

MODULE DESCRIPTION FORM

نموذج وصف المادة الدراسية

Module Information			
معلومات المادة الدراسية			
Module Title	Web Development		Module Delivery
Module Type	Core		☒ Theory ☒ Lecture ☒ Lab ☐ Tutorial ☐ Practical ☐ Seminar
Module Code			
ECTS Credits	6		
SWL (hr/sem)	150		
Module Level	2	Semester of Delivery	2
Administering Department		College	CSIT
Module Leader		e-mail	
Module Leader's Acad. Title		Module Leader's Qualification	
Module Tutor	Name (if available)	e-mail	E-mail
Peer Reviewer Name	Name	e-mail	E-mail
Scientific Committee Approval Date		Version Number	

Relation with other Modules			
العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	None	Semester	1
Co-requisites module	None	Semester	

Module Aims, Learning Outcomes and Indicative Contents	
أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	

Module Objectives أهداف المادة الدراسية	1. Understand the concept of static web design and its advantages and limitations. 2. Gain proficiency in HTML (Hypertext Markup Language) and CSS (Cascading Style Sheets) as the primary technologies for creating static web pages. 3. Learn the fundamental structure of HTML, including tags, elements, attributes, and their usage in creating web content. 4. Develop skills in creating and formatting different types of content, such as text, images, links, lists, tables, and forms using HTML. 5. Explore the principles of CSS and learn how to apply styles to HTML elements, including fonts, colors, backgrounds, margins, and padding. 6. Understand the box model in CSS and its significance in controlling the layout and positioning of elements on a web page. 7. Learn techniques for creating responsive web designs that adapt to different screen sizes and devices.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	When completing a web programming module focused on PHP, the student can gain the following learning outcomes: 1. Demonstrate a solid understanding of the concept of static web design and its purpose in creating websites. 2. Create well-structured and semantically correct HTML markup for static web pages. 3. Apply CSS styles effectively to enhance the visual presentation and layout of web content. 4. Construct responsive web designs that adapt gracefully to different screen sizes and devices. 5. Implement navigation menus, headers, footers, and other common components of static websites. 6. Optimize web graphics and images for faster loading times without sacrificing quality. 7. Incorporate accessibility considerations into web design to ensure inclusivity and compliance with accessibility standards. 8. Test and debug static web pages to ensure proper functionality across different browsers and devices. 9. Organize and manage website files and directories efficiently for ease of maintenance and scalability. 10. Demonstrate knowledge of best practices in static web design, including code documentation, version control, and collaboration techniques. 11. These learning outcomes reflect the skills and knowledge you should acquire upon completing the module on static web design. Mastery of these outcomes will enable you to design and build visually appealing, functional, and accessible static websites using HTML and CSS.
Indicative Contents المحتويات الإرشادية	1. Introduction to Static Web Design: • Overview of static web design and its role in website development. • Understanding the differences between static and dynamic websites. • Exploring the benefits and limitations of static web design. 2. HTML Fundamentals: • Introduction to HTML markup and its structure.

- Understanding HTML tags, elements, and attributes.
- Creating and formatting text, headings, paragraphs, and lists.
- Working with links, images, and multimedia content.
- Creating tables for data representation.

3. CSS Basics:

- Introduction to CSS and its role in styling web pages.
- Understanding CSS syntax, selectors, and properties.
- Applying colors, backgrounds, and borders to elements.
- Controlling typography and font styles.
- Managing spacing and layout using margins, padding, and the box model.

4. Layout and Responsive Design:

- Creating multi-column layouts using CSS.
- Understanding the concept of responsive web design.
- Using media queries to adapt layouts for different screen sizes.
- Implementing flexible grids and fluid images.
- Applying responsive techniques to navigation menus and other elements.

5. Web Graphics and Optimization:

- Optimizing images for web display, including compression techniques.
- Working with icon fonts and scalable vector graphics (SVG).
- Understanding the impact of file formats and sizes on page load times.
- Implementing techniques to improve web performance, such as caching and minification.

