Table of Contents

HTML5

<u>Introduction to HTML5</u>

HTML5 is the fifth and latest major version of the HyperText Markup Language (HTML), which is the standard language used for creating and structuring web pages and web applications. HTML5 is designed to be the foundation of the web, providing a comprehensive and robust framework that supports the latest multimedia, interactive elements, and complex applications, all while ensuring compatibility across a wide range of devices and browsers.

Fig.1 **Symbol of HTML5.**

Key Features of HTML5

Key features in HTML5 a powerful and flexible tool for modern web development, enabling developers to create more dynamic, interactive, and user-friendly web applications. The key features are listed below:

- New Semantic Elements: <article>, <section>, <nav>, etc.

- Multimedia Support: Native audio (<audio>) and video (<video>) elements.

- Graphics and Interactive Content: SVG and Canvas.

- Enhanced Form Controls: Date pickers, sliders, and other input types.

- APIs and DOM Enhancements: Geolocation API, Web Storage, and more.

Benefits of HTML5

The benefits of HTML5 are listed below:

- Cross-Platform Compatibility: Works on various devices and screen sizes.

- Improved User Experience: Rich multimedia content and better interactivity.

- Better Performance: Reduced need for third-party plugins.

- Enhanced Accessibility: Improved semantic tags for better accessibility.

- SEO Advantages: More meaningful HTML structure helps search engines.

Applications and Examples of HTML5

Applications:

Semantic Structure

- Semantic Elements: HTML5 introduces semantic elements like <header>, <footer>, <article>, <section>, <nav>, and <aside>. These elements provide a clearer structure to web pages, making them more accessible and easier to understand for both developers and users.

Multimedia

- Audio and Video: HTML5 includes built-in support for audio and video playback using the <audio> and <video> elements. This allows developers to embed media directly into web pages without the need for plugins like Flash.

Graphics and Animation

- Canvas: HTML5 introduces the <canvas> element, which allows for dynamic, scriptable rendering of 2D shapes and bitmap images. It's widely used for creating animations, games, and data visualization.

- SVG (Scalable Vector Graphics): HTML5 includes support for SVG, which allows developers to create vector-based graphics that scale perfectly to different screen sizes and resolutions.

Form Enhancements

- Form Controls: HTML5 introduces new form input types (<input type="date">, <input type="email">, etc.) and attributes (required, pattern, placeholder, etc.) that improve usability and validation.

- Validation: HTML5 adds built-in form validation through attributes like required, pattern, and more, reducing the need for JavaScript-based validation.

Example of HTML5

<header>: It is used to define introductory content, typically containing navigation links or headings.

Code:

```
<header>
    <h1>Welcome to My Website</h1>
    <nav>
        <ul>
            <li><a href="#">Home</a></li>
            <li><a href="#">About</a></li>
            <li><a href="#">Services</a></li>
            <li><a href="#">Contact</a></li>
        </ul>
    </nav>
</header>
```
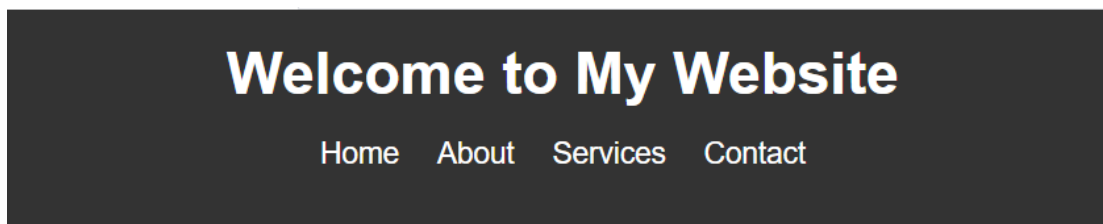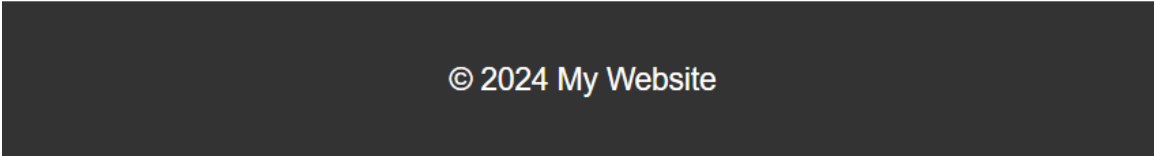
Output:

Fig.2 **Example of <header> tag.**

<footer>: Footer is used to introduced to add copyright or right reserver by company in the bottom of <body> or <html> tag.

