend)	Used in backend, less for frontend	Used in web development (backend), not common for frontend
Desktop	Limited for desktop applications	Common for cross-platform desktop apps	Less common, but used for GUI apps
Mobile	Limited for mobile development	Android app development (Java/Kotlin)	Limited (Kivy, BeeWare for cross-platform)
Performance	Moderate	High	Moderate to high

Ease of Learning	Relatively easy	Moderate learning curve	Relatively easy
Ecosystem	LAMP/WAMP stack, WordPress, Laravel	Enterprise applications, Android	Scientific computing (NumPy, SciPy), web frameworks (Django, Flask)
Libraries	Rich ecosystem of web-related libraries	Extensive libraries and frameworks	Comprehensive standard library
Concurrency	Limited concurrency support	Robust concurrency with threads	Support for multi-threading, async programming
Portability	Cross-platform	Write once, run anywhere (JVM)	Cross-platform
Use Cases	Web development, small to medium projects	Enterprise applications, large-scale systems	Web development, scripting, automation, data analysis
Popularity	Widely used in web development	Widely used in enterprise	Widely used in scripting, web, and data tasks

The differences between React, Angular, and Vue.js are outlined below:

Characteristics	React	Angular	Vue.js
Type	JavaScript library	JavaScript framework	JavaScript framework/library
Popularity	Highly popular and widely used	Popular for enterprise applications	Gaining popularity
Maintained By	Facebook	Google	Independent open-source community
Architecture	Component-based	Component-based	Component-based
Learning Curve	Moderate	Steeper learning curve	Relatively gentle learning curve
Community	Large and active community	Large and active community	Active community
Flexibility	More flexible, requires more decisions	Opinionated framework	Balanced approach
Performance	High performance, virtual DOM	Moderately performant	Good performance, virtual DOM
Rendering	Client-side rendering	Client-side rendering	Client-side rendering
State Management	Can use Context API, Redux, MobX	Built-in state management (RxJS)	Vuex for state management
Directives	No built-in directives	Rich set of directives	Limited directives
CLI Tools	Create React App	Angular CLI	Vue CLI
Integration	Easily integrates with other libraries	Fully integrated with its ecosystem	Can be integrated into projects
Data Binding	One-way data binding	Two-way data binding	Two-way data binding (via v-model)
Size	Smaller footprint	Larger bundle size	Moderate bundle size
Mobile Support	React Native for mobile apps	Ionic framework for mobile apps	Quasar, Framework7 for mobile apps

Server-side Rendering	Supported through Next.js	Supported through Angular Universal	Supported through Nuxt.js
Use Cases	Web and mobile app development	Enterprise applications, large projects	Web and small to medium app development
License	MIT License	MIT License	MIT License