CD Studio: JavaScript
PSAM3210

Thursdays, 12:10 pm – 2:50 pm
Parsons 2 W 13th, Room 1201

Eric Li
lie@newschool.edu

Description

This is a computer science class first, and a design class second. We will focus on fundamentals of computing through the lens of JavaScript and its particulars before diving into the creation of a web based application using Vue. Students will learn about computing paradigms such as objects and arrays, functions, data structures, and algorithms. We will review what a modern JS stack looks like and how to call an API, with the final project being a web app that students will design and build. This is a fast paced course and students are expected to dedicate a good amount of time outside of class to assignments from this class.

Learning Outcomes

By the end of the semester, students should be able to:

- Understand contemporary paradigms of computer science including arrays, loops, functions, and objects
- Know how to read documentation and learn about new technical topics
- Proficiently read and implement JavaScript as part of their own work
- Know how to debug code effectively
- Use Vue to build a web-based application

Grading

10% Participation
15% Midterm Exam
30% Final Project
45% Assignments
- **5% A1**
- **10% A2**
- **10% A3 (Pair)**
- **10% A4 (Pair)**
- **10% A5 (Pair)**

You will be graded on correctness, code quality and style, and creativity. The assignments are structured in a way to give you practical, hands-on experience with the concepts we discuss in lecture and in class. This class will have a programming midterm exam which will evaluate your knowledge of concepts covered in class; the purpose of the exam is not meant to be a "gotcha" and I like to think that if you are paying attention in class, and doing the work, then you should be just fine.

Assignments 1 and 2 are individual assignments, whereas 3, 4 and 5 are pair assignments. Students have the option of working in pairs or alone. If students choose to work as pairs, they will submit one assignment and both students will share the same grade. Assignments will be evaluated based on correctness and code quality.

The final project will involve both design and software development. Students are expected to define a unique software that represents original thinking, sketch on and collect feedback on their designs, and finally build that software as a functioning website. They will be evaluated on completion, creativity, and quality of work. Like the latter assignments, students may again choose to work with a partner. If they do, then students will share the same grade.

Schedule

Week	**Sync**	**Async (after class)**
1 Sept 1	**Introduce** Hellos Syllabus Review Software Review **Lecture** What is JavaScript, anyways?	**Read** What is code? by Paul Ford
2 Sept 8	**Discuss** What is code? **Lecture** JavaScript Basics	**Homework** A1 Hello, World
3 Sept 15 **Due** A1	**Lecture** Conditionals and loops, debugging code	**Homework** A2 Loops

Week	Sync	Async (after class)
4 Sept 22 **Due** A2	**Lecture** Arrays and Objects	**Homework** A3 N-Body
5 Sept 29	**Lecture** Functions and classes	
6 Oct 6 **Due** A3	**Lecture** IO/DOM	**Homework** A4 Garden
7 Oct 13	**Lecture** Functions and Modules	
8 Oct 20 **Due** A4	**Lecture** Sorting and recursion	**Homework** A5 Sierpenski
9 Oct 27	**Lecture** A Modern JS Stack	**Homework** Final Project Proposals
10 Nov 3 **Due** A5	**Lecture** MVC and Vue **Q&A**	**Study**
11 Nov 10 **Due** Project proposals	**Midterm** **Lecture** Conditional Rendering and State	**Homework** Sketch
12 Nov 17 **Due** Sketches Due	**Present sketches** **Lecture** CRUD and Webservers	**Homework** Final Projects

Week	Sync	Async (after class)
13 Nov 24 (Thanksgiving)	**No Class**	**No Class**
14 Dec 1	**Work in class**	**Homework** Continue working on Final Projects
15 Dec 8	**Work in class**	**Homework** Continue working on Final Projects
16 Dec 15	**Final Reviews** Guest Crit: Stephanie Schapowal	Have a great winter break!

Assignments

A1: Hello, World

You'll complete a series of 3 quick exercises intended to familiarize yourself with JavaScript and its syntax. These exercises are more detailed in the assignment README, but they are as follows:

1. Hello World, the classic (1 pt)
2. Alice, meet Bob (2 pt)
3. RGB to CMYK (2 pt)

You will learn: Git, JavaScript, Variables, Coding style, String interpolation, Console

A2: Loops

Using loops, write code that accomplishes the following exercises:

1. Fibonacci, prints the first N fibonacci numbers (3 pt)
2. FizzBuzz, the classic programming prompt (3 pt)
3. Albers, draw the squares (4 pt)

You will learn: Data manipulation, Conditional logic, Loops, Arrays

A3: N-Body (Pair)

Implement an n-body simulation, which keeps track of the movements of the inner planets of our solar system. You'll be provided an initial set of values and be responsible for updating the position, velocity, and acceleration of the planets using linear approximation.

