

CAMBODIA ACADEMY OF DIGITAL TECHNOLOGY

INSTITUTE OF DIGITAL TECHNOLOGY

School of Digital Engineering

Course Information			
Course Title	WEB DEVELOPMENT		
Department	Department of Telecommunications and Networking (TN)		
Course Code	XXXXXX	Hour: 45h	Credit: 3
Level	Apply/Analyze	Prerequisite	Basic Programming Concepts (e.g., Python or C), Database
Course Type	Major <input type="checkbox"/> Core <input checked="" type="checkbox"/> Elective <input type="checkbox"/> Other <input type="checkbox"/>		
Offer in Academic Year	Year 02 – Term 02		
Revision	Version 2.0, 2025/12/17		
Instructor Information			
Instructor	CHHUON Sopheakmanith	Qualification	Master’s degree
Mobile	011 263 777	Email	Manith.chhuon@cadt.edu.kh
Telegram	011 263 777		
Office Hour	8:00-12:00; 13:00-17:00		

Course Description

This "Web Development" course provides students with practical, hands-on skills to design, build, and deploy contemporary web applications. The course covers fundamental internet protocols (HTTP/S), modern frontend development using HTML5, responsive CSS (Flexbox, Grid, and Frameworks like Bootstrap or Tailwind CSS), and dynamic JavaScript (ES16). The backend segment focuses on setting up robust RESTful APIs using Node.js and the Express framework, covering routing, database interaction (conceptual), and user authentication. Students will complete a comprehensive full-stack project, integrating all components from version control to deployment.

Course Learning Outcomes

By the end of the course, learners should gain the following *knowledge*:

CLO	Outcome	Session Focus
CLO1	Explain the full web request/response cycle (DNS, HTTP/S) and apply Git for collaborative version control.	1: Git & Protocols
CLO5	Demonstrate proficiency in ES6+ JavaScript, handling DOM interaction and client-side data storage mechanisms.	5: JS, DOM, Storage
CLO6	Explain the Node.js architecture and utilize its core modules (FS, HTTP) for basic server-side logic.	6: Node.js Core Modules
CLO7	Construct modular backend services using Express, defining routes and middleware effectively.	7: Express Routing/Middleware
CLO9	Analyze and implement authentication methods (JWT vs. Session) to secure application routes.	9: Authentication

By the end of the course, learners should be able to use the following **skills**:

CLO	Outcome	Session Focus
CLO2	Design structured interfaces using semantic HTML and fundamental CSS properties (Box Model, Selectors).	2: CSS Basics
CLO3	Implement complex, responsive layouts using modern CSS techniques (Flexbox and Grid).	3: Flexbox & Grid
CLO4	Rapidly prototype and style applications using a utility-first CSS framework (Tailwind CSS).	4: Tailwind CSS
CLO8	Develop robust RESTful APIs implementing all Create, Read, Update, and Delete (CRUD) operations.	8: API/Database CRUD
CLO10	Integrate the final full-stack application, and professionally present the solution with technical documentation.	10: Final Project/Presentation

Learning Level

Course Learning Outcomes		Bloom Taxonomy					
KSA	CLOs	Remember	Understand	Apply	Analyze	Evaluate	Create
Knowledge	CLO1		✓	✓			
	CLO2			✓			
	CLO3			✓	✓		
	CLO4				✓		
Skill	CLO5				✓		
	CLO6				✓		
	CLO7				✓		

	CLO8						✓
Attitude	CLO9			✓			✓
	CLO10			✓			✓

Course Outline/Schedule

RL: P: BL: SDL denotes *Recap Lecture hours, Practical hours, Blended Learning hours, Self-directed learning hours* respectively

Session	Topic	Contact Hours				Required Reading	Homework/ Lab/ Assignments
		RL	P	BL	SDL		
1	1. Introduction to Git Protocols 2. HTML Tool to install VSCode	1.5h	3h	2h	5h	Git and Github Introduction to Internet HTML Doc HTML	1. (L) Practical exercises creating and styling web pages and store in GitHub. 2. (HW) Creating one static website that contains multimedia
2	3. CSS Basics, Selectors & Properties	1.5h	3h	2h	5h	CSS W3	3. (L) Using External and Internal CSS and Working with query Selector 4. (HW) Working with Positioning, list and Table
3	4. CSS Layout: Flexbox and Grid	1.5h	3h	2h	5h	Flex Grid Video	5. (L) Creating a static page by using Flex and grid 6. (HW) Create 1 page by using Tailwind
4	5. Introduction to JavaScript	1.5h	3h	2h	5h	Javascript VDO Introduction to JS W3 Function W3	7. (L) Creating a static page by using Flex and grid 8. (HW) Create 1 page by using Tailwind
5	6. JavaScript DOM Manipulation & Client Storage	1.5h	3h	2h	5h	DOM W3 Storage	9. (L) Working with Document Object and Cookies

