

SEMESTER-V

COURSE 12 A: WEB INTERFACE DESIGNING TECHNOLOGIES

Theory

Credits: 3

3 hrs/week

Course Objectives

1. Understand the principles of web design and distinguish between web and desktop application architectures. Develop static web pages using HTML elements, attributes, and multimedia integration techniques.
2. Style web pages effectively using CSS, including layout control, responsive design, and UI enhancements.
3. Implement dynamic behaviors and form validations using JavaScript and the Document Object Model (DOM).
4. Explore client-side scripting techniques to build responsive, interactive interfaces.
5. Gain foundational knowledge of Content Management Systems (CMS) and apply practical skills in platforms such as WordPress.

Course Outcomes

At the end of the course, students will be able to:

1. Design and structure HTML-based webpages incorporating text, images, tables, forms, and multimedia content.
2. Apply CSS styling rules to manage layout aesthetics, interactivity, and responsiveness across devices.
3. Use JavaScript for string manipulation, event handling, arrays, object operations, and basic validation.
4. Employ client-side scripting to enhance form functionality, create dialog interactions, and add animation via mouse and keyboard events.
5. Analyze different types of CMS and operate WordPress features like posts, pages, themes, plug-ins, and SEO tools to create and deploy basic websites.

Unit 1.HTML:

Introduction to web designing, difference between web applications and desktop applications, introduction to HTML, HTML structure, elements, attributes, headings, paragraphs, images, tables, lists, blocks, symbols, embedding multi-media components in HTML, HTML forms

Unit 2.CSS:

CSS home, introduction, syntax, CSS combinators, colors, background, borders, margins, padding, height/width, text, fonts, tables, lists, position, overflow, float, pseudo class, pseudo elements, opacity, tool tips, image gallery, CSS forms, CSS counters.

Unit 3.Java Script:

What is DHTML, JavaScript, basics, variables, operators, statements, string manipulations, mathematical functions, arrays, functions. objects, regular expressions, exception handling.

Unit 4. Client-Side Scripting:

Accessing HTML form elements using Java Script object model, basic data validations, data format validations, generating responsive messages, opening windows using java script, different kinds of dialog boxes, accessing status bar using java script, embedding basic animative features using different keyboard and mouse events.

Unit 5. Content Management Systems

Introduction to CMS: What is a CMS?, Types: traditional, headless, cloud-based

Popular CMS Platforms: WordPress, Joomla, Drupal, Shopify, When to choose each

Wordpress Basics: Introduction to word press, features, and advantages, wordpress (hosted access and local access), working with posts, managing pages, working with media, working with widgets, working with themes, extending wordpress with plug-ins, SEO and deployment.

Text Book(s)

1. Web Programming Building Internet Applications, Chris Bates, Second Edition, Wiley
2. An Introduction to Web Design plus Programming, Paul S.WangSanda S. Katila, Thomson.

Reference Books

1. Head First HTML and CSS, Elisabeth Robson, Eric Freeman, O'Reilly Media Inc.
2. An Introduction to HTML and JavaScript: for Scientists and Engineers, David R. Brooks. Springer, 2007
3. Schaum's Easy Outline HTML, David Mercer, Mcgraw Hill Professional.
4. Wordpress for Beginners, Dr.Andy Williams

Activities:

Outcome: Design and structure HTML-based webpages incorporating text, images, tables, forms, and multimedia content.

Activity: Create a personal profile webpage using HTML that includes:

- Text (headings, paragraphs)
- Images
- Tables (e.g., contact info or skills)
- Forms (e.g., feedback form)
- Embedded multimedia (e.g., YouTube video or audio clip)

Evaluation Method: Checklist-based review on a 10-point scale:

- Correct use of HTML tags
- Proper structure and nesting
- Functionality of form and media elements
- Visual clarity and completeness

Outcome: Apply CSS styling rules to manage layout aesthetics, interactivity, and responsiveness across devices.

Activity: Style the personal profile page using CSS:

- Apply colors, fonts, spacing
- Use Flexbox or Grid for layout
- Add hover effects and media queries for responsiveness

Evaluation Method: Rubric-based assessment on a 10-point scale:

- Visual appeal and consistency
- Responsive behaviour on different screen sizes
- Use of selectors and layout techniques
- Code cleanliness and organization

Outcome: Use JavaScript for string manipulation, event handling, arrays, object operations, and basic validation.

Activity: Enhance the profile page with JavaScript:

- Validate form inputs (e.g., email format)
- Display a greeting using string manipulation
- Use arrays/objects to store and display hobbies or skills
- Handle click events on button

Evaluation Method: Code demonstration and output testing (10-point scale):

- Correct use of JS syntax and logic
- Functionality of validation and event handling
- Use of arrays/objects
- Console output or alert behaviour

Outcome: Employ client-side scripting to enhance form functionality, create dialog interactions, and add animation via mouse and keyboard events.

Activity: Add interactive features to the form:

- Show/hide sections using mouse events
- Trigger animations on keypress
- Display confirmation dialog on form submission

Evaluation Method: Live demo and observation (10-point scale):

- Smooth interaction and feedback
- Correct use of event listeners
- Animation and dialog behavior
- User experience quality

Outcome: Analyze different types of CMS and operate WordPress features like posts, pages, themes, plug-ins, and SEO tools to create and deploy basic websites.

Activity: Create a basic website using WordPress:

- Add posts and pages
- Apply a theme and customize layout
- Install plug-ins (e.g., contact form, SEO tool)
- Configure basic SEO settings

Evaluation Method: Site walkthrough and checklist (10-point scale):

- Functionality of posts/pages
- Theme customization
- Deployment readiness

SEMESTER-V

COURSE 12 A: WEB INTERFACE DESIGNING TECHNOLOGIES

Practical

Credits: 1

2 hrs/week

List of Experiments:

1. Create an HTML document with the following formatting options:
 - (a) Bold, (b) Italics, (c) Underline, (d) Headings (Using H1 to H6 heading styles), (e) Font (Type, Size and Color), (f) Background (Colored background/Image in background), (g) Paragraph, (h) Line Break, (i) Horizontal Rule, (j) Pre tag
2. Create an HTML document which consists of:
 - (a) Ordered List (b) Unordered List (c) Nested List (d) Image
3. Create a Table with four rows and five columns. Place an image in one column.
4. Collect any ten images of your choice. Using table tag, align the images as follows:



5. Create a menu form using HTML.
6. Style the menu buttons using CSS.
7. Create a form using HTML which has the following types of controls:
 - (a) Text Box (b) Option/radio buttons (c) Check boxes (d) Reset and Submit buttons
8. Embed a calendar object in your web page.
9. Create a form that accepts the information from the subscriber of a mailing system.

Word press:

10. Installation and configuration of word press
11. Access admin panel and manage posts
12. Access admin panel and manage pages
13. Add widgets and menus
14. Create users and assign roles
15. Create a site and add a theme to it