6. Accessibility and Best Practices:

- Understanding the importance of web accessibility.
- Implementing accessibility features, such as alternative text for images and proper semantic markup.
- Following best practices for clean and maintainable code.
- Introduction to version control systems and collaboration tools.

7. Testing, Debugging, and Deployment:

- Testing web pages for cross-browser compatibility and responsiveness.
- Using browser developer tools for debugging and troubleshooting.
- Preparing web pages for deployment and publishing.
- Hosting and maintaining static websites.

8. Project Work:

- Applying the learned concepts and skills to create a complete static website.
- Incorporating responsive design, optimized graphics, and accessibility features.

	Testing, debugging, and refining the website based on feedback and evaluation.
--	--

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	Employing these strategies can create a comprehensive and engaging learning experience in a web programming module, such as lectures, interactive discussions, hands-on lab sessions, case studies, assignments, projects, guest lectures, online resources, assessments, group projects, and continuous support.

Student Workload (SWL) الحمل الدراسي للطلاب محسوب لـ ١٥ أسبوعا			
Structured SWL (hr/sem) الحمل الدراسي المنتظم للطلاب خلال الفصل	75	Structured SWL (hr/w) الحمل الدراسي المنتظم للطلاب أسبوعيا	5
Unstructured SWL (hr/sem) الحمل الدراسي غير المنتظم للطلاب خلال الفصل	75	Unstructured SWL (hr/w) الحمل الدراسي غير المنتظم للطلاب أسبوعيا	5
Total SWL (hr/sem) الحمل الدراسي الكلي للطلاب خلال الفصل	150		

Module Evaluation تقييم المادة الدراسية					
		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
	Quizzes	2	10% (10)	5 and 10	#1, #2 and #3

Formative assessment	Assignments	2	10% (10)	2 and 12	#3, #4 and #6
	Projects / Lab.	1	10% (10)	Continuous	All
	Report	1	10% (10)	13	#5, #6
Summative assessment	Midterm Exam	2hr	10% (10)	7	#1 - #4
	Final Exam	3hr	50% (50)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)	
المنهاج الاسبوعي النظري	
	Material Covered
Week 1	Introduction to Static Web Design: - Overview of static web design and its role in website development. - Understanding the differences between static and dynamic websites. - Exploring the benefits and limitations of static web design.
Week 2	HTML Fundamentals: - Introduction to HTML markup and its structure. - Understanding HTML tags, elements, and attributes. - Creating and formatting text, headings, paragraphs, and lists.
Week 3	HTML Fundamentals: - Working with links, images, and multimedia content. - Creating tables for data representation.
Week 4	CSS Basics: - Introduction to CSS and its role in styling web pages. - Understanding CSS syntax, selectors, and properties. - Applying colors, backgrounds, and borders to elements.
Week 5	CSS Basics: Controlling typography and font styles.

	Managing spacing and layout using margins, padding, and the box model.
Week 6	Layout and Responsive Design: - Creating multi-column layouts using CSS. - Understanding the concept of responsive web design. - Using media queries to adapt layouts for different screen sizes.
Week 7	Layout and Responsive Design: - Implementing flexible grids and fluid images. - Applying responsive techniques to navigation menus and other elements.
Week 8	Web Graphics and Optimization: - Optimizing images for web display, including compression techniques. - Working with icon fonts and scalable vector graphics (SVG).
Week 9	Web Graphics and Optimization: - Understanding the impact of file formats and sizes on page load times. - Implementing techniques to improve web performance, such as caching and minification.
Week 10	Accessibility and Best Practices: - Understanding the importance of web accessibility. - Implementing accessibility features, such as alternative text for images and proper semantic markup.
Week 11	Accessibility and Best Practices: - Following best practices for clean and maintainable code. - Introduction to version control systems and collaboration tools.
Week 12	Testing, Debugging, and Deployment: - Testing web pages for cross-browser compatibility and responsiveness. - Using browser developer tools for debugging and troubleshooting.

