**CHAPTER 2: TOOLS AND TECHNOLOGY**

* 1. **Introduction to HTML and CSS**

HTML (Hypertext Markup Language) and CSS (Cascading Style Sheets) are integral technologies used in web development to create and style web pages. These technologies complement each other, with HTML providing the structure and content, and CSS managing the design and presentation. This section explores the fundamental roles of HTML and CSS in web development.

* **Structuring Web Pages with HTML:**

HTML is the cornerstone of web content, defining the layout and organization of web pages. It utilizes a series of elements, or tags, to delineate content such as text, images, links, and multimedia. Here are some key functions of HTML:

* **Content Structuring:** HTML elements like headings (<h1> to <h6>), paragraphs (<p>), and lists (<ul>, <ol>) create a clear and accessible structure for web pages.
* **Media Integration:** Tags like <img> and <video> embed images and videos, enriching the visual appeal of the content.
* **Hyperlinking:** The <a> tag creates links, facilitating navigation between different pages or sections within a website.
* **Enhancing Web Pages with CSS:**

CSS is the language used to style HTML documents, enabling the separation of content from design. It enhances the aesthetic appeal and user experience of web pages. Key aspects of CSS include:

* **Styling Text:** CSS controls font styles, sizes, colors, and spacing, ensuring text is visually engaging and readable.
* **Layout Design:** Techniques like flexbox and grid layouts help in creating responsive and adaptive web designs that work seamlessly across various devices.
* **Visual Effects:** CSS can implement animations, transitions, and transformations, adding dynamic and interactive elements to web pages.
* **Interactive Design with HTML and CSS:**

Together, HTML and CSS create interactive and visually appealing web pages. This synergy allows developers to:

* **Design Navigation Menus:** CSS styles HTML structures to create intuitive and visually consistent navigation menus.
* **Build Responsive Pages:** Media queries in CSS adapt the layout and design based on the device's screen size, ensuring a consistent user experience across different devices.
* **Implement Design Themes:** CSS can apply consistent design themes across multiple HTML pages, promoting a cohesive brand identity.
* **Applications of HTML and CSS:**

HTML and CSS are not limited to basic web pages but extend to various advanced applications, including:

* **Creating Interactive Forms:** HTML forms (<form>, <input>, <textarea>) combined with CSS styles enhance user interaction and data collection.
* **Building Complex Web Applications:** Modern web applications often utilize HTML and CSS in conjunction with JavaScript and other frameworks to create sophisticated and interactive user interfaces.
* **Designing Mobile-Responsive Websites:** Ensuring websites are accessible and functional on mobile devices is achieved through responsive design techniques using HTML and CSS.
  1. **Introduction to JavaScript**

JavaScript is mainly used for web-based applications and web browsers. But JavaScript is also used beyond the Web in software, servers and embedded hardware controls. Here are some basic things JavaScript is used for:

* + - Adding interactive behavior to web pages

JavaScript allows users to interact with web pages. There are almost no limits to the things you can do with JavaScript on a web page – these are just a few examples:

* + - * Show or hide more information with the click of a button
      * Change the color of a button when the mouse hovers over it
      * Slide through a carousel of images on the homepage
      * Zooming in or zooming out on an image
      * Displaying a timer or count-down on a website
      * Playing audio and video in a web page
      * Displaying animations
      * Using a drop-down hamburger menu
    - Creating web and mobile apps

Popular JavaScript front-end frameworks include React, React Native, Angular, and Vue. Many companies use Node.js, a JavaScript runtime environment built on Google Chrome’s JavaScript V8 engine. A few famous examples include PayPal, LinkedIn, Netflix, and Uber!

* + - Building web servers and developing server applications

Beyond websites and apps, developers can also use JavaScript to build simple web servers and develop the back-end infrastructure using Node.js.

* 1. **Introduction to Node.js**
* What is Node.js?

Node.js is a server-side platform built on Google Chrome's JavaScript Engine (V8 Engine). Node.js was developed by Ryan Dahl in 2009 and its latest version is v0.10.36.

Node.js is a platform built on Chrome's JavaScript runtime for easily building fast and scalable network applications. Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient, perfect for data-intensive real-time applications that run across distributed devices.

Node.js is an open source, cross-platform runtime environment for developing server-side and networking applications. Node.js applications are written in JavaScript, and can be run within the Node.js runtime on OS X, Microsoft Windows, and Linux.

