

Teacher Information, Course Description and
Lecture-wise Plan

of

Web Technologies (CSC336)

under

Pilot Project:

Uniform Teaching, Assessment and Quality
Assurance Across All The Campuses

by

Faculty of Information Science and Technology
COMSATS University Islamabad

A: Teacher Information

Course Name: Web Technologies	Credit Hours: 2+1
Course Code: CSC336	Pre-requisite: CSC241 - Object Oriented Programming
Teacher Name:	Teacher E-mail:
Program:	Semester: Fall 2026
Department: Computer Science	Campus:

B: Course Contents

This course introduces the modern web technologies used for web development. Topics include Overview of Web Platforms; Web Architectures; Markup Languages; Styling; Client-Side Languages; Server-Side Technologies; Use of Databases in Web-based Applications; REST APIs; Application Security, Domain setup and Deployment.

Recommended Books:

1. Web Design Playground: HTML & CSS the Interactive Way, Paul McFedries, Manning, 2019.
2. Eloquent JavaScript, Marijn Haverbeke, 4th Edition, 2024.
3. Learning React: Modern Patterns for Developing React Apps, 2nd Edition, Alex Banks & Eve Porcello, 2020.
4. Learn React Hooks, Daniel Bugl, 2019.
5. Web Programming with HTML5, CSS, and JavaScript, John Dean, Jones & Bartlett Learning, 2018.
6. Murach's Modern JavaScript, Mary Delamater, 2024.

C: Lecture-wise Plan

Week #	Lecture #	Topic (Chapter/Topics Covered)	Practice/ Evaluation	Remarks
1	1 (Date)	Introduction to Web Development: - Course introduction, CLOs, assessment plan and grading policy - Internet vs. Web; client-server request-response model - Core terminologies: URL/URI, HTTP/HTTPS, DNS, hosting, web server and browser - Web standards and bodies (W3C, WHATWG, ECMA, IETF); application types: static, dynamic, MPA, SPA, PWA	Exercises as given in the books	John Dean, Chapter No 1
	2 (Date)	Tiered Web Architecture: - Tiers vs. layers; separation of concerns - One-tier (standalone) architecture with examples - Two-tier (client-server) architecture, advantages and limitations - Three-tier / N-tier architecture; selecting an architecture: scalability, maintainability and cost trade-offs	Exercises as given in the books	John Dean, Chapter No 1
2	3 (Date)	HTML and HTML5 Fundamentals: - Role of markup; HTML document structure (DOCTYPE, head, body) - Elements, tags, attributes, nesting and document validation - Headings, paragraphs, text formatting and hyperlinks (absolute vs. relative) - Images, media, tables and lists; HTML5 new elements, attributes and browser support	Exercises as given in the books	McFedries & John Dean, Chapter No. 2
	4 (Date)	Semantic HTML and HTML Forms: - Block-level vs. inline elements; div and span - Semantic elements (header, nav, main, section, article, aside, footer); accessibility and SEO benefits - Forms: action, method, input types, select, textarea and button - HTML5 client-side validation attributes (required, pattern, min, max)	Exercises as given in the books	McFedries & John Dean, Chapter No. 2 & 10
3	5 (Date)	Cascading Style Sheets (CSS) Fundamentals: - CSS syntax; inline, internal and external style sheets - Selectors: element, class, id, attribute, grouping and descendant	Exercises as given in the books	McFedries & John Dean,