Week 13	Testing, Debugging, and Deployment: • Preparing web pages for deployment and publishing. • Hosting and maintaining static websites.
Week 14	Project Work: • Applying the learned concepts and skills to create a complete static website. • Incorporating responsive design, optimized graphics, and accessibility features.
Week 15	Project Work: • Testing, debugging, and refining the website based on feedback and evaluation.
Week 16	Preparatory week before the final Exam

Delivery Plan (Weekly Lab. Syllabus) المنهاج الاسبوعي للمختبر	
	Material Covered
Week 1	Lab 1: Introduction to HTML • Setting up the development environment • Creating the basic structure of an HTML document • Working with text, headings, and paragraphs • Creating lists and adding images
Week 2	Lab 2: HTML Advanced Concepts • Creating hyperlinks and navigation menus • Formatting tables for data representation • Embedding multimedia content (audio, video) • Introduction to forms and form elements
Week 3	Lab 3: CSS Basics • Introduction to CSS and linking stylesheets • Applying colors, backgrounds, and borders

Week 4	Lab 4: CSS Basics - Controlling typography and fonts - Managing spacing and layout using margins, padding, and the box model
Week 5	Lab 5: CSS Layouts and Positioning - Creating multi-column layouts - Implementing float and clear properties
Week 6	Lab 6: CSS Layouts and Positioning - Using flexbox for flexible layouts - Positioning elements (relative, absolute, fixed)
Week 7	Lab 7: Responsive Web Design - Understanding responsive design principles - Using media queries for different screen sizes
Week 8	Lab 8: Responsive Web Design - Creating flexible grids and fluid images - Designing responsive navigation menus
Week 9	Lab 9: Web Graphics and Optimization - Optimizing images for the web (compression, formats) - Working with icon fonts and scalable vector graphics (SVG)
Week 10	Lab 10: Web Graphics and Optimization - Implementing performance optimization techniques - Caching and minification of web assets
Week 11	Lab 11: Accessibility and Best Practices - Introduction to web accessibility guidelines - Implementing accessibility features (alternative text, semantic markup)
Week 12	Lab 12: Accessibility and Best Practices - Following best practices for clean and maintainable code - Version control and collaboration using Git
Week 13	Lab 8: Testing, Debugging, and Deployment - Testing web pages for cross-browser compatibility - Using browser developer tools for debugging and troubleshooting - Preparing web pages for deployment and publishing - Hosting and maintaining static websites

Week14	Lab Project: Complete Static Website Development - Applying the learned concepts and skills to create a full static website - Incorporating responsive design, optimized graphics, and accessibility features - Testing, debugging, and refining the website based on feedback and evaluation - Documentation and final deployment of the website
Week15	Final Exam

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	Available in the Library?
Required Texts	Textbook: "Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics" by Jennifer Niederst Robbins, 5th edition, published in October 2018. "HTML and CSS: Design and Build Websites" by Jon Duckett, 2th edition , published in November 2014.	Yes (E-copy)
Recommended Texts	"HTML and CSS: Visual QuickStart Guide" by Elizabeth Castro and Bruce Hyslop, 8th edition, published in September 2013.	Yes (E-copy)
Websites	https://www.w3schools.com/html/ , https://www.w3schools.com/css/default.asp	

Grading Scheme مخطط الدرجات				
Group	Grade	التقدير	Marks %	Definition
Success Group (50 - 100)	A - Excellent	امتياز	90 - 100	Outstanding Performance
	B - Very Good	جيد جدا	80 - 89	Above average with some errors
	C - Good	جيد	70 - 79	Sound work with notable errors
	D - Satisfactory	متوسط	60 - 69	Fair but with major shortcomings
	E - Sufficient	مقبول	50 - 59	Work meets minimum criteria
Fail Group (0 – 49)	FX – Fail	راسب (قيد المعالجة)	(45-49)	More work required but credit awarded
	F – Fail	راسب	(0-44)	Considerable amount of work required