



CodeHS

Web Development Capstone Syllabus

1 year for High School (145-155 contact hours)

Course Overview and Goals

The Web Development Capstone Course is intended to teach students the fundamentals of web development in a project-based learning environment. Students will be taught the basic elements of web development, such as web hosting, file organization, and incorporating Javascript into HTML files. Over the course of the school year, students will collaboratively and independently design, develop and implement functional and responsive web pages using these foundational skills.

Learning Environment: The course utilizes a blended classroom approach. The content is fully web-based, with students writing and running code in the browser. Teachers utilize tools and resources provided by CodeHS to leverage time in the classroom and give focused 1-on-1 attention to students. Each unit of the course is broken down into lessons. Lessons consist of video tutorials, short quizzes, example programs to explore, and written programming exercises, adding up to over 100 hours of hands-on programming practice in total. Each unit ends with a comprehensive unit test that assesses student's mastery of the material from that unit as well as challenge problems where students can display their understanding of the material.

Programming Environment: Students write and run JavaScript and HTML programs in the browser using the CodeHS editor.

More information: Browse the content of this course at <https://codehs.com/course/7968>

Prerequisites: This course is the third course in the Web Development pathway. It is designed ideally for students who have an introductory knowledge of HTML, CSS, and Javascript. Students who have completed the CodeHS *Web Design*, *Intro to Programming in Javascript*, or *AP Computer Science Principles* courses are eligible to complete this course.

Course Breakdown

Unit 1: Introduction to JavaScript in HTML (3 weeks/15 hours)

Students learn about the script tag, and how it can be used to write JavaScript code in their HTML files. Students are also introduced to useful JavaScript methods that can be used to alter the state of the CSS and HTML of a webpage, as well as how the Document Object Model supports the ability to make such changes.

Objectives / Topics Covered	<ul style="list-style-type: none">• The Script Tag• Using the DOM• Creating Elements Using the DOM• Styling Elements Using JavaScript• Functions in HTML• This Keyword
-----------------------------	---

	<ul style="list-style-type: none"> ● Keyboard Interactions ● JavaScript Animations ● Positioning and Animations
Example Assignments / Labs	<ul style="list-style-type: none"> ● Make a Quilt <ul style="list-style-type: none"> ○ Students practice iteration and appending elements to a web page. They also develop a randomizer set to initiate whenever the mouse hovers over an element. ● Challenge: Make a Keyboard <ul style="list-style-type: none"> ○ Students create a clickable keyboard using only JavaScript. This is a three part project, where students develop the keyboard, make it clickable, and finally get text to show up on a webpage. ● Add Div Animation <ul style="list-style-type: none"> ○ Students practice developing animations by creating a program designed to create and move divs across a webpage. Students will need to be able to access the height and width of the page, as well as the total distance that the elements have to move before stopping.

Unit 2: Using JavaScript Libraries (2 weeks/ 10 hours)

Students are introduced to jQuery, a JavaScript library that makes webpage interaction easier. Students will learn the basic syntax of jQuery, how to incorporate it in their webpages, and useful methods that help animate and change the responsiveness of their websites.

Objectives / Topics Covered	<ul style="list-style-type: none"> ● Introduction to jQuery ● Iterating with jQuery ● Animations with jQuery ● Callback Functions and Synchronization ● Using Multiple Files in JavaScript
Assignments / Labs	<ul style="list-style-type: none"> ● Smart jQuery Table <ul style="list-style-type: none"> ○ Students learn how to create a smart table that changes colors based on the content of the table. If the values in the table are too low, then the table flags them as important. ● The Wave <ul style="list-style-type: none"> ○ Students learn how to create animations that rely on asynchronous execution. This assignment has them practice timing animations using callback functions. ● FrankenDiv <ul style="list-style-type: none"> ○ This assignment has students practice manipulating elements using the jQuery animations function. Students will create actions to change the size, color, and proportions of a div.

Unit 3: Project: Build an Interactive Resume (2-3 weeks/ 10-15 hours)

Students will create a single page, interactive resume. Students will be expected to add several animations using jQuery to prove their ability to modify HTML and CSS. This unit also examines web design theories that can help students improve the aesthetics of their resumes, and to evaluate the quality of a website based on its layout.

Objectives / Topics	<ul style="list-style-type: none"> ● Creating a Resume
---------------------	---

Covered	<ul style="list-style-type: none"> ● Layout and Web Design Theory ● Interactive Resume Development
Assignments / Labs	<ul style="list-style-type: none"> ● Online Interactive Resume <ul style="list-style-type: none"> ○ Students will develop an online interactive resume, and evaluate the quality of their peers resumes using rubrics to initiate feedback. Students will also determine the quality of the layout, using web design layout theory to establish their reasoning.

Unit 4: Storing and Collecting Data (4-5 weeks/ 20-25 hours)

This unit will explore the role that data plays in creating websites. Students will learn about the various ways that data is taken from web pages, as well as ways to secure themselves from unwanted data collection. Students will also learn how to incorporate data collection into their own websites, and collect simple information from users.