		• Pseudo-classes and pseudo-elements; cascade, specificity and inheritance • Colour, font, text and background properties		Chapter No. 3
	6 (Date)	The CSS Box Model and Display: • Box model: content, padding, border and margin; margin collapsing • box-sizing (content-box vs. border-box); width/height, min/max constraints and overflow • display property: block, inline, inline-block and none • Borders, shadows and consistent spacing techniques	Exercises as given in the books	McFedries & John Dean, Chapter No. 4
4	7 (Date)	CSS Positioning and Stacking: • Normal document flow and static positioning • relative, absolute, fixed and sticky positioning; offset properties and the containing block • z-index and stacking contexts • Floats, clearing and common layout pitfalls	Exercises as given in the books + Assignment No 1	McFedries & John Dean, Chapter No. 4
	8	CSS3 Features: • Rounded corners, gradients, shadows and opacity • 2D/3D transforms; transitions and keyframe animations • Web fonts, icon fonts and CSS custom properties (variables) • Media and feature queries; vendor prefixes and cross-browser compatibility	Exercises as given in the books	McFedries & John Dean, Chapter No. 5
5	9 (Date)	Layout Design using Flexbox and Grid: • Flex container and flex items; main axis vs. cross axis • flex-direction, justify-content, align-items, flex-wrap, flex-grow/shrink/basis • Grid container (grid-template-rows/columns, fr, gap), line placement, spanning and named areas • Choosing Flexbox vs. Grid for a given layout	Exercises as given in the books	McFedries Chapter No. 5
	10 (Date)	Responsive Design and Framework Fundamentals. (Bootstrap or Tailwind) • Mobile-first design, viewport meta tag and fluid units (% , rem, vw/vh) • Media queries, breakpoints and responsive images • Component-based vs. utility-first frameworks;	Quiz # 1	Bootstrap / Tailwind official documentation

		• <i>Bootstrap grid system, components and utilities or Tailwind structure</i> • <i>Tailwind/Bootstrap CSS: utility classes, responsive variants and configuration</i> • <i>Using custom classes</i> • <i>Adding components in UI</i>		
6	11 (Date)	Core JavaScript Concepts (ES6+): Functions, Closures and Objects • <i>Role of JavaScript; embedding scripts, defer and async</i> • <i>var, let and const; primitive vs. reference data types</i> • <i>Operators, type coercion and == vs. ===; control flow: conditionals, loops and switch</i> • <i>Scope (global, function, block), hoisting and the temporal dead zone</i> • <i>Function declarations, expressions and arrow functions</i> • <i>Default and rest parameters; spread operator and destructuring</i> • <i>this binding (call, apply, bind); closures: lexical environment, uses and pitfalls</i> • <i>Objects, arrays, template literals and ES6 modules (import/export)</i>	<i>Exercises as given in the books</i>	<i>Marijn Haverbeke ,& Mary Delamater Chapter No 1 & 2</i>
	12 (Date)	Important Built-in Methods and Array Processing: • <i>Iteration with forEach and for...of</i> • <i>Transformation with map, selection with filter and aggregation with reduce</i> • <i>find, some, every and sort; frequently used string/object methods and method chaining</i> • <i>Immutability, functional style and common mistakes</i>	<i>Exercises as given in the books</i>	<i>Marijn Haverbeke ,& Mary Delamater Chapter No 3,4 & 10</i>
7	13 (Date)	DOM Manipulation and Event Handling: • <i>DOM tree, nodes and the document object</i> • <i>Selecting elements (getElementById, querySelector, querySelectorAll); creating, updating and removing nodes</i> • <i>classList and inline styles; events, addEventListener and the event object</i> • <i>Bubbling, capturing, event delegation, preventDefault and stopPropagation</i>	<i>Exercises as given in the books + Assignment No 2</i>	<i>Marijn Haverbeke , Chapter No 14 & 15 John Dean, Chapter No 8</i>
	14	Regular Expressions and JSON:		

