COMP-496: Ethical Web Development

Cybersecurity & Networking Seminar

> It is difficult to get a man to understand something when his salary depends upon his not understanding it.
>
> —Upton Sinclair, *Oakland Tribune*, December 11, 1934

> The way to right wrongs is to turn the light of truth upon them.
>
> —Ida B. Wells

Institution
Olivet Nazarene University

School
Martin D. Walker School of Engineering and Technology

Department
Computer Science and Emerging Technologies

Term
Spring 2020

Meeting Location
Reed Hall of Science 212

Meeting Time
Tuesdays 5:30 p.m.–8:00 p.m.

Credit Hours
2

Instructor
Rev. Reuben L. Lillie, MM, MDiv

Office
Reed Hall of Science 324A

Office Hours
Tuesdays 10:00 a.m.–12:00 p.m. or by appointment

1. Course Description

This course introduces modern best practices for developing a more humane web. Students will explore the ethical ramifications behind various software decisions, including web performance, security, and user/developer experience. The course simulates a team-based working environment in which students will build individual portfolios exhibiting job-ready, front-end skills. In the process, students will learn to leverage modern JavaScript, CSS, Markdown, Git, intrinsic design principles, and static site generation. No prior coding experience necessary.

2. Course Rationale

In keeping with Olivet's mission, this course provides a vital space for the University to explore "web development with a Christian purpose" together. Other courses in the curriculum are designed specifically for advanced computer science students who have met several prerequisites. By lowering the barrier to entry, however, this course invites students from a variety of fields to learn with and from one another. Furthermore, by focusing on the ethical ramifications behind various software decisions, students may better advocate for justice along their respective career paths.

3. Course Aims and Outcomes

3.1. Course Aims

The primary goal of this course is to explore the question, *How should we make the Web?* Too many who use the Internet know too little about how it works. Worse, many who do know how the Internet works take advantage of those who know less—and with little personal consequence. Profiteering, surveillance, and malicious intent abound online. As aspiring computer scientists, entrepreneurs, business leaders, content creators, and users of the Web in general, what can we do to ensure the otherwise worthwhile opportunities the Web provides do not simultaneously encourage the violation of users' privacy, safety, or other rights?

In addition to acquiring and improving upon basic coding skills, students will shape a personal philosophy of web development. With these skills and principles in hand, students will be better prepared to lead projects and companies in making the Web a more just space. None of us has any business learning to make the Web if we are not learning to make it better.

3.2. Specific Learning Outcomes

By the end of this course, students will:

- Write semantic, accessible, and performant HTML, CSS, and JavaScript
- Leverage distributed version control systems and static site generators
- Apply principles of Jamstack, mobile-first, and documentation-driven development
- Compare licensing practices for creative works
- Contribute to a class-led open source software project
- Build a personal online portfolio

4. Format and Procedures

4.1. Class Session Format

Each two-and-a-half-hour class session comprises five thirty-minute timetables: twenty-five minutes of work followed by a five-minute break. Each timetable is dedicated either to a brief lecture, discussion topic, tutorial, in-class project, or other task. By allowing for these periodic breaks, the instructor otherwise expects students' full participation and undivided attention.

The first half of the semester focuses on basic language training and design principles. After Spring Break, the semester shifts to a whole-class, team-based software project.

4.2. Electronic Equipment Policy

Except in emergencies, students should silence and refrain from using mobile phones, personal computers, tablets, and other electronic devices without the professor's invitation during class. While in session, workstations may only be used for material related to this course. In turn, the professor's workstation can observe, gain access to, and project the screen contents for each student's workstation. The professor reserves the right to enforce these guidelines at any time.

4.3. Assignment Policy

4.3.1. Submission Process

Unless otherwise instructed, students will follow a two-step process for submitting code samples for grading.

