

**Parsons School of Design**  
DESIGN AND TECHNOLOGY  
**WEB ADVANCED: JAVASCRIPT APIS**

PGTE 5505; CRN 2817

**Spring 2019**

Time: 7:00pm - 9:40pm.

Location: 6 East 16th Street, Room 1202

**Teacher:** Mani Nilchiani

**Contact:** nilchiam@newschool.edu

**Course Description:**

Web technologies have come a long way since their inception, and they are pushing the boundaries daily, becoming an increasingly dynamic & rich platform for development and delivery.

Powered by JavaScript and a host of related technologies, in this course we will focus on designing, developing and delivering modern & dynamic applications on the client as well as the server side.

With an eye towards best practices and industry standards, We will learn about what API's mean, how they can be useful & how to build & integrate with them to craft data-driven web applications.

**Overview:**

We are creative folks. We build things. The way we look at the tools for making is different than Computer Science / Engineering majors. Most of us didn't come from a programming background and suddenly dove in the deep end, learning to survival and accomplishing goals. It was a hell of an awesome way to kick things off.

But if you are already making things for the web, and want to make them better, or want to start getting serious about the web and its language, then we will nerd out on some core concepts, programming patterns, best practices and utility belts to get you off the ground to make things more maintainable, readable and hopefully smarter.

We will: go back to basics. Take a close look at JS. See the bright and dark sides. Demystify language concepts. Learn programming patterns. Look at example code. Write code. Workshop. Examine the landscape of modern JS development. See client and server side JS. Pick our tools, and explore the

browser as a bountiful platform. We will also have cool guests.

We do have a syllabus but we all know that one syllabus doesn't fit all. Once we hit the ground running, we will gauge things based on the dynamic of the class. We will play it by ear to set the pace, fine tune the curriculum, and personalize based on interest.

Based on what we discuss in class, everyone's invited to post good readings, articles and resources. We will discuss suggested readings from the sources listed below, during the class.

You DON'T have to read the suggested readings. But it is highly recommended. What you do have to do, is to submit the weekly small assignments, midterm and final projects.

**Learning Outcomes** By the successful completion of this course, students will be able to:

1. Get comfortable with JavaScript as a powerful programming language
2. Develop front-end applications using JavaScript
3. Familiarize with modular development
4. Familiarize with object oriented & functional JavaScript
5. Create an architecture for an application based on the requirements
6. Having learned the concepts of the language, be able to approach new libraries, frameworks and techniques with confidence and use them to their advantage
7. Work with Node.js to create JavaScript API endpoints on the server
8. Deploy web applications

**Assessable Tasks** The goal of the course is to work to the best of the students' ability and the grades will reflect that effort. The components of the final grade are:

Participation /Attendance	20%
Short Assignments	30%
Midterm Project	25%
Final Project	25%
<b>TOTAL</b>	<b>100%</b>

Participation /Attendance requires contributions in class and to the course forums, the day-to-day substance of the course, including but not limited to: developing ideas, defining projects, working with others in a timely and reliable way.

Short assignments mean that there will be a few assignments throughout the semester. These will usually be take home tasks in which students will need to develop a project based on the learning materials.

Mid-term Project In the first half of the semester, we will work towards gradually building an API client, with an eye towards Node.js for server-side application development.

Final Project In the second half of the semester, we will work towards building our own backend API's, and integrate it with our front-end.