	(Date)	• <i>Regex syntax: literals, character classes, quantifiers and anchors</i> • <i>Groups, alternation, backreferences and flags (g, i, m)</i> • <i>JavaScript regex methods (test, exec, match, replace, split); validation and parsing of form data</i> • <i>JSON syntax, JSON.stringify and JSON.parse for data exchange</i>	<i>Exercises as given in the books</i>	<i>Marijn Haverbeke, Chapter No 9</i>
8	15 (Date)	Asynchronous JavaScript: Promises and Fetch API: • <i>Synchronous vs. asynchronous execution; event loop, task queue and callback hell</i> • <i>Promises: states, then/catch/finally, chaining and Promise.all</i> • <i>async / await with try...catch error handling</i> • <i>AJAX with the Fetch API: GET/POST, headers, response parsing and CORS basics</i>	Quiz # 2	<i>Marijn Haverbeke, Chapter No 11 & 18</i>
	16	Introduction to Server-Side Programming: • <i>Client-side vs. server-side responsibilities</i> • <i>Web server, application server and runtime environment</i> • <i>Framework Architecture: Major concepts like event loop, non-blocking I/O and npm in Express</i> • <i>Project setup in Express.JS (or using Python like Django/FAST API)</i> • <i>First web server, ports and a basic route</i>	<i>Exercises as given in the books</i>	<i>Express/FastAPI/Django official documentation</i>
9	17 (Date)	<i>Midterm Examination</i>	Midterm (25%)	
	18 (Date)	Request Handling and Routing: • <i>HTTP request anatomy: method, URL, headers and body</i> • <i>Routing for GET, POST, PUT, PATCH and DELETE</i> • <i>Route parameters, query strings and request body parsing</i> • <i>Serving static files; organising routers and controllers; 404 and centralised error routes</i>	<i>Exercises as given in the books</i>	<i>Express/FastAPI/Django official documentation & John Dean, Ch. 15</i>
10	19 (Date)	HTTP Status Codes and Common Headers: • <i>Status code families 1xx to 5xx and their intent</i> • <i>Common codes: 200, 201, 204, 301/302, 400, 401, 403, 404, 409, 422, 500</i>	<i>Exercises as given in</i>	<i>Express/FastAPI/</i>

		• <i>Request headers: Accept, Content-Type, Authorization and User-Agent</i> • <i>Response headers (Content-Type, Cache-Control, Set-Cookie, Location) and content negotiation</i>	<i>the books</i>	<i>Django official documentation & MDN HTTP reference</i>
	20 (Date)	Cookies and Sessions: • <i>Statelessness of HTTP and the need for state management</i> • <i>Cookies: creation and attributes (Expires, Path, HttpOnly, Secure, SameSite)</i> • <i>Sessions: session identifier, server-side session stores; session vs. cookie based state</i> • <i>Login/logout flow, session expiry and security considerations</i>	<i>Exercises as given in the books + Assignment No. 3</i>	<i>Express/ FastAPI/ Django official documentation & John Dean, Ch. 16</i>
11	21 (Date)	Middleware or Functions: • <i>Middleware concept and the request/response pipeline</i> • <i>Application-level, router-level and error-handling middleware</i> • <i>Built-in and third-party middleware or functions</i> • <i>Writing custom middleware</i>	<i>Exercises as given in the books</i>	<i>Express/ FastAPI/ Django official documentation</i>
	22 (Date)	Database Connectivity with PostgreSQL / MongoDB: • <i>Relational (PostgreSQL) vs. document (MongoDB) data models</i> • <i>Choosing right Database</i> • <i>Connection setup, connection strings and connection pooling</i> • <i>Performing CRUD operations from server-side code</i> • <i>Schema/collection design and validation</i> • <i>Environment variables and error handling</i>	Quiz # 3	<i>Express/ FastAPI/ Django & MongoDB or PostgreSQL official documentation</i>
12	23 (Date)	Object Relational Mapping (ORM / ODM): • <i>Purpose of an ORM/ODM and object-relational mapping concepts</i> • <i>Models, schemas, fields and data types (Sequelize/Prisma or Mongoose)</i> • <i>CRUD through the ORM: queries, filtering, projection and pagination</i>	<i>Exercises as given in the books</i>	<i>Prisma / Mongoose official documentation</i>