1. Push code sample assignments to the relevant GitLab project repository
2. Post a link to the GitLab commit in the corresponding assignment in Canvas

4.3.2. Style Guidelines

Each student is responsible for submitting grammatically correct work. All written assignments must follow the *Chicago Manual of Style* and *Merriam-Webster's Collegiate Dictionary* as well as any templates and supplements distributed by the professor for style, formatting, grammar, mechanics, and spelling. In turn, all code samples and issue posts must follow the relevant coding standards and collaboration guidelines in the course GitLab project repository. Students are expected to use gender-inclusive, gender-neutral, and other inclusive forms of language whether speaking or writing. When in doubt, students should consult the professor.

4.3.3. Late Assignments

Any assignment a student fails to submit by the time it is due will automatically receive a grade reduction of fifty percentage points. Late assignments will be reduced an additional ten percent for each successive day that assignment is late. No assignment will be accepted after 4:00 p.m. on the final day of classes. Graduating seniors must likewise make arrangements to submit all work as instructed by the Office of the Registrar.

5. Professor's Assumptions

While this course does not require any prior coding experience, the professor assumes each student, by registering for this course, (a) comes motivated to learn and improve upon all skills necessary throughout the semester; and (b) is comfortable making and troubleshooting mistakes in class. To that end, students should be prepared to allocate four to eight hours per week outside of class to complete readings, tutorials, and other assignments, as is typical of a two-credit-hour course. The work for a given session may not require that entire time allotment. The rapid pace and scope of this course, however, may require a student to complete additional research and tutorials outside of class. In any case, the professor aims to create a charitable and justifiably challenging learning environment for each student. Programming is hard! Students are welcome to work ahead. And students should not hesitate to ask the professor and other classmates for assistance.

6. Course Requirements

Students will:

- Attend all class sessions
- Read, listen to, and view all written, audio, and video assignments
- Participate fully in class discussions
- Complete all assigned in- and out-of-class tutorials
- Complete all portfolio assignments

6.1. Course Materials

6.1.1. Supplies

Students are expected to bring either (a) paper and a writing utensil or (b) a tablet and stylus to class.

Although desktop and laptop computers are useful machines, they typically are not suitable for spatial reasoning and higher-order thinking, particularly the facilities needed in the planning phases of software development. Furthermore, the limited desktop space around classroom workstations is not conducive to other electronic equipment.

6.1.2. Accounts and Subscriptions

In lieu of a formal textbook, students are required to purchase the complete set of “Vanilla JavaScript Pocket Guides” by Chris Ferdinandi (https://vanillajsguides.com/complete-set/). The professor has arranged for a special discount code with the author. Prior to purchase, students must contact the professor directly to receive the discount.

Students must also maintain a personal GitLab account for completing assignments (https://gitlab.com/). Those students who do not have a GitLab account will create one during the first class session.

6.1.3. Other Required and Supplemental Materials

All other required readings, audio files, or videos are free. Some publishers may require e-mail validation from the student before granting access.

The professor will include external URLs and other bibliographic information for all required and supplemental online materials in the references section of this syllabus or in Canvas modules for the relevant class session.

6.1.4. Recommended Textbooks

For those interested in more textbook-like resources, students are encouraged to purchase the following.

Falbe, Tina, Kim Anderson, and Martin Michael Fredricksen. *The Ethical Design Handbook*. Freiburg, Germany: Smashing Media, 2020.

Robbins, Jennifer N. *Learning Web Development: A Beginner's Guide to HTLM, CSS, JavaScript, and Web Graphics*, 5th ed. Sebastopol, CA: O'Reilly, 2018.

6.2. Attendance Policy

Students should arrive promptly for each class session, sign in to a workstation, and be prepared to follow the professor's instructions at the start of class.