WEEK 1	1/24/19	<b>Learn Concepts:</b> > Introduction, History, JS fundamentals part 1 (Data Types, Expressions, Statements, Control flow, iteration, constructors)  <b>Learn Tools:</b> > Chrome Dev Tools	<b>Assignment:</b>  Write an <code>`api`</code> function that returns an array of objects.  Iterate over this array and print all the properties of this array to the console. Bonus points if you manage to print this in HTML
WEEK 2	1/31/19	<b>Learn Concepts:</b> > JS fundamentals part 2 [Scope] > Introduction to DOM API [Querying elements, manipulating classes] > JSON > XHR requests > Ports > Basics of CSS Styling [Selectors, properties, values] > Synchrony & Asynchrony > The <code>`window`</code> object > Using 3rd Party libraries part 1 [Where to find code, the <code>`script`</code> tag] > Quick introduction to ES2015 and TC39 [Why & How]  <b>Learn Tools:</b> > Unix command line part 1 [ <code>`ls`</code> , <code>`.bash_profile`</code> , <code>`\$PATH`</code> variable, <code>`touch`</code> , <code>`man`</code> , <code>`mkdir`</code> , <code>`rm`</code> , <code>`mv`</code> , <code>`cp`</code> , <code>`find`</code> , <code>`cd`</code> ]	<b>Assignment:</b>  <i>Part One:</i> > Load in the provided JSON file > Parse the JSON file > Iterate over the JSON file and create HTML elements representing the data in the JSON data structure > Bonus points for tasteful CSS Styling  <i>Part Two:</i> Find an API that you'd like to work with. You'll work with this API in your next assignment. Example: <a href="#">Giphy API</a>

		<ul style="list-style-type: none"> <li>&gt; Install Homebrew</li> <li>&gt; Local HTTP servers</li> <li>&gt; Setup git, setup github &amp; ssh</li> <li>&gt; Markdown</li> <li>&gt; <code>`.gitignore`</code></li> </ul> <p><b>Class activity:</b></p> <ul style="list-style-type: none"> <li>&gt; Consume the <code>`api`</code> function we wrote last week and output data on an html page. Style it.</li> <li>&gt; Use constructor functions to create objects</li> </ul>	
WEEK 3	2/7/19	<p><b>Learn Concepts:</b></p> <ul style="list-style-type: none"> <li>&gt; Introduction to DOM API part 2 [Interactivity]</li> <li>&gt; Forms &amp; Inputs part 1 [text inputs]</li> <li>&gt; Using ES2015 classes</li> <li>&gt; CRUD</li> <li>&gt; REST API's</li> <li>&gt; HTTP Codes</li> <li>&gt; CORS (Cross Origin Resource Sharing)</li> <li>&gt; Anatomy of a URL [parameters]</li> <li>&gt; Modular code organization: (IIFE, CommonJS, ES Modules)</li> <li>&gt; Dependency Management</li> </ul> <p><b>Class Activity:</b></p> <ul style="list-style-type: none"> <li>&gt; We will write a simple client for <a href="#">Dad Jokes API</a></li> <li>&gt; We will look at how to authenticate with APIs <a href="#">using Giphy API as an example.</a></li> </ul> <p><b>Learn Tools:</b></p> <ul style="list-style-type: none"> <li>&gt; Chrome Dev Tools Networks Panel</li> <li>&gt; Unix command line part 2 [<code>`curl`</code>, <code>`&gt;`</code>, <code>`&gt;&gt;`</code>, <code>` `</code>, <code>`grep`</code>]</li> <li>&gt; Postman</li> <li>&gt; Setup Node.JS via <code>`npm`</code></li> <li>&gt; Dependency Management via <code>`npm`</code></li> </ul>	<p><b>Assignment:</b></p> <ul style="list-style-type: none"> <li>&gt; Make HTTP GET request calls to the API you have chosen</li> <li>&gt; Parse the response</li> <li>&gt; Use the data to create and output HTML elements on the page.</li> <li>&gt; Bonus points for tasteful styling</li> </ul>
WEEK 4	2/14/19	<b>Learn Concepts:</b>	<b>Assignment:</b>