Node.js also provides a rich library of various JavaScript modules which simplifies the development of web applications using Node.js to a great extent.

Node.js = Runtime Environment + JavaScript Library

* + Features of Node.js

Following are some of the important features that make Node.js the first choice of software architects.

Asynchronous and Event Driven: All APIs of Node.js library are asynchronous, that is, non- blocking. It essentially means a Node.js based server never waits for an API to return data. The server moves to the next API after calling it and a notification mechanism of Events of Node.js helps the server to get a response from the previous API call.

Very Fast: Being built on Google Chrome's V8 JavaScript Engine, Node.js library is very fast in code execution.

Single Threaded but Highly Scalable: Node.js uses a single threaded model with event looping. Event mechanism helps the server to respond in a non-blocking way and makes the server highly scalable as opposed to traditional servers which create limited threads to handle requests. Node.js uses a single threaded program and the same program can provide service to a much larger number of requests than traditional servers like Apache HTTP Server.

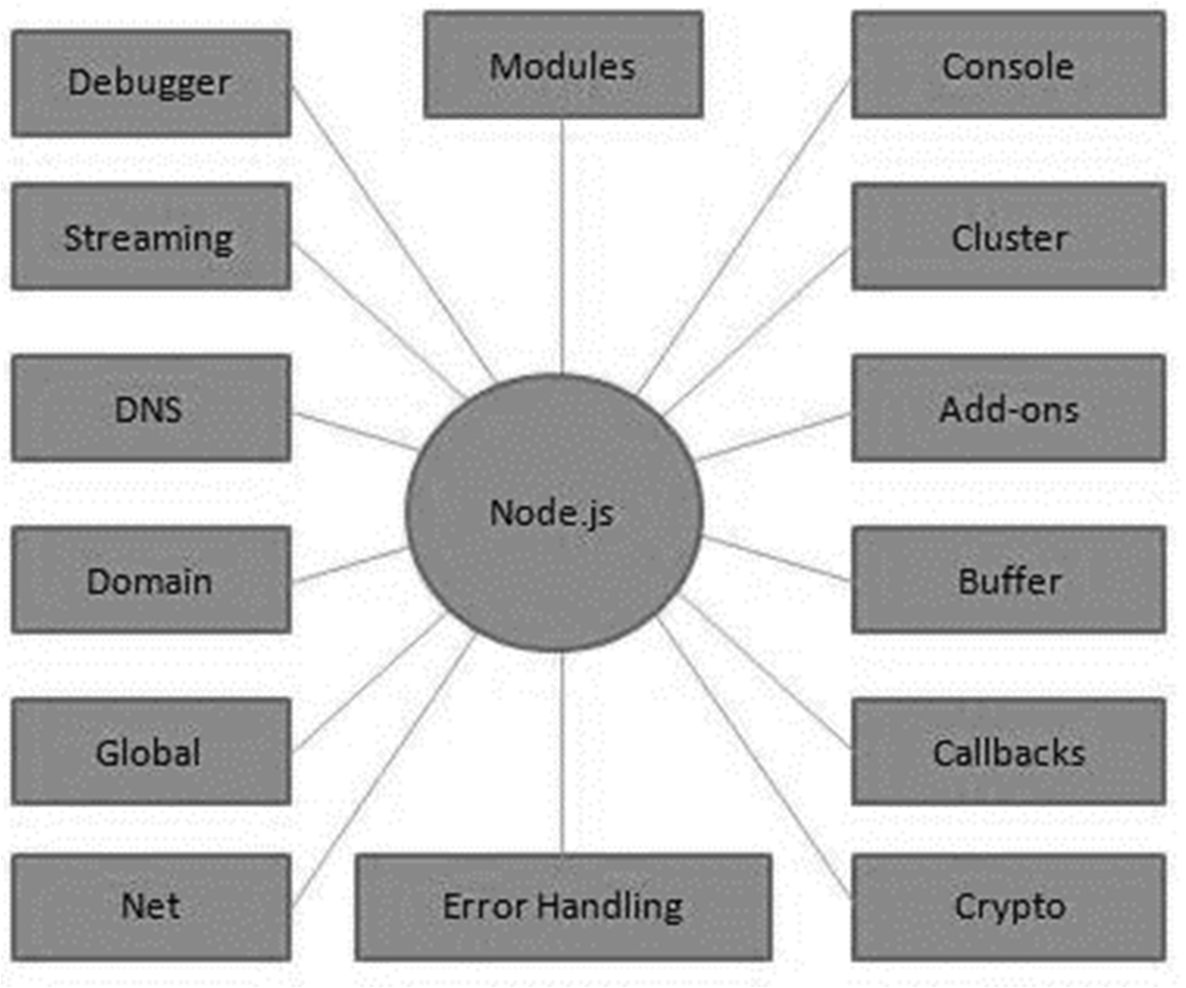
No Buffering: Node.js applications never buffer any data. These applications simply output the data in chunks.