Bearing in mind that this course meets once per week and for a total of thirteen class sessions, the following guidelines for absences apply in accordance with the University's Class Attendance Policy.[1]

- Each unexcused absence will automatically result in a loss of ten percentage points from the student's final grade.
- At the professor's discretion and in consultation with the department chair and Vice President of Academic Affairs, a student with three absences may be disqualified from receiving credit for the course.
- A student with more than three unexcused absences will definitely be disqualified from receiving credit.
- Excused absences are not penalized, but students must report them when they occur in order to be excused.
- Students receiving educational leniency for an absence will be marked as present.
- Students missing class for any reason remain personally responsible for submitting all assignments on time.

6.3. Participation Policy

Participation accounts for at least fifteen percent of the student's final grade. Although students mostly will be seated at workstations during class, each student should be prepared, nonetheless, to respond appropriately to any and all questions and requests from the professor as well as any discussion prompts and other in-class assignments. The nature of the material further requires students to devise creative solutions to inexact problems. The professor, therefore, reserves the right to award or reduce any student's final grade based on the quality of that student's in-class participation throughout the term.

6.4. Vanilla JavaScript Projects

Students must complete all eight projects from Chris Ferdinandi's "Vanilla JavaScript Pocket Guides." Students also must complete a minimum of nine of the project ideas listed at https://learnvanillajs.com/projects/.

For extra credit, students may complete as many as eleven additional projects from the idea list (a maximum of two per week following Spring Break). Each extra credit project is eligible for up to five points (i.e., eleven projects at two points each equals a maximum of fifty-five points of extra credit, or 5.5% of the final grade). Note that the "Accordion" project idea is ineligible since it is already included among the Pocket Guides projects.

6.5. Portfolio Project

Students will contribute to a personal portfolio project throughout the term. The professor will grade contributions individually as they are made. All other portions of the portfolio will be assessed at the end of the term.

7. Grading Procedures

The professor will maintain a record of attendance and scholarship for each student. A report of the student's class standing is given at the close of the semester. The following percentages are used to calculate final letter grades.

Table 1. Final Grading Scale

Percentage Range	Letter Grade
93–100%	A
90–92%	A-
86–89%	B+
83–85%	B
80–82%	B-
76–79%	C+
73–75%	C
70–72%	C-
66–69%	D+
63–65%	D
60–62%	D-
0–59%	F

The professor will distribute a rubric for each assignment as relevant material is introduced. Assignments will be eligible for the following point values.

Table 2. Point Distribution for Assessment

Graded Item	Point Value
Hello World!	30
Vanilla JS Pocket Guide Projects (8 total)	80
Philosophy of Web Development—Rough Draft	75
Learn Vanilla JS Projects (9 minimum)	90
In-Class Project	150
Philosophy of Web Development—Final Draft	75
Portfolio Project	350
Participation	150
Total	**1000**

8. Statement of Academic Integrity

Seeking after truth is at the heart of an education at a Christian university like Olivet. ONU expects students to be truthful in all areas of life, including the academic arena. Those who engage in any form of academic dishonesty value their own gain more than their desire to seek truth; consequently, their behavior is incompatible with the goals and objectives of the University. Such dishonesty takes the form of cheating, plagiarism, or falsification. Specific examples include, but are not limited to:

- Submitting another’s work as one’s own or allowing others to submit one’s work as though it were theirs.
- Failing to properly acknowledge authorities quoted, cited, or consulted in the preparation of oral or written work. All work submitted by students must represent their original work. Outside sources used as references should reveal the name and source and the extent to which the source is used.
- Tampering with experimental data to obtain a “desired” result or creating results for experiments not conducted.
- Lying or otherwise deceiving university personnel about academic matters.
- Falsifying college records, forms, or other documents.
- Students who knowingly assist another in dishonest behavior are held equally responsible.