You may choose to complete this assignment individually or with a partner.

Kudos to Princeton's COS126 for the source material for this assignment.
You will learn: Loops, Arrays, JSON

A4: Garden (Pair)

Create a virtual garden which implements the ability to render and place elements where you click your mouse. Inspiration for this assignment is based on the demo for Put That There.

You may choose to complete this assignment individually or with a partner.

You will learn: DOM and JS Relationship, Functions, Modularizing code, Event listeners

A5: Sierpenski (Pair)

Implement a sierpenski triangle which renders in an HTML Canvas by recursively calling itself again and again. A user should be able to specify the level of detail (n) that the triangle renders at.

You may choose to complete this assignment individually or with a partner.

You will learn: Recursion, Canvas

Final Project

More and more, devices around us are clawing for our attention. Content creators around us are capitalizing on our attention, assigning a dollar value to our time. As designers, we must be mindful of how our designs can encourage, discourage, or run parallel to the established conventions of technology and the internet.

Laurel Schwulst writes, in *My website is a shifting house next to a river of knowledge* of the many shapes a website can take.

Design and develop a piece of software that lets **you** use the internet mindfully. In your project proposal, succinctly (in 200 words or less) describe the problem you are trying to solve, and your solution for it.

You may choose to work with a partner if you would like.

Following your proposal, you will explore 3 different directions for your design and then implement a singular refined direction of your software in code.

Your final project submission should be a link to the website and your project repository.

Policies

Grading, participation, attendance, academic honesty, and other CD program policies can be found here.

Inclusion

My intent is to respect and give forum to a range of perspectives and backgrounds, including culture, race, gender, sexual orientation, socioeconomic status, disability, and age. In instances where I am personally not qualified to speak from a specific perspective, students are encouraged to explore this area themselves. And please let me know if there are ways that the course can better serve these goals.

Engagement

Students are expected to actively and passionately participate in this course. This means more than showing up and turning things in on time. Beyond that baseline, students should be curious, prepared, thoughtful, vocal, and intentional throughout the course. They should make us understand why they are here, and demonstrate that they care about themselves, their work, and each other.

Office hours

I will have limited availability outside of our class time, and won't keep scheduled "office hours." Students should not rely on me to solve specific design or technical problems. Their first resource should be themselves, then the course site and its materials, and then each other. If there are still questions — such as logistical or content ones — students can message me on Slack, and I will respond when I can. But this should never be a bottleneck; all of this works better when not done at the last minute.

Additional technical help

For more specific technical instruction and questions, Parsons has dedicated CD-program tutors available to help students with HTML, CSS, and JavaScript, as well as offering general design critiques and feedback. The drop-in schedules are available in the CD@Parsons app under "Make & Remake."

As CD-program tutors are available only a limited number of hours per week, it is advisable to start on your projects early and seek help early to avoid the usual end of project/semester rush for additional help.

Code plagiarism

Students may find code similar to our exercises or projects elsewhere online. But the copying or adapting of any code beyond our provided course material (lectures, exercises, demos) without attribution is not allowed under any circumstances. If adapting, with attribution, students must explain the usage and demonstrate an understanding of how it works. There is zero tolerance for any sort of plagiarism — which ranges from “verbatim copying” (cutting-and-pasting code) to “thorough paraphrasing” (changing names or rearranging code). Students should also review the *Academic Honesty and Integrity* policies.

Pairing

On certain assignments you are allowed to pair with a partner. This is a form of programming called pair programming. If you choose to pair with a partner, you must both be present anytime the assignment is being worked on. One partner will "drive" and write the code, while the other is the "navigator" and will review the code and ask questions. You must swap roles every 30 minutes. If you decide to pair, let me know who you are partnering with over Slack prior to submitting the assignment.

Recording sessions

Screen recordings of our sessions will be made available for students to reference later. As these will include the students and their work, the recordings will be stored on Google Drive and made available only to New School email users.

Slack

We will be using Slack as our primary form of communication with each other. There are channels for general questions, as well as for each individual assignment and students are encouraged to use Slack as an open forum for communicating with each other and with myself. Students can also use the OpenSay app to anonymously ask questions.