Objectives / Topics Covered	<ul style="list-style-type: none"> ● Collecting and Storing Data ● Storing Data with Local Storage ● Storing Data Structures with Local Storage ● JavaScript Objects ● Using Input Fields [ADD MATERIALS]
Example Assignments / Labs	<ul style="list-style-type: none"> ● Saving Active Divs <ul style="list-style-type: none"> ○ Students will learn how to save which elements on their page are active at a given time, and reproduce the same active state the next time a user loads the page. ● Contact List <ul style="list-style-type: none"> ○ Students will create a phonebook that stores and collects information about users' names and numbers, as well as retrieve information about users in their phonebook. ● Birthday Tracker <ul style="list-style-type: none"> ○ Students will create a Birthday Tracker to log birthdays. If a user enters a date, it will generate a list of all users who have that birthday. ○ [Add Materials]

Unit 5: Project: Collecting Data (2-3 weeks/ 10-15 hours)

Students will create a multi-file webpage that tracks the number of clicks that items of content on the site receive, allowing the web owner to make decisions about which content should be kept, and which should be changed out the next time that they update their site. Students will test out one another's webpages, and write a written response highlighting the content that they will change on the next iteration and why.

Objectives / Topics Covered	<ul style="list-style-type: none"> ● Define Your Product ● Develop Hypotheses <ul style="list-style-type: none"> ○ Create a User Persona ● Create Your Data Collection Site ● Collecting User Data ● Analyzing User Data ● Making Decisions
-----------------------------	---

Assignments / Labs	<ul style="list-style-type: none"> ● Create Your Own User Persona <ul style="list-style-type: none"> ○ Students create a user persona for their website to make predictions about how users will interact with their site, and how to tailor their site to the needs of potential users. ● Data Collection Site <ul style="list-style-type: none"> ○ Students will create a website that tracks user data. They will use this data to make informed decisions about how to improve their website in future lessons.
--------------------	---

Unit 6: How to Build and Maintain a Website (2 weeks/ 10 hours)

While students can run functional and responsive websites on CodeHS, this unit explores how students can create and run websites without the help of CodeHS. This unit will explore how to store web files, secure a domain name, and maintain a website. The majority of the lessons will be explanatory - students will not be expected to host and maintain a webpage off CodeHS, but will be given the tools needed to do so if they desire.

Objectives / Topics Covered	<ul style="list-style-type: none"> ● Choosing a Domain Name ● Personal Servers vs Web Servers ● Web Optimization ● Choosing a CMS
Assignments / Labs	<ul style="list-style-type: none"> ● Comparing CMS Systems <ul style="list-style-type: none"> ○ Students compare popular CMS systems and make decisions about which CMS is best suited for their own personal needs. ● Yoast Analyzer <ul style="list-style-type: none"> ○ Students use a Yoast analyzer to determine if their website, or websites that they frequent are optimized for search engines. Students learn how to optimize their websites by organizing their file structure and using metadata for search engine crawlers.

Unit 7: Final Project (2-4 weeks/ 10-20 hours)

Students will be tasked with creating a website of their own choosing. The website will have to follow specific criteria - certain number of pages, responsiveness, and use of APIs. Students will go through a feedback process, and learn about making their websites more accessible to a wide array of users.

Objectives / Topics Covered	<ul style="list-style-type: none"> ● Planning Your Site ● Providing Feedback
Assignments / Labs	<ul style="list-style-type: none"> ● Version Control <ul style="list-style-type: none"> ○ Students create a running version document that tracks that changes they make to their website. Students learn how to catalog each version, and are asked to document how the website changes over time. ● Present your Innovation <ul style="list-style-type: none"> ○ Students are required to make a presentation highlighting the innovative website that they created, and how that site addresses a particular problem in their community. Students highlight how their website changed over the course of development, and as a result of feedback provided by user testing.

Supplemental Modules:

This course offers two supplemental modules, **HTML Bootcamp** and **JavaScript Bootcamp**. These modules are designed to refresh students on concepts in HTML/CSS and JavaScript that they have learned in previous courses. Each lesson is designed to last **45 minutes - 1 hour**.

The course materials for the two bootcamps are as follows:

HTML Bootcamp	JavaScript Bootcamp
Introduction to HTML	Hello World
Structure of an HTML page	Variables
Formatting Text	User Input
Links	Basic Math in JavaScript
Images	Booleans
HTML Lists	Logical Operators
HTML Tables	Comparison Operators
HTML Styling	If Statements
Introduction to CSS	For Loops in JavaScript
CSS Select by Tag	For Loop Practice
CSS Select by Class	Random Numbers
CSS Select by ID	While Loops
The Cascade	Loop and a Half
Multi-File Websites	Functions and Parameters 1
Divs	Functions and Parameters 2
Spans	Functions and Return Values 1
Combining CSS Selectors	Functions and Return Values 2
Special Selectors	Local Variables and Scope
Visibility	Intro to Lists/Arrays
Animation	Indexing Into an Array
Interaction	Adding/Removing From an Array
	Array Length and Looping