> —University *Catalog*[2]

9. Learning Support Services

> Olivet Nazarene University welcomes students with disabilities and is committed to complying with laws regarding equal opportunity for students with disabilities. Qualified individuals who have a disability will not be excluded from participation in, denied the benefits of, or be subjected to discrimination in connection with the services, programs, or activities offered by ONU. These prohibitions against discrimination apply not only to the University's academic programs but also to other programs and services at the University such as financial aid, housing, student employment, athletics, counseling and placement services, and extracurricular opportunities.
>
> The Office of Learning Support Services at Olivet Nazarene University is dedicated to supporting the University's efforts toward accessibility and inclusion for students with disabilities. This is accomplished through collaboration, advocacy, education, and accommodation. The Office of Learning Support Services works with students on an individual, case-by-case basis determining appropriate, reasonable accommodations to promote the student's learning, growth, development, and success. Those with questions should contact the office of Learning Support Services at lss@olivet.edu.
>
> —University *Catalog*[3]

10. Tentative Schedule

10.1. Session 1

- Hypertext Markup Language (HTML) and Accessibility
- Cascading Style Sheets (CSS) Syntax, Cascade, and Box Model
- Free and Open Source Software (FOSS)
- Git: Command Line Interface (CLI) and GitLab

10.2. Session 2

- Git: Issues and Documentation-Driven Development
- CSS: Styling Text
- HTML: Multimedia and Hypertext Transfer Protocol (HTTP)
- CSS: Styling Images
- HTML: Tables and Data Lists

10.3. Session 3

- HTML and CSS Documentation
- CSS: Layouts, Intrinsic Design, and Mobile-First Development
- CSS: Custom Properties and Object-Oriented CSS (OOCSS)
- HTML: Forms and Data Privacy
- Debugging HTML and CSS

10.4. Session 4

- JavaScript: Document Object Model (DOM) Manipulation
- JavaScript: Strings and Numbers
- JavaScript: Arrays and Objects
- JavaScript: Variables, Functions, and Scope
- JavaScript: DOM Injection and Traversal

10.5. Session 5

- Browser Storage and the General Data Protection Regulation (guest speaker): Paul D. McGrady, Taft Stettinius & Hollister, LLP
- JavaScript: Documentation, Debugging, and Refactoring
- JavaScript: Application Programming Interfaces (API) and JavaScript Object Notation (JSON)
- Jamstack: Node.js and Eleventy Static Site Generator
- Jamstack: Markdown and Front Matter

10.6. Session 6

- Modern Agile (guest speaker): Tim Ottinger, Industrial Logic
- JavaScript: Plugins and Code Reuse
- JavaScript: Web Apps and Working with Stateful Data
- Jamstack: Organizing the Code Base
- Transitioning to In-Class Project

10.7. Session 7

- Cybersecurity (guest speaker): Prof. Jeffrey Rice, Program Director for Cybersecurity
- Favicons
- Metadata

10.8. Session 8

- Humane Documentation
- Web Typography

10.9. Session 9

- Environment Variables
- Internationalization (i18n) and Localization (L10n)

10.10. Session 10

- Scalable Vector Graphics (SVG)

10.11. Session 11

- Digging Deeper into CSS Grid

10.12. Session 12

- Progressive Web Apps

10.13. Session 13

- The Indie Web

10.14. Final Session

- Portfolio Project Presentations

11. Notes

1. ^ Olivet Nazarene University, “Class Attendance Policy,” in *Catalog* 2019–2020, http://catalog.olivet.edu/content.php?catoid=7&navoid=408#Class_Attendance_Policy.
2. ^ Olivet, “Statement of Academic Integrity,” in *Catalog*, http://catalog.olivet.edu/content.php?catoid=7&navoid=408#academic-integrity.
3. ^ Olivet, “Learning Support Services,” in *Catalog*, http://catalog.olivet.edu/content.php?catoid=7&navoid=408#learning_support_services.

12. References

12.1. The Topic of Ethics in Software and Web Development

Burns, Heather. 2018. “How GDPR Will Change The Way You Develop.” *Smashing Magazine*, February 27. https://www.smashingmagazine.com/2018/02/gdpr-for-web-developers/.

Center for Humane Technology. n.d. “Center for Humane Technology: Realigning Technology with Humanity.” https://humanetech.com/.