		<ul style="list-style-type: none"> <li>&gt; Anatomy of Web Applications, the need for frameworks</li> <li>&gt; MV* patterns</li> <li>&gt; Forms and Inputs part 2 [binary data]</li> <li>&gt; Promises API</li> <li>&gt; Thinking in Components</li> <li>&gt; React part 1 [Setup, Core concepts, JSX, state]</li> </ul> <p><b>Learn Tools:</b></p> <ul style="list-style-type: none"> <li>&gt; Babel</li> <li>&gt; Webpack</li> <li>&gt; React Dev Tools</li> </ul> <p><b>Class Activity:</b></p> <ul style="list-style-type: none"> <li>&gt; We'll write some React code together</li> </ul>	<ul style="list-style-type: none"> <li>&gt; Refine your previous homework</li> <li>&gt; Draw a diagram of how you might think about your client application in terms of components.</li> <li>&gt; Propose a draft of your midterm project. You can continue building on top of the current assignment too.</li> <li>&gt; You can collaborate on your midterm projects. Teams should not be larger than 3 people, and the division of responsibilities should be outlined in the proposal.</li> <li>&gt; Submit the proposal in Markdown format on Github.</li> </ul>
WEEK 5	2/21/19	<p><b>Guest Lecture:</b></p> <ul style="list-style-type: none"> <li>&gt; Alex Staudt [AKA <a href="#">FuzzyWobble</a>] Interaction designer, creative coder &amp; Digital Alchemist at <a href="#">IDEO</a>.</li> </ul> <p><b>Learn Concepts:</b></p> <ul style="list-style-type: none"> <li>&gt; React part 2 [Functional components, props]</li> </ul> <p><b>Class Activity:</b></p> <ul style="list-style-type: none"> <li>&gt; We'll write some React code together</li> </ul>	<p><b>Assignment:</b></p> <ul style="list-style-type: none"> <li>&gt; Re-write your previous assignment in React. Refer to the component diagram you drew last session.</li> </ul>
WEEK 6	2/28/19	Workshop Time	<p><b>Assignment:</b></p> <p>Work on the Midterm project</p>
WEEK 7	3/7/19	Mid term presentations	<p><b>Assignment:</b></p> <p>No assignment. Take this time to reflect and ask me questions. We have covered a lot</p>

			of ground and it's normal to feel overwhelmed / lost. It's OK!
WEEK 8	3/14/19	<b>Learn Concepts:</b> > React part 3 [Forms, Routing]  <b>Learn Tools:</b> > NPM Scripts  <b>Class Activity:</b> > We'll write some React code together	<b>Assignment:</b> I will provide you with an API that you can use to build a client for.
WEEK 9	3/21	<b>Spring Break. No class.</b>	
WEEK 10	3/28	<b>Learn Concepts:</b> > Introduction to CLI (Command Line Interface) > JS on the backend: Introduction to Node.js > Introduction to Databases > Environment variables  <b>Class Activity:</b> > We'll write a simple Node.JS CLI tool together in class. > We'll make a MongoDB database and explore it	<b>Assignment:</b> Write a Node.JS advice generator CLI tool. (Do you have cooler CLI ideas? That's even better!)
WEEK 11	4/4	<b>Learn Concepts:</b> > Writing a Node.JS HTTP Server > Node.js `http` and `fs` modules > Express.JS > Define API routes > More MongoDB  <b>Class Activity:</b> > We'll write a simple Node.js server together > We'll explore Express.js and its documentation  <b>Learn Tools:</b> > nodemon > MongoDB CLI (Install MongoDB)	<b>Assignment:</b> Write a Node.js server (With or without Express) that reads jokes from a JSON file and can: > Serve a random joke > Serve a specific joke by it's ID I will provide the frontend client code to you.  Start thinking about your Final Project.
WEEK 12	4/11	<b>Learn Concepts:</b> > RESTful development with Express.js	<b>Assignment:</b> Write a Node.js server with Express