License: Node.js is released under the MIT license

* + Who Uses Node.js?

eBay, General Electric, GoDaddy, Microsoft, PayPal, Uber, Wikipins, Yahoo!, and Yammer to name a few, and Projects, Applications, and Companies Using Node.js

* + Concepts

The following diagram depicts some important parts of Node.js

* + Where to Use Node.js?

Following are the areas where Node.js is proving itself as a perfect technology partner.

* + - I/O bound Applications
    - Data Streaming Applications
    - Data Intensive Real-time Applications (DIRT)
    - JSON APIs based Applications
    - Single Page Applications
  + Where Not to Use Node.js?

It is not advisable to use Node.js for CPU intensive applications

* 1. **Introduction to Express.js**
* What is Express.js?

Express is a fast, assertive, essential and moderate web framework of Node.js. You can assume express as a layer built on the top of the Node.js that helps manage a server and routes. It provides a robust set of features to develop web and mobile applications.

Some of the core features of Express framework:

* + It can be used to design single-page, multi-page and hybrid web applications.
  + It allows to setup middlewares to respond to HTTP Requests.
  + It defines a routing table which is used to perform different actions based on HTTP method and URL.
  + It allows to dynamically render HTML Pages based on passing arguments to templates.
* Why use Express.js?

For following purpose we use Express.js:

* + Ultra-fast I/O
  + Asynchronous and single threaded
  + MVC like structure
  + Robust API makes routing easy
  1. **Introduction to MongoDB**

MongoDB is an open-source document-oriented database that is designed to store a large scale of data and allows you to work with that data very efficiently. It is categorized under the NoSQL (Not only SQL) database because the storage and retrieval of data in the MongoDB are not in the form of tables.

* What is the main features of MongoDB?
* A Schema-less database means one collection can hold different types of documents in it
  + - In MongoDB, all the data stored in the documents instead of tables like in RDBMS
    - In MongoDB database, every field in the documents is indexed with primary and secondary indices this makes easier and takes less time to get or search data from the pool of the data
    - MongoDB provides horizontal scalability with the help of sharing.
    - It allows to perform operations on the grouped data and get a single result or computed result
    - The performance of MongoDB is very high and data persistence as compared to another database due to its features like scalability, indexing, replication, etc.
* Who use MongoDB?
  + many high-profile businesses and organizations such as Forbes, Facebook, Google, IBM, Twitter, and many more.
  1. **Introduction to React**

React is a JavaScript-based UI development library. Facebook and an open-source developer community run it. Although React is a library rather than a language, it is widely used in web development. The library first appeared in May 2013 and is now one of the most commonly used frontend libraries for web development

* What is the main features of React?
  + - Everything in React is treated as a **component**. This means that the development of a web interface or application in React is made up of several components.
    - Developers use it to create React components, making the syntax similar to HTML. **JSX** is one of the best features of React as it makes it extremely simple for developers to write the building blocks.
    - React uses a unidirectional data flow, forcing developers to edit components via the callback feature rather than directly editing them.
    - The **declarative user interface**is the ability to create a dynamic and interactive user interface for websites and mobile applications.
* Who use React?
  + React is used by Facebook, Netflix, Yahoo, Codecademy, Dropbox, Airbnb, Asana, Microsoft, Slack, and many more.
  1. **Software Requirement**

The software requirement specification can produce at the culmination of the analysis task. The function and performance allocated to software as part of system engineering are refined by establishing a complete information description, a detailed functional description, a representation of system behavior, an indication of performance and design constrain, appropriate validation criteria, and other information pertinent to requirements.

* Operating System: Windows
* Coding Language: HTML, CSS, JavaScript, React, Node.js & Express.js
* Database: MongoDB
  1. **Hardware Requirement**
* Processor: RYZEN 5
* Memory: 16GB RAM
* Hard Disk: 512GB