Criado Perez, Caroline. 2019. *Invisible Women: Data Bias in a World Designed for Men*. New York: Abrams

Eubanks, Virginia. 2018. *Automating Inequality: How High-Tech Tools Profile, Police, and Punish the Poor*. New York: St. Martins.

Falbe, Tina, Kim Anderson, and Martin Michael Fredricksen. 2020. *The Ethical Design Handbook*. Freiburg, Germany: Smashing Media.

Ferdinandi, Chris. 2019. *The Lean Web*. Self-published e-book. https://leanweb.dev/ebook/.

Fry, Hannah. 2018. *Hello World: Being Human in the Age of Algorithms*. New York: W. W. Norton.

Kadlec, Tim. 2019. “The Ethics of Web Performance.” January 9. https://timkadlec.com/remembers/2019-01-09-the-ethics-of-performance/.

Keith, Jeremy. 2017. *Resilient Web Design*. Self-published e-book. https://resilientwebdesign.com/.

King, Kat. 2018. “Building Empathy-Driven Documentation.” Presentation at *Write the Docs Portland 2018, Crystal Ballroom, Portland, Oregon, May 7. Video, 28:01. https://www.youtube.com/watch?v=_HcmFvxxKaQ.*

Linders, Ben. 2018. “Why Software Developers Should Take Ethics into Consideration.” *InfoQ*, March 8. https://www.infoq.com/news/2018/03/software-developers-ethics/.

Noble, Safia Umoja and Brendesha M. Tynes, eds. 2016. *The Intersectional Internet: Race, Sex, Class, and Culture Online*. New York: Peter Lang.

Noble, Safia Umoja. 2018. *Algorithms of Oppression: How Search Engines Reinforce Racism*. New York: New York University.

Pomerantz, Jeffrey. 2015. *Metadata*. Cambridge, MA: MIT.

Rand-Hendriksen, Morten. 2018. “Using Ethics In Web Design.” *Smashing Magazine*, March 22. https://www.smashingmagazine.com/2018/03/using-ethics-in-web-design/.

Scott, Adam. 2017. “Adam Scott on Ethical Web Development.” Interview by Jeff Bleiel. *O’Reilly Programming Podcast*. August 24. Audio, 19:52. https://www.oreilly.com/ideas/adam-scott-on-ethical-web-development/.

———. 2016–17. *Ethical Web Development*, four volumes. Sebastopol, CA: O’Reilly. https://ethicalweb.org/.

Simonite, Tom. 2018. “Should Data Scientists Adhere to a Hippocratic Oath?” *Wired*, February 8. https://www.wired.com/story/should-data-scientists-adhere-to-a-hippocratic-oath/.

Waldman, Ari Ezra. 2018. *Privacy as Trust: Information Privacy for an Information Age*. New York: Cambridge.

Vallor, Shannon. 2018. *Technology and the Virtues: A Philosophical Guide to a Future Worth Wanting*. New York: Oxford.

Zuboff, Shoshana. 2019. *The Age of Surveillance Capitalism: The Fight for a Human Future at the New Frontier of Power*. New York: Hatchette.

12.2. Web Languages and Methodologies

Listed alphabetically

12.2.1. Accessibility (a11y)

Accessible Rich Internet Applications Working Group. 2019. “(WAI-ARIA) Authoring Practices 1.1.” World Wide Web Consortium (W3C). https://www.w3.org/TR/wai-aria-practices/.

The Accessibility Project. 2020. “The A11Y Project: A community-driven effort to make web accessibility easier.” https://a11yproject.com/.

W3C. 2020. “Web Accessibility Initiative (WAI).” https://www.w3.org/WAI/.

WebAIM. 2020. “WebAIM: Web Accessibility in Mind.” https://webaim.org/.

12.2.2. Agile Software Development

Kerievsky, Joshua. 2016. “An Introduction to Modern Agile.” *InfoQ*, October 20. https://www.infoq.com/articles/modern-agile-intro/.

“Modern Agile.” 2020. http://www.modernagile.org/.

