COURSE OUTLINE: CSD213 - WEB DEVELOPMENT II

Prepared: Rodney Martin
Approved: Corey Meunier, Chair, Technology and Skilled Trades

Course Code: Title	CSD213: WEB DEVELOPMENT II
Program Number: Name	2095: COMPUTER PROGRAMMING
Department:	COMPUTER STUDIES
Academic Year:	2022-2023
Course Description:	Students in this course learn how to take static web sites and turn them into dynamic and interactive web applications using modern web technologies. The Document Object Model (DOM) of web browsers is introduced, and students learn to create and manipulate DOM objects in response to user actions and system events. Specific focus is given to securing and validating HTML Forms. Students also learn how to interact securely with Web APIs and various client-side APIs, such as the Web Storage and Geolocation APIs. The programming languages JavaScript and TypeScript are used in this course.
Total Credits:	4
Hours/Week:	4
Total Hours:	56
Prerequisites:	CSD112, CSD121
Corequisites:	There are no co-requisites for this course.
This course is a pre-requisite for:	CSD223, CSD226, CSD227, CSD228
Vocational Learning Outcomes (VLO's) addressed in this course: **Please refer to program web page for a complete listing of program outcomes where applicable.**	**2095 - COMPUTER PROGRAMMING** VLO 2 Contribute to the diagnostics, troubleshooting, documenting and monitoring of technical problems using appropriate methodologies and tools. VLO 3 Implement and maintain secure computing environments. VLO 8 Adhere to ethical, legal, and regulatory requirements and/or principles in the development and management of computing solutions and systems. VLO 10 Contribute to the development, documentation, implementation, maintenance and testing of software systems by using industry standard software development methodologies based on defined specifications and existing technologies/frameworks. VLO 11 Apply one or more programming paradigms such as, object-oriented, structured or functional programming, and design principles, as well as documented requirements, to the software development process.
Essential Employability Skills (EES) addressed in this course:	EES 1 Communicate clearly, concisely and correctly in the written, spoken, and visual form that fulfills the purpose and meets the needs of the audience. EES 2 Respond to written, spoken, or visual messages in a manner that ensures effective communication.

	EES 5 Use a variety of thinking skills to anticipate and solve problems. EES 6 Locate, select, organize, and document information using appropriate technology and information systems. EES 10 Manage the use of time and other resources to complete projects. EES 11 Take responsibility for ones own actions, decisions, and consequences.
Course Evaluation:	Passing Grade: 50%, D A minimum program GPA of 2.0 or higher where program specific standards exist is required for graduation.
Other Course Evaluation & Assessment Requirements:	To successfully pass this course, the student must receive passing grades for both the Test and Evaluation portion of the class AND the Laboratory portion. Grade Definition Grade Point Equivalent A+ 90 - 100% 4.00 A 80 - 89% B 70 - 79% 3.00 C 60 - 69% 2.00 D 50 - 59% 1.00 F (Fail) 49% and below 0.00 CR (Credit) Credit for diploma requirements has been awarded. S Satisfactory achievement in field /clinical placement or non-graded subject area. U Unsatisfactory achievement in field/clinical placement or non-graded subject area. X A temporary grade limited to situations with extenuating circumstances giving a student additional time to complete the requirements for a course. NR Grade not reported to Registrar`s office. W Student has withdrawn from the course without academic penalty.
Books and Required Resources:	Learn Web Development by MDN Publisher: MDN https://developer.mozilla.org/en-US/docs/Learn Eloquent JavaScript, 3rd edition by Marijn Haverbeke https://eloquentjavascript.net/

Course Outcomes and Learning Objectives:

Course Outcome 1	Learning Objectives for Course Outcome 1
1. Use JavaScript to build working programs	1.1 Write programs that use variables, conditionals, loops, functions, I/O, and error handling 1.2 Use built-in string and mathematics functions 1.3 Use basic data types such as strings, numbers, arrays, objects, Maps, and Dates 1.4 Explain JavaScript`s type coercion and weak type system 1.5 Create inheritance hierarchies using classes 1.6 Describe JavaScript`s prototype system 1.7 Discuss how JavaScript`s dynamic typing precludes the need for explicit interfaces 1.8 Describe the difference between client-side and server-side

	JavaScript programming 1.9 Use JavaScript debugging tools to debug JavaScript applications
Course Outcome 2	**Learning Objectives for Course Outcome 2**
2. Use the DOM API to manipulate and enhance static web pages	2.1 Access elements by id, tag name, class, name, or selector 2.2 Read and change element content, CSS properties, and attributes 2.3 Add and remove document nodes 2.4 Create and close new browser tabs and windows
Course Outcome 3	**Learning Objectives for Course Outcome 3**
3. Use events and event handling to respond to user interactions and system events	3.1 Define events and event handlers 3.2 Discuss the nature of callback functions 3.3 Create programs that respond to user and system events 3.4 Create programs that feature timed/periodic operations
Course Outcome 4	**Learning Objectives for Course Outcome 4**
4. Enhance HTML forms with custom validation	4.1 Enhance form usability with JavaScript 4.2 Customize browser-based HTML validation 4.3 Implement custom validation to check for errors and display error messages 4.4 Discuss the security vulnerabilities that may arise when using forms, and how to prevent them 4.5 Use encoding and decoding to prevent cross-site scripting (XSS) attacks 4.6 Explain the how cross-site request forgery (CSRF) tokens prevent CSRF attacks
Course Outcome 5	**Learning Objectives for Course Outcome 5**
5. Create secure, asynchronous HTTP requests and handle responses	5.1 Create and use XmlHttpRequest objects to make dynamic HTTP requests 5.2 Discuss the Same-Origin Policy and its role in web security 5.3 Use the Cross-Origin Resource Sharing (CORS) mechanism to make cross-origin requests 5.4 Explain the difference between synchronous and asynchronous operations 5.5 Describe the disadvantages of using callbacks in asynchronous operations 5.6 Describe the nature of Promises 5.7 Write Promise-based code using both the raw and the async/await syntax 5.8 Create HTTP requests using the Fetch API
Course Outcome 6	**Learning Objectives for Course Outcome 6**
6. Use other common Web APIs such as Web Storage, Geolocation, etc, as time permits	6.1 Use the History, Location, Navigation, and Screen objects to manipulate the browser window 6.2 Explain what cookies are and how they are used in web applications 6.3 Describe the Web Storage APIs, and explain when they are useful instead of cookies 6.4 Write programs that use cookie, localStorage, and sessionStorage data

	6.5 Write programs using other browser APIs as time permits

Course Outcome 7	Learning Objectives for Course Outcome 7
7. Use TypeScript to add type safety to web code	7.1 Describe the structural type system that TypeScript adds to JavaScript 7.2 Configure a TypeScript development environment 7.3 Add simple type annotations to existing JavaScript code

Evaluation Process and Grading System:

Evaluation Type	Evaluation Weight
Lab Assignments	40%
Test 1	20%
Test 2	20%
Test 3	20%

Date: June 1, 2022

Addendum: Please refer to the course outline addendum on the Learning Management System for further information.