		<p>&gt; Integrating node.js API with MongoDB</p> <p><b>Learn a thing:</b></p> <p>&gt; Deployment and related tools</p>	<p>and MongoDB that reads &amp; writes from and to the database. Your API should have routes for:</p> <ul style="list-style-type: none"> <li>&gt; Serving a random joke</li> <li>&gt; Serving a specific joke by it's ID</li> <li>&gt; Writing a Joke to the database.</li> </ul> <p>I will provide the frontend client code to you.</p> <p>Submit your Final Project proposal in Markdown format.</p>
WEEK 13	4/18	<p><b>Guest Lecture:</b></p> <p>&gt; <a href="#">Angie Meitzler</a>, Artist, Programmer, program coordinator at <a href="#">Pioneer Works</a> Technology Lab.</p> <p>Workshop Time</p>	<p><b>Assignment:</b></p> <p>Continue working on the Final Project.</p>
WEEK 14	4/25	Workshop Time	<p><b>Assignment:</b></p> <p>Continue working on the final project.</p>
WEEK 15	5/2	<p>Final presentations!</p> <p>[Guest crits TBA]</p>	<p><b>Assignment:</b></p> <p>Enjoy the show!</p>

### Recommended Reading

1. Eloquent JavaScript 3<sup>rd</sup> edition: <http://eloquentjavascript.net/>
2. You Might Not Need JQuery: <http://youmightnotneedjquery.com/>
3. ECMAScript Guide: <https://flaviocopes.com/ecmascript/>
4. Javascript Design Patterns: <https://addyosmani.com/resources/essentialjsdesignpatterns/book/>
5. Additional handouts, downloads, and resources will be provided where required.
6. YDKJS series <https://github.com/getify/You-Dont-Know-JS>

## Resources

The university provides many resources to help students achieve academic and artistic excellence. These resources include:

- The University (and associated) Libraries: <http://library.newschool.edu>
- The University Learning Center:  
<http://www.newschool.edu/learning-center>
- University Disabilities Service:  
[www.newschool.edu/student-disability-services/](http://www.newschool.edu/student-disability-services/)

In keeping with the university's policy of providing equal access for students with disabilities, any student with a disability who needs academic accommodations is welcome to meet with me privately. All conversations will be kept confidential. Students requesting any accommodations will also need to contact Student Disability Service (SDS). SDS will conduct an intake and, if appropriate, the Director will provide an academic accommodation notification letter for you to bring to me. At that point, I will review the letter with you and discuss these accommodations in relation to this course.

## Making Center

The Making Center is a constellation of shops, labs, and open workspaces that are situated across the New School to help students express their ideas in a variety of materials and methods. We have resources to help support woodworking, metalworking, ceramics and pottery work, photography and film, textiles, printmaking, 3D printing, manual and CNC machining, and more. A staff of technicians and student workers provide expertise and maintain the different shops and labs. Safety is a primary concern, so each area has policies for access, training, and etiquette that students and faculty should be familiar with. Many areas require specific orientations or trainings before access is granted. Detailed information about the resources available, as well as schedules, trainings, and policies can be found at [resources.parsons.edu](http://resources.parsons.edu).

## Grading Standards

- A Work of exceptional quality
- A- Work of high quality
- B+ Very good work
- B Good work; satisfies course requirements
- Satisfactory completion of a course is considered to be a grade of B or higher.*
- B- Below-average work
- C+ Less than adequate work
- C Well below average work
- C- Poor work; lowest possible passing grade
- F Failure
- GM Grade missing for an individual



*Grades of D are not used in graduate level courses.*

#### **Grade of W**

The grade of W may be issued by the Office of the Registrar to a student who officially withdraws from a course within the applicable deadline. There is no academic penalty, but the grade will appear on the student transcript. A grade of W may also be issued by an instructor to a graduate student (except at Parsons and Mannes) who has not completed course requirements nor arranged for an Incomplete.

#### **Grade of Z**

The grade of Z is issued by an instructor to a student who has not attended or not completed all required work in a course but did not officially withdraw before the withdrawal deadline. It differs from an "F," which would indicate that the student technically completed requirements but that the level of work did not qualify for a passing grade.

#### **Grades of Incomplete**

The grade of I, or temporary incomplete, may be granted to a student under unusual and extenuating circumstances, such as when the student's academic life is interrupted by a medical or personal emergency. This mark is not given automatically but only upon the student's request and at the discretion of the instructor. A Request for Incomplete form must be completed and signed by student and instructor. The time allowed for completion of the work and removal of the "I" mark will be set by the instructor with the following limitations:

Work must be completed no later than the seventh week of the following fall semester for spring or summer term incompletes and no later than the seventh week of the following spring semester for fall term incompletes. Grades of "I" not revised in the prescribed time will be recorded as a final grade of "F" by the Office of the Registrar.