							10. (HW) Working with Document Object and Cookies
6	7. Node.js Environment & Core Modules	1.5h	3h	2h	5h	Node.js VDO	11. (L) Working with Functions, Buffer, Module, web server 12. (HW) Working with File system and Events
7	8. Express.js: Routing and Middleware	1.5h	3h	2h	5h		13. (L) Working with Routes, express 14. (HW) Working with static file and middle ware
8	9. Express.js: API Development (CRUD)	1.5h	3h	2h	5h		15. (L) Working with connection, CRUD 16. (HW) Working with CRUD
9	10. Project	1.5h	3h	2h	5h		17. Form team and define topic
10	11. Final and presentation	1.5h	3h	2h	5h		18. Presentation and Final exam

Learning Resource

- **Core Textbook**

1. <https://eloquentjavascript.net/>
2. https://digilib.stekom.ac.id/assets/dokumen/ebook/feb_37d416dac79e1ac59be9d798f8badfd98e8cb48b_1652344592.pdf

- **Additional Reading Materials**

1. <https://tailwindcss.com/docs/installation/using-vite>
2. <https://www.freecodecamp.org/>
3. <https://hackr.io/documents/webdev-ebook.pdf>
4. <https://www.youtube.com/watch?v=xiWUL3M9D8c>

Student Responsibilities

- Students must come to class regularly according to the class schedule.
- Students must watch the video lectures in the LMS system before practicing in physical class.
- Students require to submit their homework and/or assignment on time.
- Students require to submit every quiz in the physical class.

Academic Policy: (Assessment Policy, Plagiarism, and Cheating Policy...)

- **Attendance:** Attending on-site classes is strongly encouraged, and each week students must watch assigned videos and read the assigned chapter, and students need to fill in the preparation outline before class. Students will be required to do discuss, sharing solution, formative assessments.
- **Academic Integrity & Collaboration:** Students must do problem sets, final project and quizzes (exams) only by himself/herself. Questions might be asked by the instructor during/after the submission of the solutions, final project, problem sets and etc. Students shall be guilty of violating the honour code if they: (1) Present the work of others as their own, (2) Use or obtain unauthorized assistance in any academic work, and (3) Give unauthorized assistance to other students.
- **Lab & Homework:** Student needs to submit problem sets on time and read the assigned reading homework before class. Late homework submissions will result in getting a zero score. You are encouraged to discuss the course material with other students. However, you are required to work independently to solve the assigned problem for lab and homework. Copying or plagiarism will be treated as academic dishonesty and will be handled according to institution regulations.
- **Exam:** You are encouraged to discuss the course material with other students and lecturer before the exam. However, you are required to work independently on the midterm and final exams. Students are not allowed to use phone, computer, or book during final exams.
- **Final project:** Students encourage to have discuss with lecturer and classmate about the problems. However, they need to formulate their own solution to solve final project.
- **Penalty:** The penalty for violating the institute honour code is severe. Any student violating the honour code is subject to receive a failing grade for the course and will be reported to The Office of Student Affairs. If you are unclear about whether a particular situation may constitute an honour code violation, you should ask the instructor.

Grading Policy

<i>Activities</i>	<i>Percentage (%)</i>
Attendance & Class Participation	10
Homework	10
Quiz	10
Mid Term	20
Final Project	20
Final exam	30

- **Attendance**
 - To comply with the university regulation, students are encouraged to join all classes. They need to arrive at the class on time.
 - Coming to class after 15min for three times will be counted as one absence.
 - Permission will be allowed only when they have the letter to prove the reason and fill in the form to request permission

- **Class participation**

Students are encouraged to fully engage in all class activities. Punctuality, preparation for class, and prior reading of the assigned chapters/materials are counted as vital parts of class participation. More importantly, daydreaming in class, sending SMS, and especially checking Facebook are strictly prohibited. Students are encouraged to ask questions to their lecturer. Class participation is assessed through individual students' engagement in continuous class activities. 10% of attendance includes participating in the classes regularly and involving in-class activities such as answering individual questions and discussing in groups.