Ottinger, Tim. 2018. “BONUS: Tim Ottinger on Practical Ways to Enable Psychological Safety: The Trust Transaction,” Interview by Vasco Duarte. *Scrum Master Toolbox Podcast*, May 5. Audio, 41:58. https://scrum-master-toolbox.org/2018/05/podcast/bonus-tim-ottinger-on-practical-ways-to-enable-psychological-safety-the-trust-transaction/#more-3551.

———. 2017. “Make People Awesome? Give Them Superpowers!” *Agile Otter Blog*, March 12. https://agileotter.blogspot.com/2017/03/make-people-awesome-give-them.html.

12.2.3. CSS (Cascading Style Sheets)

MDN. 2020. “CSS: Cascading Style Sheets.” https://developer.mozilla.org/en-US/docs/Web/CSS.

Leatherman, Zach. 2017. “A Comprehensive Guide to Font Loading Strategies.” *Non-Canonical Web Standards Fan Fiction*, August 14. https://www.zachleat.com/web/comprehensive-webfonts/.

Simmons, Jen. 2019. “Layout Land.” YouTube channel. https://www.youtube.com/layoutland.

Simmons, Jen, and Roger Black. 2017. “Web Typography & Layout: Past, Present, and Future.” Interview by Jeffrey Zeldman. *A List Apart*, December 6. Video, 1:13:10. https://alistapart.com/event/web-typography-layout-past-present-future/.

12.2.4. Git

Chacon, Scott and Ben Straub. 2014. *Pro Git*, 2nd ed. New York: Apress. https://git-scm.com/book/en/v2.

Schafer, Corey. 2019. “Git Tutorials.” YouTube playlist, six videos, 2:24:44. https://www.youtube.com/playlist?list=PL-osiE80TeTuRUfjRe54Eea17-YfnOOAx.

12.2.5. HTML (Hypertext Markup Language)

Bernard, Philippe. 2019. “Favicons, Touch Icons, Tile Icons, etc. Which Do You Need?” *CSS-Tricks*, January 11. https://css-tricks.com/favicon-quiz/.

Coti, Adam. 2016. “The Essential Meta Tags for Social Media.” *CSS-Tricks*, December 21. https://css-tricks.com/essential-meta-tags-social-media/.

Grundy, Julie. 2017. “The UX of Form Design: Designing an Effective Form.” Presentation at *UX New Zealand*, Shed 6, Wellington, New Zealand, November 1. http://www.uxnewzealand.com/2017-speakers/julie-grundy/.

MDN. 2020. “HTML: Hypertext Markup Language.” https://developer.mozilla.org/en-US/docs/Web/HTML.

Web Hypertext Application Technology Working Group (WHATWG). 2020. “HTML Standard.” https://html.spec.whatwg.org/multipage/.

12.2.6. HTTP (Hypertext Transfer Protocol)

MDN. 2020. “HTTP.” https://developer.mozilla.org/en-US/docs/Web/HTTP.

W3C. 2014. “HTTP—Hypertext Transfer Protocol.” https://www.w3.org/Protocols/.

12.2.7. Internationalization (i18n) and Localization (L10n)

Coupé, Jérôme. 2019a. “Language Switcher for Multilingual JAMstack Sites.” *Webstoemp*, September 12. https://www.webstoemp.com/blog/language-switcher-multilingual-jamstack-sites/.

———. 2019b. “Multilingual Sites with Eleventy.” *Webstoemp*, April 27. https://www.webstoemp.com/blog/multilingual-sites-eleventy/.

MDN. 2020. “Creating Localizable Web Applications.” https://developer.mozilla.org/en-US/docs/Mozilla/Localization/Web_Localizability/Creating_localizable_web_applications.

W3C. 2018. “Internationalization.” https://www.w3.org/standards/webdesign/i18n.