		• <i>Associations and joins or population; migrations, seeding and ORM trade-offs</i>		
	24 (Date)	Response Generation using Templates: • <i>Server-side rendering vs. client-side rendering</i> • <i>Template engines (EJS/Pug/Handlebars or Django templates); passing data to views and interpolation</i> • <i>Loops, conditionals, partials/includes and layouts</i> • <i>Output escaping, form redisplay and flash messages</i>	<i>Exercises as given in the books + Assignment No. 4</i>	<i>Express / Django templating documentation</i>
13	25 (Date)	REST API Development and Authentication: • <i>REST principles: resources, URLs, verbs and statelessness</i> • <i>Designing CRUD endpoints and consistent JSON responses</i> • <i>Session-based authentication and token-based authentication with JWT (issue, verify, refresh)</i> • <i>Password hashing, protected routes and role-based access control</i>	Quiz # 4	<i>Express / Django Official documentation</i>
	26 (Date)	Introduction to React.js and Rendering Approaches: • <i>Why React: single-page applications, declarative UI and components</i> • <i>Virtual DOM and the reconciliation process</i> • <i>Project setup (Vite / CRA), folder structure; JSX expressions, conditional rendering, lists and keys</i> • <i>Rendering approaches: CSR, SSR, SSG and hydration overview</i>	<i>Exercises as given in the books</i>	<i>Alex Banks Chapter No 1-4</i>
14	27 (Date)	Components, Props and Event Handling: • <i>Function components, composition and reusability</i> • <i>Props, children, default props and prop drilling</i> • <i>Rendering behaviour and re-render triggers; React event handling and synthetic events</i> • <i>Controlled components and form handling</i>	<i>Exercises as given in the books</i>	<i>Alex Banks Chapter No 5 & 6</i>
	28 (Date)	Hooks and State Management: • <i>Rules of hooks; useState for local component state</i>	<i>Exercises as given in</i>	<i>Alex Banks</i>

		• <i>useEffect</i> for side effects: dependency array and cleanup • <i>useRef</i>, <i>useMemo</i> and <i>useCallback</i> for references and memoisation • <i>Lifting state up</i>, <i>Context API</i> with <i>useContext</i>, <i>custom hooks</i>; <i>Redux / Zustand</i> overview	<i>the books</i>	<i>Chapter No 7 & 8</i>
15	29 (Date)	Routing and API Integration: • <i>Client-side routing with React Router: routes, links and params</i> • <i>Nested routes, shared layouts, not-found route; programmatic navigation and protected routes</i> • <i>Fetching data with fetch/axios inside effects or query hooks</i> • <i>Loading and error states; consuming the REST API built earlier</i>	<i>Exercises as given in the books</i>	<i>Alex Banks Chapter No 10 & 11</i>
	30 (Date)	Application Security Basics: • <i>Security principles: CIA triad, least privilege and defence in depth</i> • <i>HTTPS/TLS, certificates and security response headers</i> • <i>Authentication vs. authorization; secure password storage</i> • <i>Input validation, output encoding, secrets management</i>	<i>Exercises as given in the books</i>	<i>John Dean, Chapter No 16</i>
16	31 (Date)	Common Web Attacks and Defences: • <i>SQL / NoSQL injection and parameterised queries</i> • <i>Cross-Site Scripting (stored, reflected, DOM-based), sanitisation and CSP</i> • <i>Cross-Site Request Forgery and anti-CSRF tokens</i> • <i>Broken access control, session hijacking/fixation</i> • <i>CORS and security misconfiguration, rate limiting</i>	<i>Exercises as given in the books</i>	<i>OWASP Top 10 Guidelines</i>
	32 (Date)	Domain / DNS Setup and Deployment: • <i>Domain names, registrars and DNS record types (A, AAAA, CNAME, MX, TXT)</i> • <i>DNS resolution flow, TTL and nameserver configuration</i> • <i>Hosting options: shared, VPS, PaaS and serverless</i>	<i>Course Review and Project Deployment</i>	<i>Hosting Guidelines</i>

		• <i>Build and environment configuration; deploying frontend/backend, reverse proxy, SSL and basic monitoring</i>		
17	(Date)		<i>Final (50%)</i>	

Note: All Assignments/Quizzes weightage is 25%