- **Quizzes:**

Quizzes will be administered throughout the course on Moodle. Students should be ready for each of the quizzes. Quizzes are usually fill-in type and MCQ Question type.

- **Midterm:**

Preparation is crucial for success in the midterm exam. Unlike final exams, which cover a comprehensive range of topics, midterm exams typically focus on material covered during the first half of the semester. Therefore, consistent and early preparation is essential for a thorough understanding of the subject matter.

- **Final project:**

For final project, students are required to build single and multithreaded application individually. Then, they need to write a technical report. Students need to submit both whole project and technical report at the end of semester.

- **Final exam**

Preparation is the key to semester exam success. Thus, students need to prepare early during the semester, rather than waiting until the last minute. "One-night policy" is not recommended for the semester exam or other types of assessment above.

Rating Scale

Letter Grade	Grade Point	Score	Explanation
A	4.00	85% - 100%	Excellent
B+	3.50	80% - 84%	Very Good
B	3.00	70% - 79%	Good
C+	2.50	65% - 69%	Fairly Good
C	2.00	50% - 64%	Fair
F	0.00	< 50%	Failure

1. STUDENT LEARNING EXPERIENCE ACTION PLANNING

* Sample are available in the annex below

Week	Contents/Topics	Intended Learning outcomes	Teaching Approach	Delivery modes	Teaching-Learning Activities	Assessment Strategy and Methods	Student Behaviours (Think-feel-do)
1	Introduction to Git, Protocols & HTML <ul style="list-style-type: none"> - <i>Git and GitHub Fundamentals</i> - <i>Web Request/Response Cycle</i> - <i>Semantic HTML Basics</i> 	<ul style="list-style-type: none"> - Explain the full web request/response cycle (DNS, HTTP/S) - Apply Git for collaborative version control. 	<ul style="list-style-type: none"> - Recap lecture - Problem-Based Learning - Lab Experiment - Group Discussion - Quiz 	<ul style="list-style-type: none"> - UMS System - MS. Teams - Forum Discussion - Telegram Group 	<ul style="list-style-type: none"> - Lecture Review: Detailed walkthrough of the Git workflow, HTTP verbs, and semantic HTML structure. - Demonstration: Live demo of cloning, committing, branching, and pushing to GitHub. 	<ul style="list-style-type: none"> - Formative and/or summative assessment - Individual or group <p>Assessment types and weightage</p>	<p>How do you hope this learning experience will impact your students' feelings, thinking, and doing – individually and as a group?</p> <ul style="list-style-type: none"> - Critical thinking skills - Problem solving abilities - Communication skill - Teamwork - Research material
2	CSS Basics, Selectors & Properties <ul style="list-style-type: none"> - CSS Box Model - External/Internal CSS - Positioning and Selectors 	<ul style="list-style-type: none"> - Design structured interfaces using semantic HTML and fundamental CSS properties (Box Model, Selectors). 	<ul style="list-style-type: none"> - Lecture review - Recap lecture - Problem-Based Learning - Lab Experiment - Group Discussion 	<ul style="list-style-type: none"> - UMS System - MS. Teams - Forum Discussion - Telegram Group 	<ul style="list-style-type: none"> - Active discussion in class regarding efficient solving of problems - Lecture & Review: Discuss specificity, cascade, and inheritance in CSS. Review the four main types of positioning. - Lab Experiment: Guided coding session on using advanced CSS selectors (e.g., attribute selectors, pseudo-classes). 	<ul style="list-style-type: none"> - Lab activities - Homework 2 (HW2): Working with Positioning, lists, and Table styling to reproduce a given design accurately. - Blended learning hours on LMS - Classwork 	<ul style="list-style-type: none"> - Critical thinking skills - Problem solving abilities - Communication skill - Teamwork - Research material
3	CSS Layout: Flexbox and Grid <ul style="list-style-type: none"> - <i>Flexbox (1D Layout)</i> - <i>CSS Grid (2D Layout)</i> - <i>Responsiveness Basics</i> 	<ul style="list-style-type: none"> - Implement complex, responsive layouts using modern CSS techniques (Flexbox and Grid). 	<ul style="list-style-type: none"> - Lecture review - Recap lecture - Problem-Based Learning - Lab Experiment - Group Discussion 	<ul style="list-style-type: none"> - UMS System - MS. Teams - Forum Discussion - Telegram Group 	<ul style="list-style-type: none"> - Active discussion in class regarding efficient solving of problems - Assignment and presentation of diverse problems based on the corresponding lecture - Evaluation of class performance to reach each student in class for every topic - Problem-Based Learning: Present common website layouts (e.g., complex blog post, product gallery) and challenge students to 	<ul style="list-style-type: none"> - Lab activities - Homework 3 (HW3): Create one fully responsive landing page using only CSS Grid, ensuring adaptation across device sizes. - Assignment - Blended learning hours on LMS - Classwork 	<ul style="list-style-type: none"> - Critical thinking skills - Problem solving abilities - Communication skill - Teamwork - Research material