Code:

```
<footer>
    <p>&copy; 2024 My Website</p>
</footer>
```

Output:

Fig.3 **Example of <footer> tag**.

COCOMO II Model

Introduction of COCOMO II Model

COCOMO II stands for Constructive Cost Model II. Developed by Barry Boehm and others at the University of Southern California, it is a model designed for estimating software project costs, effort, and schedule. The model builds upon the original COCOMO, introducing additional features and refinements to enhance accuracy and applicability.

Key Components of COCOMO II

The key components of COCOMO II are given below:

Sizing Models:

- Early Design Model (Basic COCOMO): Used in the early stages of development when the specifics of the system aren't yet defined. Estimates are based on the size of the software.

- Post-Architecture Model (Intermediate and Detailed COCOMO): Used when more detailed information about the system is known. Includes more factors that affect development effort and cost.

Cost Drivers:

- Factors that affect the overall cost of the project, effort required, and schedule. These factors can be categorized into:

  - Product Attributes: e.g., required software reliability, database size, product complexity.

  - Hardware Attributes: e.g., execution time constraints, main storage constraints.

  - Personnel Attributes: e.g., analyst capability, applications experience, programming language experience.

  - Project Attributes: e.g., use of modern programming practices, development flexibility.

Scale Factors:

- Exponents that reflect the degree of influence of cost drivers on effort. These are used in the effort formula and can range from "Very Low" to "Extra High" influence.

Basic COCOMO II Equation

$E = ab \times (KLOC)c$

Where:

- **E** is the effort in person-months.

- **KLOC** is the estimated number of thousands of lines of code in the project (in Kilo Lines of Code).

- **a**, **b**, and **c** are parameters derived from historical project data and are defined as follows:

    - **a** is a scale factor reflecting the product attributes.

    - **b** is a scale factor reflecting the project attributes.

    - **c** is an exponent derived from the sum of the scale factors (a and b).

Example of COCOMO II in Use

Developing a Web Application

Project Definition and Scope:

- You are tasked with developing a new web application for an e-commerce platform.

- The application will include user registration, product catalog, shopping cart, and payment processing functionalities.

Step 1: Estimate Size in Kilo Lines of Code (KLOC):

- Based on initial requirements and design, the estimated size of the application is 50 KLOC.

Step 2: Determine the Scale Factors (SF):

- Identify and assess the cost drivers (product, hardware, personnel, and project attributes).

- Assign a rating to each factor based on its impact (Very Low, Low, Nominal, High, Very High, Extra High).

- Calculate the overall scale factors and their total effect on the project.

Step 3: Calculate Effort (E):

- Use the COCOMO II Basic Equation to calculate the effort (E) required:

$E = ab \times (KLOC)c E = ab \times (KLOC)c$

- **a** = 2.94 * (50)^0.91 = 34.19

- **b** = 0.91 * (EAF) (where EAF= estimated adjustment

Conclusion:

HTML5

HTML5 is the latest version of the Hypertext Markup Language (HTML) standard, developed by the World Wide Web Consortium (W3C) and the Web Hypertext Application Technology Working Group (WHATWG). It introduces new semantic elements, multimedia support, graphics capabilities, and enhanced form controls, among other features.

COCOMO II Model

COCOMO II is a robust and widely adopted model for estimating software project costs, effort, and schedule. Its comprehensive approach, incorporating sizing models, cost drivers, and scale factors, provides accurate estimations tailored to project specifics. By supporting different stages of development and offering flexibility, COCOMO II enhances project planning, resource allocation, and decision-making, contributing to successful project outcomes.

Thank You!

Thank you for your precious time.

Contact:

- **Email:** prasangaramanpokharel@gmail.com

- **Website:** prasangapokharel.com.np