CS 50C Course Outline as of Fall 2023

CATALOG INFORMATION

Dept and Nbr: CS 50C Title: WEB DEVELOPMENT 3
Full Title: Web Development 3 - JavaScript
Last Reviewed: 10/24/2022

Units		Course Hours per Week		Nbr of Weeks	Course Hours Total	
Maximum	3.00	Lecture Scheduled	3.00	17.5	Lecture Scheduled	52.50
Minimum	3.00	Lab Scheduled	0	6	Lab Scheduled	0
		Contact DHR	0		Contact DHR	0
		Contact Total	3.00		Contact Total	52.50
		Non-contact DHR	0		Non-contact DHR	0

Total Out of Class Hours: 105.00 Total Student Learning Hours: 157.50

Title 5 Category:	AA Degree Applicable
Grading:	Grade or P/NP
Repeatability:	00 - Two Repeats if Grade was D, F, NC, or NP
Also Listed As:	
Formerly:	CS 50.12

Catalog Description:
This course focuses on JavaScript programming for client-side Web development. Students learn to create advanced interactive projects including games, data visualizations, generative art, mobile applications, and other browser-based interactive experiences. Students gain experience working with open-source JavaScript libraries such as jQuery, the Google Maps API, D3.js and many others. Project-based assignments lead to a comprehensive portfolio website of all class projects.

Prerequisites/Corequisites:
Course Completion or Current Enrollment in CS 50B

Recommended Preparation:
Eligibility for ENGL 1A or equivalent

Limits on Enrollment:

Schedule of Classes Information:
Description: This course focuses on JavaScript programming for client-side Web development. Students learn to create advanced interactive projects including games, data visualizations, generative art, mobile applications, and other browser-based interactive experiences. Students

gain experience working with open-source JavaScript libraries such as jQuery, the Google Maps API, D3.js and many others. Project-based assignments lead to a comprehensive portfolio website of all class projects. (Grade or P/NP)
Prerequisites/Corequisites: Course Completion or Current Enrollment in CS 50B
Recommended: Eligibility for ENGL 1A or equivalent
Limits on Enrollment:
Transfer Credit: CSU;
Repeatability: Two Repeats if Grade was D, F, NC, or NP

ARTICULATION, MAJOR, and CERTIFICATION INFORMATION:

AS Degree:	**Area**			Effective:	Inactive:
CSU GE:	**Transfer Area**			Effective:	Inactive:
IGETC:	**Transfer Area**			Effective:	Inactive:
CSU Transfer:	Transferable	Effective:	Fall 2011	Inactive:	
UC Transfer:		Effective:		Inactive:	

CID:

Certificate/Major Applicable:
Both Certificate and Major Applicable

COURSE CONTENT

Student Learning Outcomes:
At the conclusion of this course, the student should be able to:
1. Code and deploy web and mobile projects using advanced HTML, CSS, and JavaScript.
2. Demonstrate mastery and application of shared JavaScript libraries to create highly interactive user experiences.
3. Follow professional best practices for file management and version control of web and mobile projects.

Objectives:
At the conclusion of this course, the student should be able to:
1. Develop interactive websites and mobile applications that integrate HTML, CSS, and JavaScript.
2. Utilize shared JavaScript libraries to implement advanced interactivity and functionality.
3. Analyze and customize JavaScript code.
4. Write JavaScript code that selects, manipulates, and creates document elements, accesses, validates, and parses external data sources.
5. Apply appropriate user experience and interactive design concepts to custom web and mobile applications.

Topics and Scope:

I. Advanced HTML and CSS Review
   A. Semantic HTML review
   B. SCRIPT and NOSCRIPT elements

C. CSS language review
II. JavaScript Language Fundamentals
A. Browser-based JavaScript: working with JavaScript in the browser
B. Data types, values and variables
C. Operators and expressions
D. Control structures: loops, conditionals, functions
E. Events
F. Data structures: objects and arrays
G. Common JavaScript objects in the browser: navigator, document, window, Math
H. Document Object Model and JavaScript
I. Data storage and dynamic data: cookies, local storage, JavaScript Object Notation (JSON)
J. Forms: events, elements and validation
K. Error-handling, debugging, and troubleshooting
L. Asynchronous programming
M. Browser API access via JavaScript
N. JavaScript timers
O. Server-side JavaScript: working with Node.js and Node Package Manager (NPM)
P. History of JavaScript and ECMAScript
III. Common JavaScript Libraries and Functionalities
A. jQuery vs. pure JavaScript for accessing and modifying DOM
B. jQuery UI and other jQuery plugins for advanced user interfaces
C. Geographic visualizations with libraries such as OpenLayers and Google Maps API
D. Data visualization with HTML Canvas and visualization libraries such as Data-Driven Documents (D3.js)
E. Highly interactive gamified user experiences with JavaScript game engine libraries such as Phaser.js
F. Creating your own JavaScript library or jQuery plugin
IV. Introduction to Web and Mobile Applications
A. Model-View-Controller (MVC) coding pattern
B. Reactive JavaScript libraries such as Vue.js and React
C. Mobile gesture-based user interfaces
D. Introduction to progressive Web applications (PWA)
E. Introduction to full-stack applications
F. Application user experience and interaction design
V. Professional Practices
A. Introduction to git command-line version control and practices
B. Introduction to file management and developer operations
C. Introduction to build systems
D. Transpilation of code to pure JavaScript

Assignment:

1. Textbook and other assigned reading (25-60 pages per week)
2. Browser-based JavaScript assignments (7-15). All HTML and CSS code submitted must be validated.
3. Midterm and final JavaScript-powered projects (2). Each project must include:
A. Custom JavaScript coding
B. Fully validated and accessible code
4. Quizzes and exams (2-4)
5. Discussions (5-8). Conducted in-class or online with participation from all students.
6. Project presentations and peer feedback (2). Each student must present their project to classmates, either online or in-class, and provide feedback to at least two peers. May be

ungraded.

Methods of Evaluation/Basis of Grade:

Writing: Assessment tools that demonstrate writing skills and/or require students to select, organize and explain ideas in writing.

Project presentations and peer feedback	Writing 0 - 20%

Problem Solving: Assessment tools, other than exams, that demonstrate competence in computational or non-computational problem solving skills.

Browser-based JavaScript assignments	Problem solving 30 - 60%

Skill Demonstrations: All skill-based and physical demonstrations used for assessment purposes including skill performance exams.

Midterm and final JavaScript-powered projects	Skill Demonstrations 10 - 30%

Exams: All forms of formal testing, other than skill performance exams.

Quizzes and exams	Exams 20 - 40%

Other: Includes any assessment tools that do not logically fit into the above categories.

Participation in discussions	Other Category 10 - 20%

Representative Textbooks and Materials:
Eloquent JavaScript. 3rd ed. Haverbeke, Marijn. No Starch Press. 2018.
JavaScript: The Definitive Guide. 7th ed. Flanagan, David. O'Reilly Media. 2020.
Instructor prepared materials