12.2.8. JAMstack, IndieWeb, and Static Site Generation

Biilmann, Mathias and Phil Hawksworth. 2019. *Modern Web Development on the JAMstack*. Sebastopol, CA: O’Reilly. https://www.netlify.com/oreilly-jamstack/.

Coyier, Chris. 2020. “The Power of Serverless for Front-End Developers.” https://serverless.css-tricks.com/.

IndieWeb Camp. 2020. “What is the IndieWeb?” https://indieweb.org/.

Leatherman, Zach. 2020. “Eleventy Documentation.” https://www.11ty.dev/docs/.

———. 2019. “Own Your Content on Social Media Using the IndieWeb.” *Non-Canonical Web Standards Fan Fiction*, October 18. https://www.zachleat.com/web/own-your-content/.

Netlify. 2020. “Netlify Docs.” https://docs.netlify.com/.

———. 2020. “StaticGen: Top Open Source Static Site Generators.” https://www.staticgen.com/.

12.2.9. JavaScript (ECMAScript)

Alicea, Anthony. 2015. “JavaScript: Understanding the Weird Parts.” http://learnwebdev.net/.

Crockford, Douglas. 2018. *How JavaScript Works*. San Jose, CA: Virgule-Solidus

———. 2008. *JavaScript: The Good Parts*. Sebastopol, CA: O’Reilly.

Ferdinandi, Chris. 2019a. “Vanilla JavaScript Pocket Guides.” https://vanillajsguides.com/.

———. 2019b. “The Vanilla JS Toolkit.” https://vanillajstoolkit.com/.

Haverbeke, Marijn. 2018. *Eloquent JavaScript: A Modern Introduction to Programming*, 3rd ed. San Francisco: No Starch. https://eloquentjavascript.net/.

JSDoc. 2017. “Use JSDoc.” https://jsdoc.app/.

MDN. “JavaScript.” 2020. https://developer.mozilla.org/en-US/docs/Learn/JavaScript.

Simpson, Kyle. 2019. *You Don't Know JS Yet*, 2nd ed., seven volumes. Frontend Masters. https://github.com/getify/You-Dont-Know-JS.

12.2.10. JSON (JavaScript Object Notation)

Crockford, Douglas. n.d. “JSON.” https://json.org/.

ECMA International. 2017. “ECMA-404 The JSON Data Interchange Syntax,” 2nd ed. https://www.ecma-international.org/publications-and-standards/standards/ecma-404/.

GeoJSON. 2016. “The GeoJSON Format.” https://geojson.org/.

Mapbox. n.d. “Tutorials.” https://docs.mapbox.com/help/tutorials/.

How To GraphQL. n.d. “The Fullstack Tutorial for GraphQL.” https://www.howtographql.com/.

12.2.11. Progressive Web Apps

Frain, Ben. 2019. “Designing And Building A Progressive Web Application Without A Framework,” three-part series. *Smashing Magazine*, July 23–30. https://www.smashingmagazine.com/2019/07/progressive-web-application-pwa-framework-part-3/.

MDN. 2020. “Progressive Web Apps.” https://developer.mozilla.org/en-US/docs/Web/Progressive_web_apps.

Rzutkiewicz, Jason and Jeremy Lockhorn. 2019. “Why Progressive Web Apps Are The Future of Mobile Web: 2019 Research.” *Y Media Labs*. https://ymedialabs.com/progressive-web-apps.

12.2.12. SVG (Scalable Vector Graphics)

Coyier, Chris. 2019. “Using SVG.” *CSS-Tricks*, May 2. https://css-tricks.com/using-svg/.

Drasner, Sarah. 2019. “SVG Will Save Us.” Presenation at *#PerfMatters Conference*, Canada College Theater, Redwood City, CA, April 2. Video, 36:42. https://www.youtube.com/watch?v=sxte3WpyO60.

W3C SVG Working Group. 2016. “Scalable Vector Graphics (SVG).” https://www.w3.org/Graphics/SVG/.

View this document online: https://reubenlillie.com/comp-496/