					implement them using Flexbox or Grid. - Demonstration: Live coding of a responsive navigation bar using Flexbox.		
4	Introduction to JavaScript <i>-JS Fundamentals & Syntax</i> <i>-Variables (let, const) & Data Types</i> <i>-Basic Logic & Control Flow</i>	- Understand basic JavaScript syntax - Implement conditional logic to control simple web interactions.	- Lecture review - Recap lecture - Problem-Based Learning - Lab Experiment - Group Discussion	- UMS System - MS. Teams - Forum Discussion - Telegram Group	- Active discussion in class regarding efficient solving of problems - Assignment and presentation of diverse problems based on the corresponding lecture - Evaluation of class performance to reach each student in class for every topic - Lecture Review: Explanation of the JS engine and variable hoisting. Focus on the role of JS in the web stack. - Demonstration: Live coding of arithmetic, string concatenation, and if/else logic in the browser console.	- Lab activities - Homework 4 (HW4): Create a "Decision Maker" script that responds dynamically to different user inputs by using form. - Blended learning hours on LMS - Classwork	- Critical thinking skills - Problem solving abilities - Communication skill - Teamwork - Research material
5	JavaScript DOM Manipulation & Client Storage <i>- Event Handling and DOM Methods</i> <i>- localStorage and sessionStorage</i>	- Demonstrate proficiency in ES6+ JavaScript, handling DOM interaction and client-side data storage mechanisms.	- Lecture review - Recap lecture - Problem-Based Learning - Lab Experiment - Group Discussion	- UMS System - MS. Teams - Forum Discussion - Telegram Group	- Active discussion in class regarding efficient solving of problems - Assignment and presentation of diverse problems based on the corresponding lecture - Evaluation of class performance to reach each student in class for every topic - Lecture Review: Focus on addEventListener, createElement, and safe DOM manipulation. - Lab Experiment: Guide students through building an interactive web widget that requires state persistence.	- Lab activities - Homework 5 (HW5): Create a basic to-do list application where tasks persisted (saved and loaded) using localStorage. Assignment - Blended learning hours on LMS - Classwork	- Critical thinking skills - Problem solving abilities - Communication skill - Teamwork - Research material