#### **Divisional, Program and Class Policies**

##### ● Responsibility

Students are responsible for all assignments, even if they are absent. Late assignments, failure to complete the assignments for class discussion and/or critique, and lack of preparedness for in-class discussions, presentations and/or critiques will jeopardize your successful completion of this course.

##### ● Participation

Class participation is an essential part of class and includes: keeping up with reading, assignments, projects, contributing meaningfully to class discussions, active participation in group work, and coming to class regularly and on time.

### ● Attendance

Parsons' attendance guidelines were developed to encourage students' success in all aspects of their academic programs. Full participation is essential to the successful completion of coursework and enhances the quality of the educational experience for all, particularly in courses where group work is integral; thus, Parsons promotes high levels of attendance. Students are expected to attend classes regularly and promptly and in compliance with the standards stated in this course syllabus.

While attendance is just one aspect of active participation, absence from a significant portion of class time may prevent the successful attainment of course objectives. A significant portion of class time is generally defined as the equivalent of three weeks, or 20%, of class time. Lateness or early departure from class may be recorded as one full absence. Students may be asked to withdraw from a course if habitual absenteeism or tardiness has a negative impact on the class environment.

Whether the course is a lecture, seminar or studio, faculty will assess each student's performance against all of the assessment criteria in determining the student's final grade.

### ● Canvas

Use of Canvas may be an important resource for this class. Students should check it for announcements before coming to class each week.

### ● Delays

In rare instances, I may be delayed arriving to class. If I have not arrived by the time class is scheduled to start, you must wait a minimum of thirty minutes for my arrival. In the event that I will miss class entirely, a sign will be posted at the classroom and/or on canvas indicating your assignment for the next class meeting.

### ● Electronic Devices

The use of electronic devices (phones, tablets, laptops, cameras, etc.) is permitted when the device is being used in relation to the course's work. All other uses are prohibited in the classroom and devices should be turned off before class starts.

### ● Academic Honesty and Integrity

Compromising your academic integrity may lead to serious consequences, including (but not limited to) one or more of the following: failure of the assignment, failure of the course, academic warning, disciplinary probation, suspension from the university, or dismissal from the university.

Students are responsible for understanding the University's policy on academic honesty and integrity and must make use of proper citations of sources for writing papers, creating, presenting, and performing their work, taking examinations, and doing research. It is the responsibility of students to learn the procedures specific to their discipline for correctly and appropriately differentiating their own work from that of others. The full text of the policy, including adjudication procedures, is found at

<http://www.newschool.edu/policies/#> Resources regarding what plagiarism is and how to avoid it can be found on the Learning Center's website:  
<http://www.newschool.edu/university-learning-center/student-resources/>

The New School views "academic honesty and integrity" as the duty of every member of an academic community to claim authorship for his or her own work and only for that work, and to recognize the contributions of others accurately and completely. This obligation is fundamental to the integrity of intellectual debate, and creative and academic pursuits. Academic honesty and integrity includes accurate use of quotations, as well as appropriate and explicit citation of sources in instances of paraphrasing and describing ideas, or reporting on research findings or any aspect of the work of others (including that of faculty members and other students). Academic dishonesty results from infractions of this "accurate use". The standards of academic honesty and integrity, and citation of sources, apply to all forms of academic work, including submissions of drafts of final papers or projects. All members of the University community are expected to conduct themselves in accord with the standards of academic honesty and integrity. Please see the complete policy in the Parsons Catalog.

#### ● Intellectual Property Rights

The New School (the "university") seeks to encourage creativity and invention among its faculty members and students. In doing so, the University affirms its traditional commitment to the personal ownership by its faculty members and students of Intellectual Property Rights in works they create. The complete policy governing Intellectual Property Rights may be seen on the [university website, on the Provost's page](#).