6	Node.js Environment & Core Modules <ul style="list-style-type: none"> - <i>Node.js Architecture</i> - <i>Core Modules (http, fs)</i> 	<ul style="list-style-type: none"> - Explain the Node.js architecture and utilize its core modules (FS, HTTP) for basic server-side logic. 	<ul style="list-style-type: none"> - Lecture review - Recap lecture - Problem-Based Learning - Lab Experiment - Group Discussion 	<ul style="list-style-type: none"> - UMS System - MS. Teams - Forum Discussion - Telegram Group 	<ul style="list-style-type: none"> - Active discussion in class regarding efficient solving of problems - Assignment and presentation of diverse problems based on the corresponding lecture - Evaluation of class performance to reach each student in class for every topic - Code Walkthrough: Step-by-step introduction to initializing a Node.js project (npm init). Demonstrate creating a simple HTTP server. - Lab Experiment: Guide students through using the fs module to read and write files. 	<ul style="list-style-type: none"> - Lab activities - Homework 6 (HW6): Working with the File System (fs) module to read/write a log file, and implementing Node.jsEvents. - Blended learning hours on LMS - Classwork 	<ul style="list-style-type: none"> - Critical thinking skills - Problem solving abilities - Communication skill - Teamwork - Research material
7	Mid-Term Exam	Paper Exam (0.5 hours); Laboratory (1 hours)					
8	Express.js: Routing and Middleware <ul style="list-style-type: none"> - <i>Express Setup and Installation</i> - <i>Routing and HTTP Methods</i> 	<ul style="list-style-type: none"> - Construct modular backend services using Express, defining routes and middleware effectively. 	<ul style="list-style-type: none"> - Lecture review - Recap lecture - Problem-Based Learning - Lab Experiment - Group Discussion 	<ul style="list-style-type: none"> - UMS System - MS. Teams - Forum Discussion - Telegram Group 	<ul style="list-style-type: none"> - Active discussion in class regarding efficient solving of problems - Assignment and presentation of diverse problems based on the corresponding lecture - Evaluation of class performance to reach each student in class for every topic - Demonstration: Show how Express simplifies routing. Explain the concept and order of execution for middleware. - Lab Experiment: Guided setup of an Express application with parameterised routes. 	<ul style="list-style-type: none"> - Lab activities - Homework 7 (HW7): Working with static file serving and implementing a custom middleware function. - Assignment - Blended learning hours on LMS - Classwork 	<ul style="list-style-type: none"> - Critical thinking skills - Problem solving abilities - Communication skill - Teamwork - Research material
9	Express.js: API Development (CRUD) <ul style="list-style-type: none"> - <i>RESTful API Design Principles</i> - <i>CRUD Operations Implementation</i> 	<ul style="list-style-type: none"> - Develop robust RESTful APIs implementing all Create, Read, Update, and Delete (CRUD) operations. 	<ul style="list-style-type: none"> - Lecture review - Recap lecture - Lab Experiment - Group Discussion 	<ul style="list-style-type: none"> - UMS System - MS. Teams - Forum Discussion - Telegram Group 	<ul style="list-style-type: none"> - Active discussion in class regarding efficient solving of problems - Assignment and presentation of diverse problems based on the corresponding lecture 	<ul style="list-style-type: none"> - Lab activities - Homework 8 (HW8): Implement a second, related resource (e.g., "Comments" for a "Post") and implement 	<ul style="list-style-type: none"> - Critical thinking skills - Problem solving abilities - Communication skill - Teamwork - Research material

					<ul style="list-style-type: none"> - Evaluation of class performance to reach each student in class for every topic - Lecture Review: Discuss best practices for designing REST APIs and using appropriate HTTP status codes. - Problem Solving: Guide the implementation of a mock database to handle persistent data and apply all four CRUD methods. 	all CRUD operations for it. <ul style="list-style-type: none"> - Assignment - Blended learning hours on LMS - Classwork 	
10	Project Planning and Setup <ul style="list-style-type: none"> - <i>Team Formation</i> - <i>API Endpoint Planning</i> - <i>Authentication Basics</i> 	- Analyze and implement authentication methods (JWT vs. Session) to secure application routes.	- Presentations	<ul style="list-style-type: none"> - UMS System - MS. Teams - Forum Discussion - Telegram Group 	<ul style="list-style-type: none"> - Active discussion in class regarding efficient solving of problems - Assignment and presentation of diverse problems based on the corresponding lecture - Evaluation of class performance to reach each student in class for every topic - Group Formation: Facilitate team formation and project idea generation/vetting. - Workshop: Guided session on creating the conceptual database schema and mapping out the full list of required API endpoints. 	<ul style="list-style-type: none"> - Lab activities - Assignment: Documenting the Project Plan, including wireframes, API endpoints, and database schema. - Assignment - Blended learning hours on LMS - Classwork 	<ul style="list-style-type: none"> - Critical thinking skills - Problem solving abilities - Communication skill - Team work - Research material
11	Final Project and Presentation	- Integrate the final full-stack application, and professionally present the solution with technical documentation.	<ul style="list-style-type: none"> - Lecture review - Recap lecture - Problem-Based Learning - Lab Experiment - Group Discussion 	<ul style="list-style-type: none"> - UMS System - MS. Teams - Forum Discussion - Telegram Group 	<ul style="list-style-type: none"> - Active discussion in class regarding efficient solving of problems - Assignment and presentation of diverse problems based on the corresponding lecture - Evaluation of class performance to reach each student in class for every topic 	<ul style="list-style-type: none"> - Assignment: Final Project Presentation 	<ul style="list-style-type: none"> - Critical thinking skills - Problem solving abilities - Communication skill - Teamwork - Research material
12	Final Exam	Paper Exam (3 hours); Laboratory (3 hours)					